Supplementary Material: A Benchmark for Compositional Visual Reasoning

1 Social Impact

Our work presents a novel visual reasoning benchmark that evaluates sample efficiency and compositionality. Benchmarks have played an important role in guiding the development of Machine Learning and AI research. Since our benchmark exposes shortcomings of modern architectures in terms of sample efficiency and compositionality, it promotes the development of efficient and human-inspired solutions for visual reasoning and AI research in general. The byproduct of sample-efficient solutions is a lower training carbon footprint. Furthermore, our work encourages researchers to propose general solutions that can be applied to other task setups and extend beyond the simple odd-one-out task. Finally, this work has implications for cognitive science research since the benchmark can be utilized to test various hypotheses on non-verbal learning and abstraction in humans. We could not identify our research's negative ethical or social impact.

2 Dataset Generation

Prior Assumptions

- Object are defined as closed contours sampled randomly with 2 parameters: the radius of the largest circle that can fit inside the object and the distance between two consecutive points in the closed contour.
- The object size scales the width and height of the object linearly.
- The background color is always white and object colors are sampled in the HSV color space with saturations and values that maintain contour visibility in a white background.
- The center of an object is defined as the mid point between its horizontal and vertical extents. It is used for placing the object based on its position. By default, objects are not in contact and do not overlap unless they have an insideness or a contact relation. To ensure this, their positions are sampled with rejection sampling.

All parameter values are sampled from a fixed range for each task. Within the specified ranges, value combinations might violate priors. Rejection sampling is employed to avoid choosing such value combinations. For example, the object size is sampled such that all objects can fit inside the image based on the number of objects in the scene. The range of sizes decreases with the number of objects in the scene. Functions that sample positions within the scene, positions inside an object, positions of objects in contact, and colors are shared among all 103 programs. Position sampling rejects samples where at least one pair of object bounding boxes intersect. To sample contact between two objects, a direction (2D vector) is sampled or specified, then the maximum distance between two intersecting objects is computed and relative positions with respect to their center of mass are assigned to the objects. While sampling a position inside an object, samples are rejected if the position is outside the object or if contours intersect. When objects contain other objects, their shapes are sampled with a large inner radius. Objects are flipped either horizontally or vertically.

Dataset Statistics Among the 103 tasks, 9 are based on a single elementary relation, 20 are compositions of one elementary relation, 65 are compositions of a pair of relations and 9 are

36th Conference on Neural Information Processing Systems (NeurIPS 2022) Track on Datasets and Benchmarks.

compositions of more than 2 elementary relations. Figure 1 details the number of unique rules for each pair of elementary relations. The dataset was designed to include at least one unique rule for each pair. However, one of its limitations is the uneven distribution of unique rules across the pairs. This limitation could bias results in favor of models that perform better on pairs with a larger number of unique rules.

Generalization Sets Among the limitations of our dataset is that tasks do not account for shortcuts in the test split. Shortcuts are biases in the tasks that neural networks exploit to solve them. An explanatory example is the counting task. If objects have the same size, the neural network can easily solve the task by summing the pixels of the image without analyzing the scene. If a model exploits this shortcut, it does not need to learn the concept of an object or counting. To account for this limitation and evaluate out-of-distribution generalization, we develop a generalization test set that differs from the in-distribution test set in several ways.

- **Parameter value ranges**: the ranges used for sampling parameters, especially target-relation parameters, are changed.
- **Object contour specification**: the distance between dots in contours is randomized locally, resulting in fuzzier contours.
- **Sets of random and fixed parameters**: the generation process samples 1 value that is fixed across the 4 images for fixed parameters and samples 4 values for random parameters. In the generalization test set, the sets of random and fixed parameters are changed without affecting the rules.

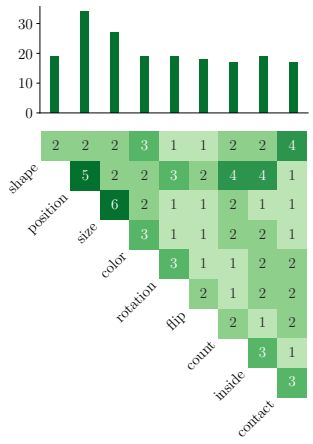


Figure 1: **Dataset rules**: Each square represents the number of rules that are a composition of the associated elementary relations and the bar plot shows the number of rules that involve each elementary relation.

3 Dataset Documentation

We provide the full dataset generation and model training code here. We bear all responsibility in case of violation of rights. The dataset and the generation methods are released with the Apache License, Version 2.0. The Github repository will reflect all updates to the dataset.

4 Experiment Details

Learning to spot the Odd-One-Out The training setup for standard vision models is straightforward; models are trained to represent the odd-one-out differently from the three other images. The four images of the problem are fed separately to the model. Their representations are transformed into a low-dimensional space where the distances between the four representations are computed. The cosine similarity of the odd-one-out to the group is minimized with a cross-entropy-based loss. Given the 4 image representations x_i, the logits y_i used for computing the softmax cross-entropy loss are the negative sum of the similarity scores.

$$y_i = -\sum_{j \neq i} \frac{x_i \cdot x_j}{||x_i||_2 \cdot ||x_j||_2}$$

Figure 2: **Shapes in the generalization test set.**

Although we follow a specific training setting for the baseline architectures, we do not impose this methodology for future work as more sophisticated methods for comparing the four images of a problem can be designed.

RPM Baselines In order to provide a fair comparison for models designed for solving RPMs, we adapt the odd-one-out task to the matrix and choice selection task setup. In the RPM setting, models are fed the nine panels with tags that indicate the position on the matrix. Each of the eight choice panels is concatenated individually to the eight context panels. The model outputs a logit for each of the eight matrices used to compute the cross-entropy loss. The training process is explained in detail here [2]. We discard the position tags in our setting since the four images have no sense of progression. We replace context panels with the four problem images and use the same four images as choice panels, the correct choice being the outlier.

Self-Supevised Pretraining The SSL pretraining objective function maximizes the similarity between transformations of the same image. When SSL is performed on datasets such as ImageNet [7], where the downstream task is classification, the images are transformed in ways that maintain the class information. These transformations are augmentations, including spatial transformations such as random cropping and flipping, noise (eg. additive Gaussian noise or blurring), and color transformations such as grey-scale and color jittering. In our setup, we use the augmentations: random resize, random Gaussian blurring, random horizontal flips, and random rotations in multiples of 90 degrees. To ensure the variety of images covers all structures used in the dataset, we select one image from each problem in the dataset, in total 1M images. This number of images is equivalent to the number of images in the ImageNet dataset, which is customarily used in SSL pretraining.

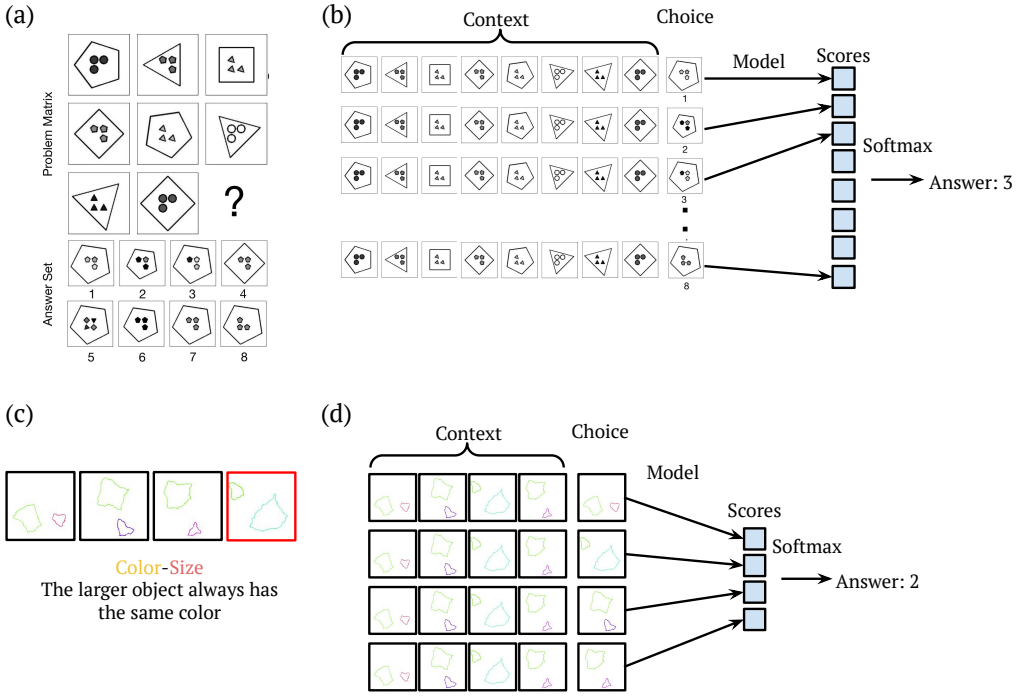


Figure 3: **RPM training setup**: (a) A sample RPM problem adapted from [28], the matrix contains context panels, and a choice is taken from the answer set. (b) Inference in RPM models, SCL and WReN. The model takes all context panels with one of the choices and outputs a score. These scores from 8 choices are used for computing the cross-entropy loss. (c) An odd-one-out problem based on size and color. (d) The problem is adapted to RPM by placing all images as context and choice. The odd-one-out has the highest score among the choices.

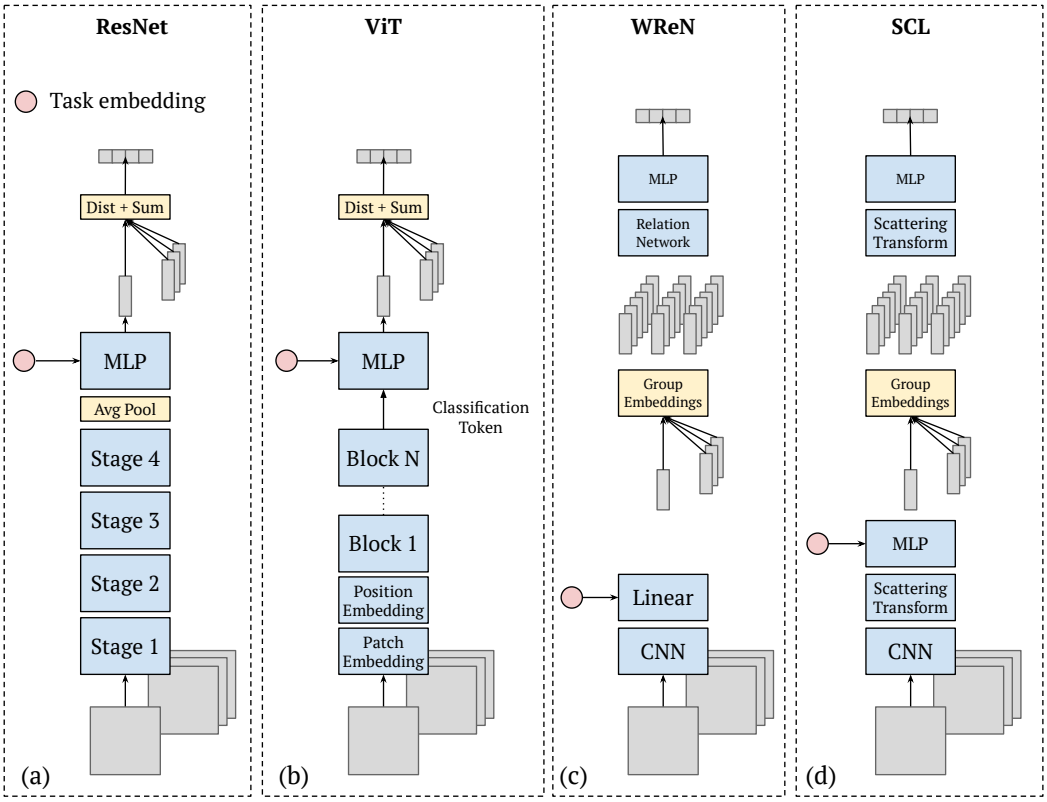


Figure 4: **Model Architectures**: (a) ResNet [13] stages consist of several residual blocks. (b) The patch embedding in transformers splits the image into patchs and transforms them into embeddings. Each ViT [9] block consists of self-attention blocks and MLP transformations, ViT-small uses 12 blocks. (c) WReN [2] is trained in the RPM setting, each image is processed by a CNN then all image embeddings are processes by a Relation Network. (d) Similarly to WReN, SCL [26] is also trained in the RPM setting. Each image is processed by a CNN and a scattering transformation. All image embeddings are processed by a second scattering transformation. In SCL-ResNet-18, the CNN encoder is substituted with ResNet-18. Details of model architectures can be found in their respective references.

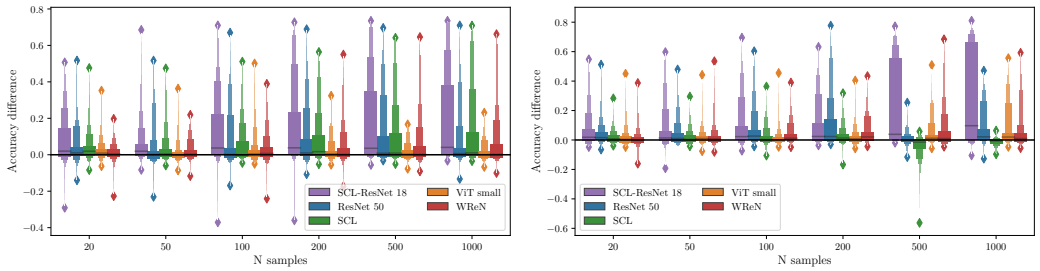
5 Architectures and Hyperparameters

We adapt model architectures from reference implementations: ResNet-50[1], ViT[6][2], SCL [3] and WReN [4]. We also endow SCL with ResNet18, a strong vision backbone. We name this architecture SCL-ResNet-18. All model architectures All models are trained using images with size 128×128 pixels. Figure 4 illustrates the architectures of all the baselines.

In preliminary experiments, we hand-tuned the learning rate and weight decay for all models and selected the hyperparameters that achieved the highest performance in the joint training setting for each model. We also analysed the random seed effect and observed that training results are robust with respect to the seed.

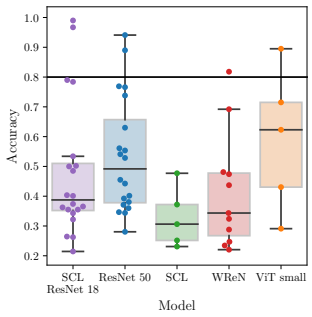
We equipe standard vision models, ViT-small and Resnet 50, with an MLP that extracts task-specific information. It takes as input image features and the task embedding and outputs a lower dimensional vector used for computing the pairwise distances and the loss. The MLP contains 2 layers, the hidden layer's size is 2048 and the output size is 128. The task embedding space has 64 dimensions. All

[1] https://pytorch.org/vision/stable/models.html
[2] https://github.com/facebookresearch/moco-v3
[3] https://github.com/dhh1995/SCL
[4] https://github.com/Fen9/WReN



Curriculum - Indivudual setting

Curriculum - Joint setting



Reverse Curriculum

Figure 5: **Compositionality**: We evaluate models' capacity to reuse previous knowledge. **Curriculum**: Models trained with a curriculum are compared to models trained from scratch. The distribution of difference in accuracy across tasks is plotted for each model. **Reverse Curriculum**: In the 1000 samples data regime, we pick rules for which models achieved higher than 80% accuracy and we evaluate them on the respective elementary rules.

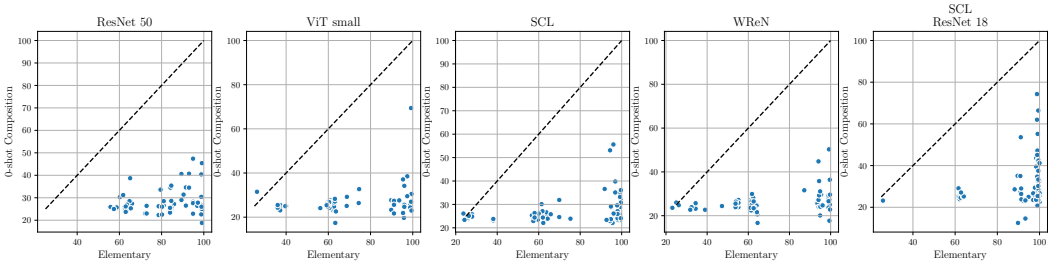


Figure 6: **Compositionality**: Models trained on elementary tasks are zero-shot evaluated on their compositions. Models fail at all compositions without finetuning.

models were trained using Adam optimizer [17] for 100 epochs with early stopping after 30 epochs. The mini-batch size used for training all models is 64, except when the training set is smaller. The learning rates are scaled linearly with the batch size. The learning rate and weight decay values are provided in 1. All experiments were conducted on an internal cluster. We used 1500 GPU hours on NVIDIA V100, TitanRTX and QuadroRTX.

6 Additional Results

We provide more results on compositionality evaluation in Figure 5, Figure 6, Table 4 and Table 2. Results on the joint training setting are consistent with the individual training setting results. We also provide results on the generalization test set in Table 3 and Figure 7. Task difficulty is expanded with more analysis in Figure 8 and Figure 9. Finally, we use attribution methods to visualize gradient maps of ResNet-50 in Figure 10.

	Backbone Params	Total Params	learning rate	weight decay
ResNet 50	23.5 M	28.1 M	0.0001	0.0001
ViT-small	21.6 M	21.8 M	0.00001	0.0001
SCL	176 k	176 k	0.001	0.0001
WReN	1.5 M	1.5 M	0.0001	0
SCL-ResNet 18	11.2 M	11.6 M	0.0005	0.0001

Table 1: **Model sizes and training hyperparameters**.

Model	Score
ResNet 50	12.1
ViT small	2.62
SCL	12.6
WReN	6.76
SCL-ResNet 18	**23.1**

Table 2: **Compositionality**: Models are quantitatively evaluated in the curriculum condition. The score is the maximum gain in accuracy across data regimes computed for each task then averaged across tasks. We observe that the qualitative advantage for SCL-ResNet-18 is consistent with the quantitative evaluation.

7 Comparison to SVRT

Synthetic Visual Reasoning Test [11] (SVRT) is a suite of 23 tasks developed for comparing machines to humans on the semantic description of visual scenes. Each test is a binary classification task based on the rules involved in generating those images. Each image contains randomly generated close contour objects based on a rule such as a similarity judgment, spatial reasoning or numerosity. SVRT tasks were designed such that binary classes cannot be separated based on the appearance of objects, spatial positioning, or any geometric or topological properties of scene components. Figure 12 shows some SVRT examples. CVR takes inspiration from SVRT's scene design – object contours on a white background and the rules for generating scenes. However, with a set of elementary relations and a method for combining them with compositionality prior, the 103 tasks proposed in CVR are more diverse than SVRT tasks. CVR also uses the Odd-One-Out task setting, which enables a more general instantiation of rules. For example, task #7 in SVRT requires dissociating images of 3 groups of 2 similar shapes from images of 2 groups of 3 similar shapes, as shown in Figure 12. This task is generalized in CVR to a *shape-count* rule where images of n groups of m objects are to be discriminated from images with different counts. In this regard, the odd-one-out task can be considered a 4-shot learning setting for SVRT tasks. Furthermore, CVR is a systematic reorganization of SVRT based on compositionality. It can be used for evaluating generalization, transfer learning and compositionality, unlike attainable with the SVRT.

8 Behavioral Experiments

We used 20 problem samples for each rule, which corresponds to the lowest number of samples used for training baseline models. We recruited 21 participants from Prolific; 13 females and 8 males aged between 19 and 49 years. All participants signed a consent form prior to participation and received $10.50 US per hour for participation. The study was approved by the Institutional Review Board of Brown University. 40 individuals were initially enrolled to participate, but 19 were disqualified based on technical malfunctions, misunderstanding of instructions or failure on attention checks. Participants were instructed to identify the odd stimulus violating the rule they had to infer over a series of trials. Prior to the practice phase, they were quizzed on their understanding of the task. Participants practiced the task on a separate set of visual stimuli different from the benchmark. During the experiment, participants were informed about the start of each block as well as the concomitant rule switch. For each trial, they were presented with 4 choices on the screen and instructed to choose

N train samples			20		50		100		200		500		1000		SES	AUC
rand-init	ind	ResNet-50[13]	26.3	0	28.1	0	29.1	1	30.3	2	31.5	3	34.3	6	29.4	29.9
		ViT-small[9]	26.9	0	28.1	0	28.8	1	29.4	2	30.2	3	31.6	2	28.8	29.2
		SCL[26]	26.0	0	27.8	0	28.0	0	27.9	0	28.5	1	29.7	2	27.7	28.0
		WReN[2]	27.2	0	28.8	1	29.4	1	30.1	2	31.4	3	32.3	5	29.5	29.9
		SCL-ResNet-18	**28.8**	**1**	**31.1**	**1**	**31.7**	**3**	**32.4**	**4**	**34.4**	**6**	**38.4**	**11**	**32.2**	**32.8**
	joint	ResNet-50[13]	26.0	0	26.6	0	27.8	1	30.0	1	**37.3**	**7**	**41.3**	**15**	**30.3**	**31.5**
		ViT-small[9]	**26.2**	0	26.4	0	26.6	1	26.9	1	27.4	1	26.9	1	26.6	26.7
		SCL[26]	25.4	0	25.6	0	27.5	0	30.3	0	33.6	5	35.6	8	28.8	29.6
		WReN[2]	26.1	0	25.9	0	26.8	0	27.8	0	31.9	4	34.1	6	28.1	28.8
		SCL-ResNet-18	26.0	0	**27.0**	0	**29.9**	**3**	**32.1**	**4**	34.7	6	37.9	6	**30.3**	31.3
SSL	ind	ResNet-50	32.0	5	37.0	6	38.8	**10**	40.9	**10**	42.4	**12**	44.4	**17**	38.3	39.2
		ViT-small	**36.2**	5	**39.7**	**7**	**40.6**	7	**41.7**	7	**43.2**	9	**45.3**	14	**40.5**	**41.1**
	joint	ResNet-50	34.0	4	**34.3**	4	**37.9**	**8**	**38.4**	6	**46.4**	**15**	**51.0**	**20**	**39.0**	**40.3**
		ViT-small	**34.1**	**6**	33.0	**5**	32.5	4	33.2	6	33.4	6	35.9	11	33.6	33.7

Table 3: **Out-Of-Distribution Generalization Results**: Most models perform significantly worse on the generalization test set.

the image which seemed to be different according to the rule that they had to learn. They rated confidence in their choice and received feedback after each trial. In addition, they were asked to describe the rule at the end of each block.

9 Model Design For Sample Efficiency And Compositionality

This paper proposes a benchmark that encourages the development of visual reasoning models with better sample efficiency, OOD generalization, transfer learning, and that harness compositionality. In the literature, general-purpose models such as ViTs and CNNs are provided as baselines with more complex approaches relying on additional inductive biases for reasoning such as RNNs, GNNs, and Relation Networks [16, 22, 4]. These architectures achieve decent performance but have a poor generalization and sample efficiency. More promising solutions for visual reasoning use the idea of modularity [1, 5, 15, 14, 19, 20, 12]. Modular neural networks are composed of a set of modules that perform different operations. These models are generally orchestrated by a controller module that executes language-based instructions. We conjecture that modularity could be a fundamental inductive bias for compositionality. When equipped with a proper controller module and information routing mechanisms, a modular network could flexibly manipulate novel concepts and build contextual representations. Although these models bring the advantage of interpretability and better OOD generalization, they are notoriously difficult to train. Other methods focus on scene decomposition [3, 10, 18], these models rely on attention and object-centered representations as inductive biases for building scene representations which are useful for visual reasoning [8]. In another vein, certain approaches scale up simple architectures, based on transformers and convolutions, and rely on self-supervised pretraining to achieve impressive performance on several multi-modal computer vision tasks [21, 27]. However, the capacity of these models at leveraging compositionality is limited by their architectural components; Transformers and ResNets. We believe that modularity, attention, and objectness are essential inductive biases to achieve sample efficiency and compositionality in CVR. Attention is used for extracting the scene graph from the image while the modules implement various strategies to solve different visual reasoning tasks. We believe that future models of visual reasoning should implement these inductive biases while taking inspiration from human cognition in orchestrating visual reasoning as program execution.

		N train samples	20	50	100	200	500	1000
individual	ResNet-50	rand init	25.9	27.7	28.5	29.4	32.6	39.0
		transfer	30.0	30.7	34.2	36.9	42.3	45.0
		difference	4.10	3.06	5.73	7.47	11.5	5.75
	ViT-Small	rand init	25.9	27.0	27.5	28.6	30.4	31.2
		transfer	28.3	27.9	30.0	30.7	31.9	32.9
		difference	2.41	0.89	2.51	2.11	1.44	1.21
	SCL	rand init	26.2	29.3	29.6	29.4	30.6	32.0
		transfer	30.3	32.7	34.9	37.4	40.1	43.0
		difference	4.11	3.43	5.27	7.93	9.49	11.0
	WReN	rand init	28.8	30.5	30.9	31.4	32.4	34.6
		transfer	29.8	32.5	34.0	35.2	37.7	40.4
		difference	1.04	2.03	3.08	3.78	5.28	5.79
	SCL-ResNet-18	rand init	28.7	33.3	32.9	35.5	37.6	41.3
		transfer	36.3	39.1	45.6	49.1	55.1	61.2
		difference	7.60	5.80	12.7	13.6	17.5	20.0
joint	ResNet-50	rand init	25.5	26.2	26.6	29.2	48.6	55.7
		transfer	29.8	30.5	33.8	40.0	49.4	62.9
		difference	4.31	4.30	7.19	10.8	0.86	7.13
	ViT-Small	rand init	25.6	26.0	26.3	26.5	27.0	28.1
		transfer	27.6	27.9	27.9	28.5	31.2	34.9
		difference	2.00	1.97	1.56	1.95	4.14	6.84
	SCL	rand init	25.3	26.2	26.9	27.2	41.8	45.1
		transfer	27.2	28.0	29.0	30.1	37.6	44.0
		difference	1.87	1.90	2.11	2.90	-4.22	-1.07
	WReN	rand init	27.0	26.9	27.7	29.1	33.8	39.1
		transfer	28.0	29.5	31.5	34.2	40.0	44.4
		difference	1.06	2.56	3.86	5.04	6.20	5.21
	SCL-ResNet-18	rand init	26.2	27.5	27.6	30.1	25.8	26.1
		transfer	32.5	34.3	36.9	40.0	48.6	55.5
		difference	6.33	6.76	9.34	9.91	22.7	29.4

Table 4: **Curriculum Condition**: Models are pretrained on the elementary tasks before finetuning on the complex tasks (transfer). They are compared to models trained from a random initialization (rand init).

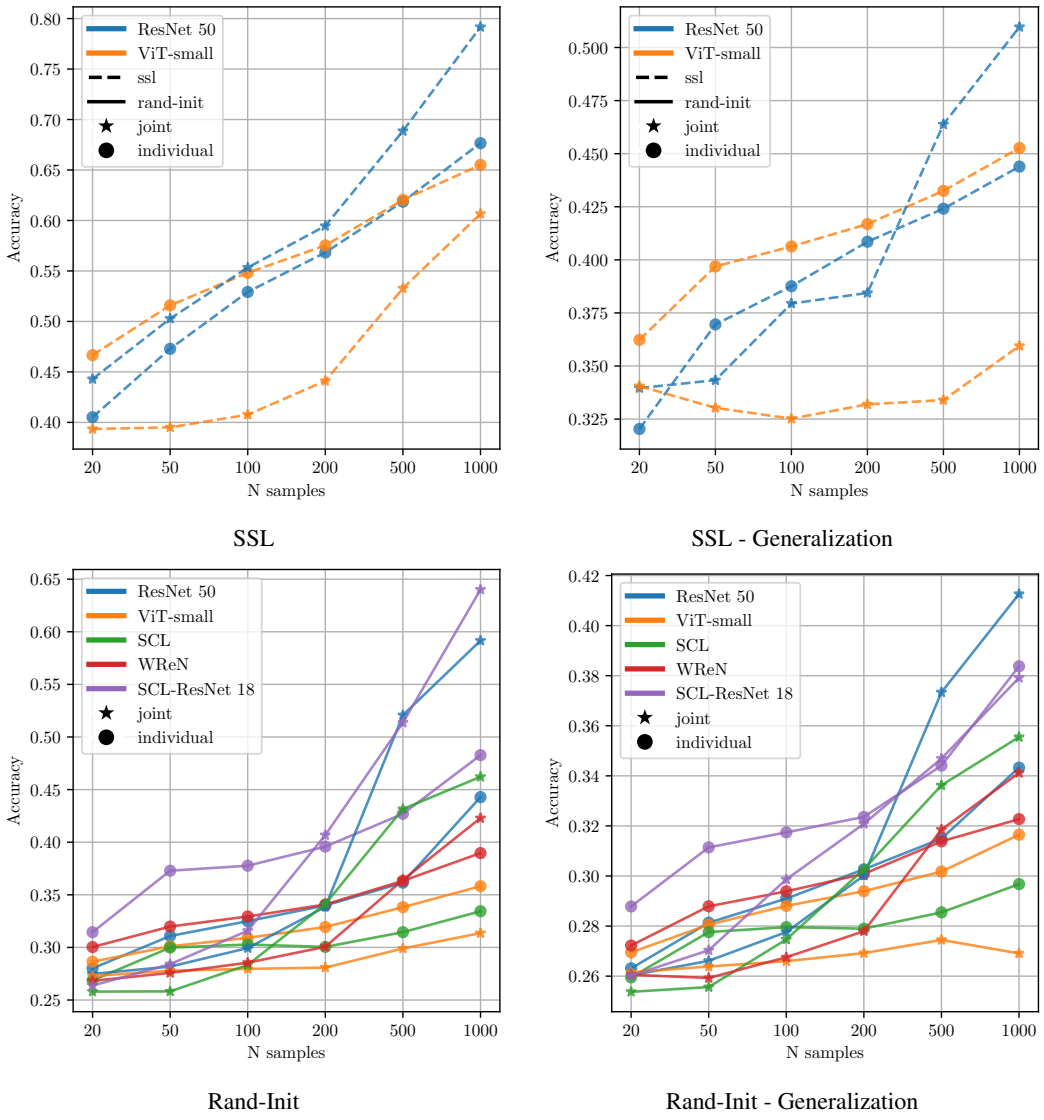


Figure 7: **Performance across settings for all the models**. The accuracy is aggregated over all rules. Random choice accuracy is 0.25.

Figure 8: **Task difficulty**: Average accuracy on the elementary rules and their pair-wise compositions. **Individual vs. Joint**: Models are trained on each rule separately or trained jointly on all rules. **Rand-Init vs. SSL**: models are randomly initialized or pretrained with self-supervision.

Figure 9: **Task Difficulty Analysis**: The difference in SES per task is computed in various configurations. **Joint vs. Individual rule learning**: highlights the benefit of joint rule learning compared to learning rules separately. Results vary over spatial tasks; while some models benefit from joint learning in these tasks (SCL and ResNet50), others have the opposite effects (ViT-small and SCL-ResNet18). **Initializations**: ResNet50 is pretrained in various settings. Initializations benefit downstream CVR performance differently. We observe that pretraining improves performance over elementary tasks overall for ResNet50. SSL pretraining improves performance on spatial and color-based tasks while ImageNet and CLIP pretraining differ mostly in color based tasks. **Models**: The performance in the joint rule learning setting is compared across models. The comparison shows variations in performance over elementary tasks and spatial tasks mainly.

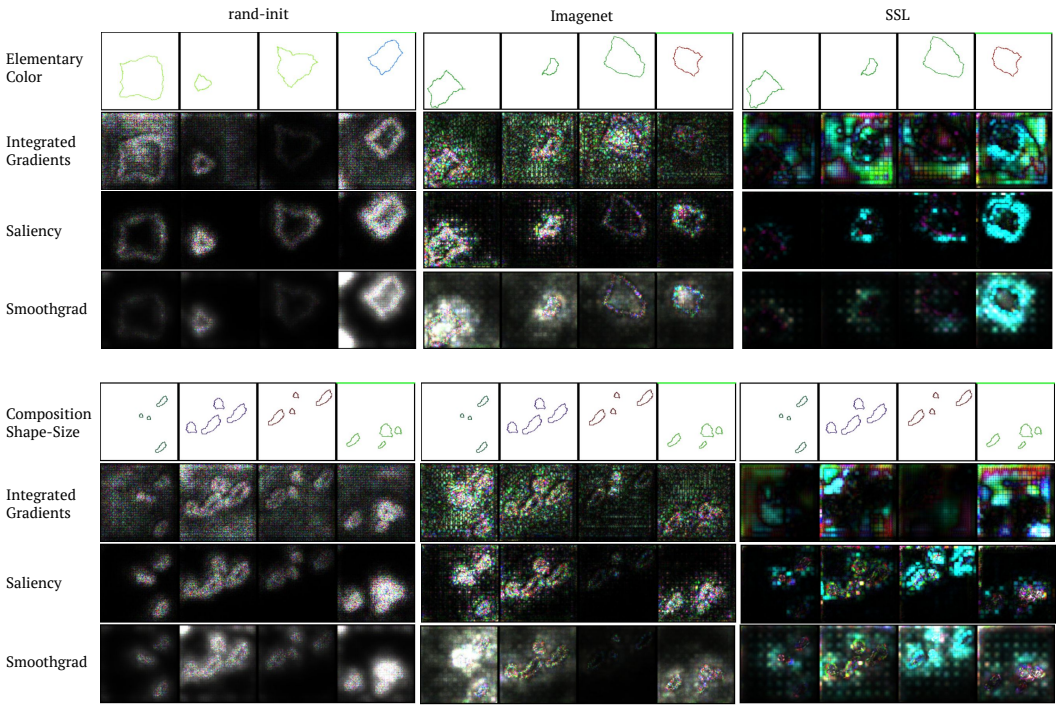


Figure 10: **Gradient-based Attribution**: Attribution methods are used for explaining ResNet50 decisions (Integrated Gradients [25], Saliency [23] and SmoothGrad [24]). Gradients are focused on object contours and their surroundings in general. It's difficult to extract task-relevant interpretations from the visualizations.

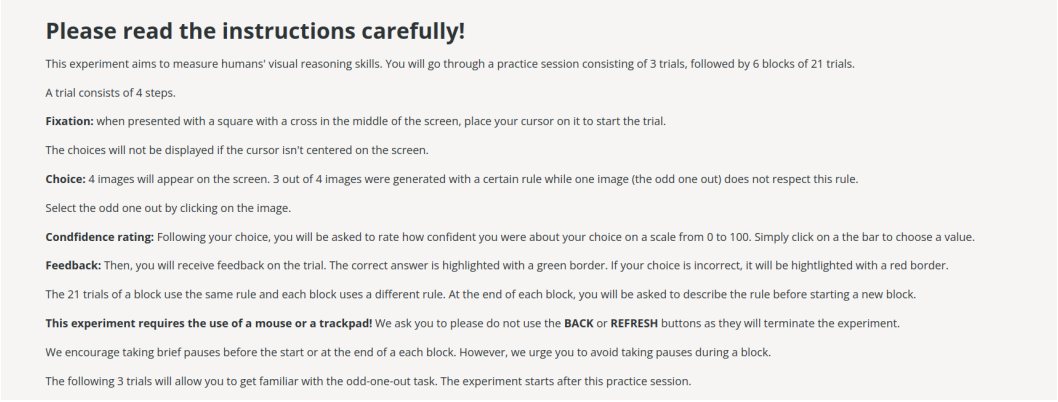


Figure 11: **Behavioral experiment instructions.**

Algorithm 1: Problem Generation Program: Generates problem samples of a counting-contact task in Figure[3] of the paper.

$n = sample\_number(range = [6, 10])$
$s_0 = sample\_size()$
$[n_a, n_b] = sample(n = 2, sum = n)$
$[n'_a, n'_b] = sample(n = 2, sum = n, reject = [n_1, n_2])$
for $i \leftarrow 1$ **to** 4 **do**
  if $i = 4$ **then**
    `// Odd-One-Out`
    $[n_1, n_2] \leftarrow [n'_a, n'_b]$
  else
    $[n_1, n_2] \leftarrow [n'_a, n'_b]$
  end
  $[o]_{i,1,1-n_1} \leftarrow sample\_shapes(n = n_1)$
  $[o]_{i,2,1-n_2} \leftarrow sample\_shapes(n = n_2)$
  $sc_1, scs_1 = \leftarrow sample\_contact(o = [o]_{i,1,1-n_1}, s = s_0)$
  $sc_2, scs_2 = \leftarrow sample\_contact(o = [o]_{i,2,1-n_2}, s = s_0)$
  $[scp]_{1-2} \leftarrow sample\_position([scs]_{1-2})$
  $[p]_{(i,1,1-n)} = scp_1 + sc_1$
  $[p]_{(i,2,1-n)} = scp_2 + sc_2$
  $[s]i, 1-2, 1-n = s_0$
  $c = sample\_color(n = 1)$
  $[c]i, 1-2, 1-n = c$
end
$[scene]_{1-4} = [[o, p, s, c]_{i,1-2,1-n}]_{1-4}$
$[image]_{1-4} = [render(scene)]_{1-4}$

Algorithm 2: Problem Generation Program: Generates problem samples of a position-rotation task in Figure[3] of the paper.

$n = sample\_number(range = [4, 6])$
$o_0 = sample\_shape(n = 1)$
$scs_0 = sample\_size()$
$s_0 = sample\_size(scale = scs_0)$
for $i \leftarrow 1$ **to** 4 **do**
  `// spatial config`
  $sc \leftarrow sample\_sc()$
  $[a_1, a_2] \leftarrow sample\_angle(n = 2)$
  $sc_1 \leftarrow rot(sc, a_1)$
  if $i = 4$ **then**
    `// Odd-One-Out`
    $sc_2 \leftarrow sample\_sc(reject = sc_1)$
  else
    $sc_2 \leftarrow rot(sc, a_2)$
  end
  $[scp]_{1-2} \leftarrow sample\_position([scs]1_2)$
  $[p]_{(i,1,1-n)} = scp_1 + sc_1 \times scs$
  $[p]_{(i,2,1-n)} = scp_2 + sc_2 \times scs$
  $[o]_{i,1,1-n} = rot(o_0, a1)$
  $[o]_{i,2,1-n} = rot(o_0, a2)$
  $[s]_{i,1-2,1-n} = s_0$
  $c = sample\_color(n = 1)$
  $[c]_{i,1-2,1-n} = c$
end
$[scene]_{1-4} = [[o, p, s, c]_{i,1-2,1-n}]_{1-4}$
$[image]_{1-4} = [render(scene)]_{1-4}$

Algorithm 3: Problem Generation Program: Generates problem samples of a size-color task in Figure[3] of the paper.

$c_0 = sample\_color(n = 1)$
for $i \leftarrow 1$ **to** 4 **do**
  $c_i^a \leftarrow sample\_color(reject = c_0)$
  $c_i^b \leftarrow c_0$ $s_i^a = sample\_size()$
  $s_i^b = s_i^a \times rand([1/4, 1/2])$
  $[p_i^a, p_i^b] \leftarrow sample\_position([s_i^a, s_i^b])$
  $[o_i^a, o_i^b] = sample\_shapes(n = 2)$
  if $i = 4$ **then**
    `// Odd-One-Out`
    $[c_i^a, c_i^b] \leftarrow [c_i^b, c_i^a]$
  end
end
$[scene]_{1-4} = [[o, p, s, c]^{a-b}]_{1-4}$
$[image]_{1-4} = [render(scene)]_{1-4}$

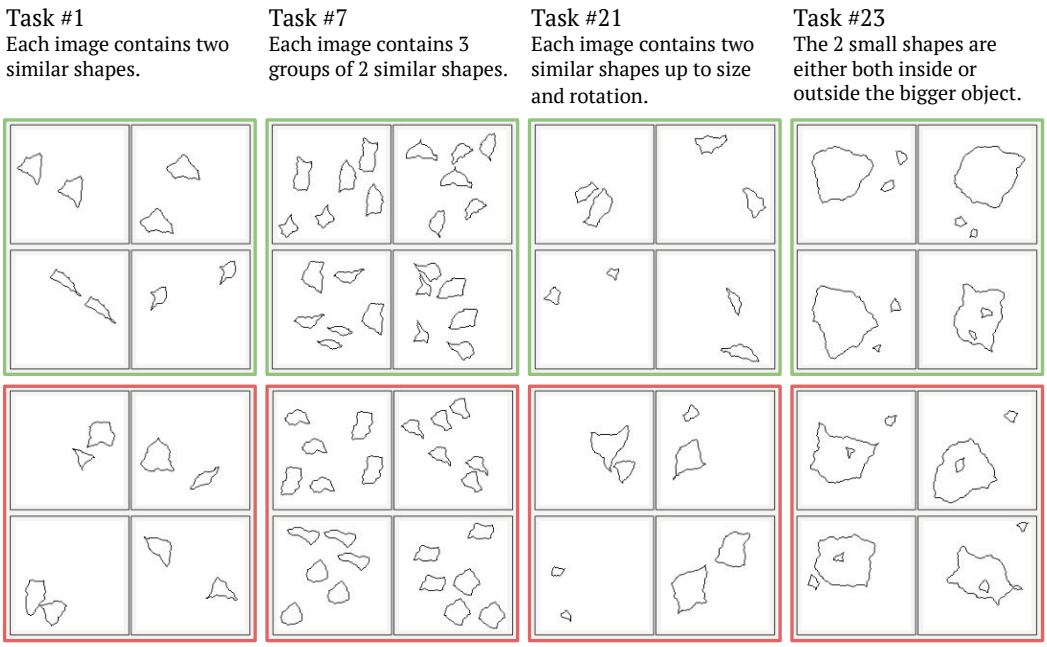
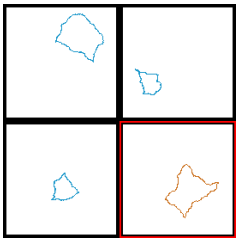
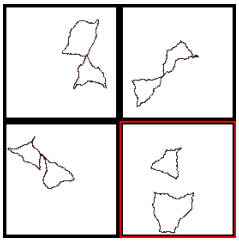


Figure 12: **SVRT task examples**: positive examples are highlighted by a green border and negative examples are highlighted by a red border.

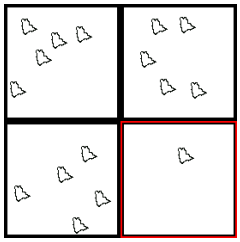
10 Rule Examples



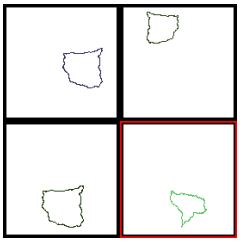
The hue of the object is constant



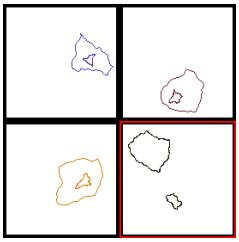
Two objects are in contact



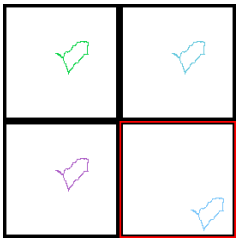
The number of objects is constant



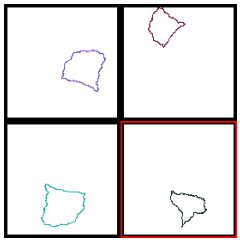
Flips of the same object



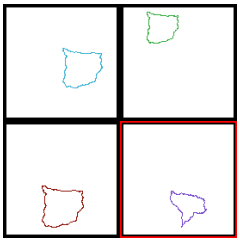
An object contains another object



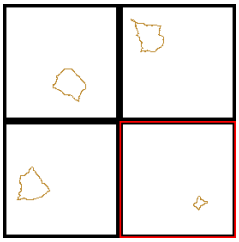
The object is always in the same position



Rotations of the same object



The shape is constant



The size of the object is constant

Figure 13: **Elementary rules**

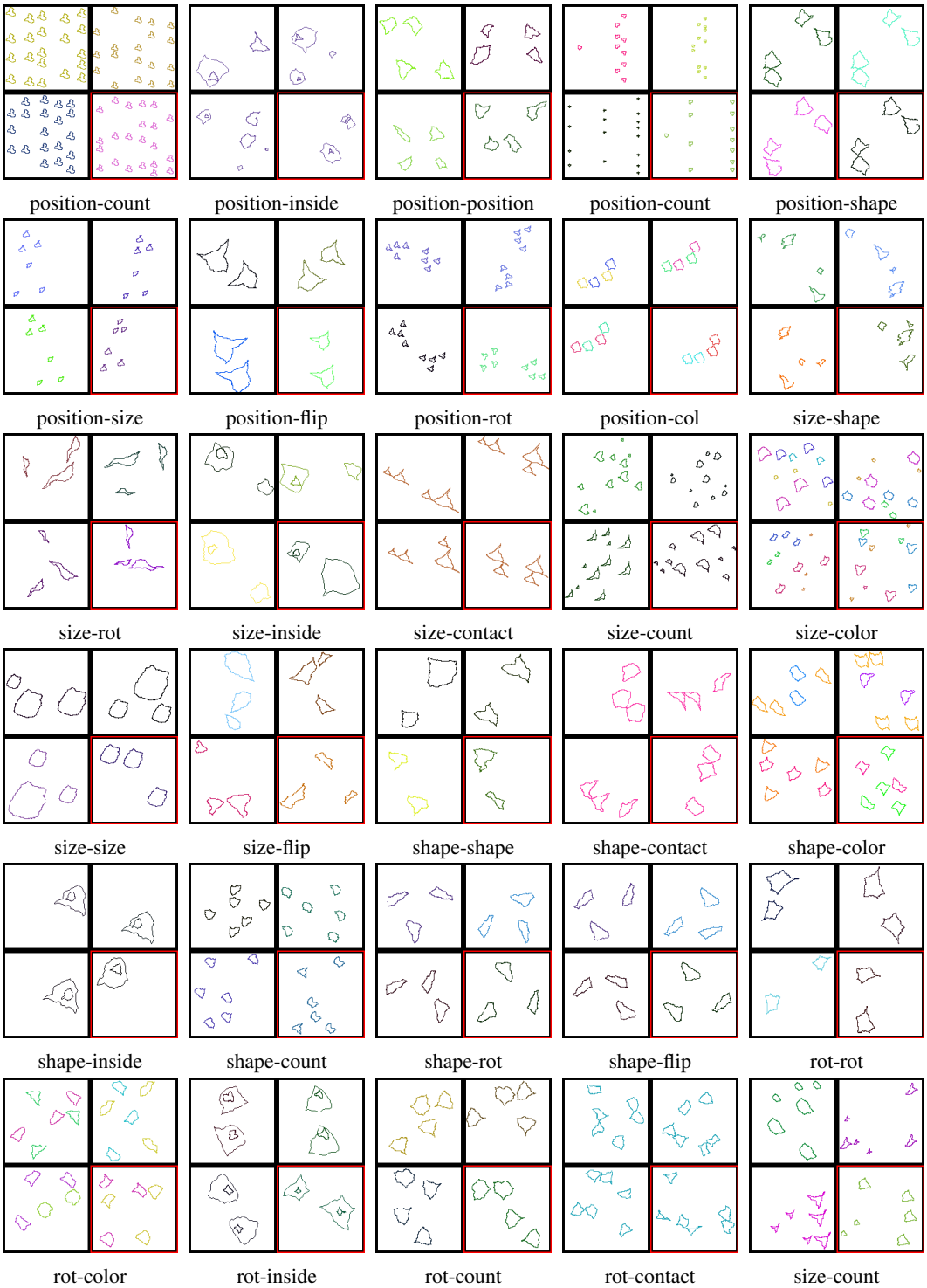


Figure 14: **Composition rules**

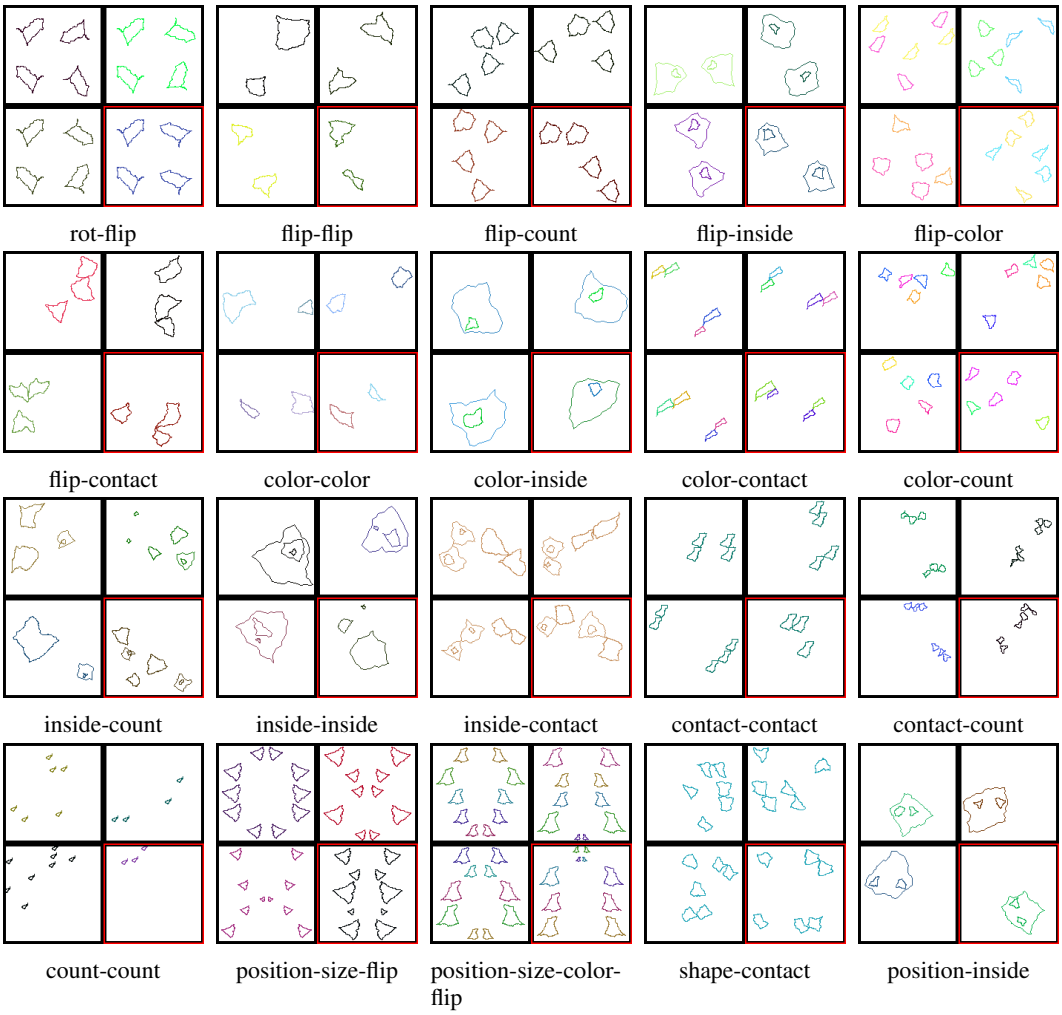


Figure 15: **Composition rules**

11 Datasheet

11.1 Motivation

For what purpose was the dataset created? Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.

CVR was created for evaluating the sample efficency and compositionality of visual reasoning models. An important difference between current non-verbal reasoning benchmarks is the number of unique task rules they compose out of their priors and the number of samples available for training architectures on individual rules. On one end, ARC has a large number of unique tasks (1000) and very low number of training samples (3). On the other end, SVRT, RAVEN and PGM have a low number of unique rules and a large number of training samples. These fundamental differences explain the success of neural architectures in one case; few unique tasks and many training samples, and their failure in the other; many unique tasks and few training samples. CVR fills a gap between the current benchmarks by proposing a large number of unique tasks and a reasonable number of training samples. Furthermore, our benchmark encourages the development of more sample-efficient and versatile neural architectures by evaluating both aspects.

Who created this dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?

This dataset was created by Aimen Zerroug, Mohit Vaishnav, Julien Colin, Sebastian Musslick and Thomas Serre, in affiliation with ANITI, Serrelab, Brown University and CerCo.

Who funded the creation of the dataset? If there is an associated grant, please provide the name of the grantor and the grant name and number.

This work was supported by ONR (N00014-19-1-2029), NSF (IIS-1912280 and EAR-1925481), DARPA (D19AC00015), NIH/NINDS (R21 NS 112743), and the ANR-3IA Artificial and Natural Intelligence Toulouse Institute (ANR-19-PI3A-0004). Additional support provided by the Carney Institute for Brain Science and the Center for Computation and Visualization (CCV) and CALMIP supercomputing center (Grant 2016-p20019, 2016-p22041) at Federal University of Toulouse Midi-Pyrénées. We acknowledge the Cloud TPU hardware resources that Google made available via the TensorFlow Research Cloud (TFRC) program as well as computing hardware supported by NIH Office of the Director grant S10OD025181.

11.2 Composition

What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.

The dataset is generated synthetically. Each sample consists of an odd-one-out problem that contains 4 images. The goal is to spot outlier among the 4 images.

How many instances are there in total (of each type, if appropriate)? The dataset contains 103 unique tasks, that we call rules. For each task, it contains 10000 training samples, 500 validation samples, 1000 test samples and 1000 generalization test samples.

Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set? If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).

Since the dataset is synthetic, an arbitrary number of samples can be generated using the provided code. We limit the number of generated samples to 10000 and the random seed for benchmarking consistency.

What data does each instance consist of? "Raw" data (e.g., unprocessed text or images) or features? In either case, please provide a description.

Each odd-one-out problem consist of 4 images. Each image contains objects defined as closed contours. The outlier can be detected based on relationships between objects.

Is there a label or target associated with each instance? If so, please provide a description.

The label in each problem is the index of the outlier.

Is any information missing from individual instances? If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.

No information is missing, data is complete.

Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)? If so, please describe how these relationships are made explicit.

A task token is provided with each sample.

Are there recommended data splits (e.g., training, development/validation, testing)? If so, please provide a description of these splits, explaining the rationale behind them.

We provide training, validation, test and generalization test splits for each task. They are generated using different random seeds.

Are there any errors, sources of noise, or redundancies in the dataset? If so, please provide a description.

Since the generation process relies on random sampling from uniform distributions, it is possible to generate the same problem twice. However, this possibility is minimal because all generation processes use several continuous and discrete variables.

Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)? If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a future user? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.

The dataset is self-contained.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor-patient confidentiality, data that includes the content of individuals non-public communications)? If so, please provide a description.

No, the dataset is synthetic.

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety? If so, please describe why.

No, the dataset is synthetic.

Does the dataset relate to people? If not, you may skip the remaining questions in this section.

No, the dataset contains no human-related data.

Does the dataset identify any subpopulations (e.g., by age, gender)? If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.

N/A

Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset? If so, please describe how.

N/A

Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals racial or ethnic origins, sexual orientations, religious beliefs, political opinions or

union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)? If so, please provide a description.

N/A

Any other comments?

11.3 Collection Process

N/A This data is synthetic, no data collection process was involved.

11.4 Preprocessing/cleaning/labeling

Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)? If so, please provide a description. If not, you may skip the remainder of the questions in this section.

No processing is applied.

Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)? If so, please provide a link or other access point to the "raw" data.

N/A

Is the software used to preprocess/clean/label the instances available? If so, please provide a link or other access point.

N/A

11.5 Uses

Has the dataset been used for any tasks already? If so, please provide a description.

This is the first use for the dataset.

Is there a repository that links to any or all papers or systems that use the dataset? If so, please provide a link or other access point.

N/A

What (other) tasks could the dataset be used for?

Th dataset is created for non-verbal visual reasoning. However, it can be modified to incorporate language and be used for tasks such as visual question answering (VQA).

Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses? For example, is there anything that a future user might need to know to avoid uses that could result in unfair treatment of individuals or groups (e.g., stereotyping, quality of service issues) or other undesirable harms (e.g., financial harms, legal risks) If so, please provide a description. Is there anything a future user could do to mitigate these undesirable harms?

No, the dataset contains no human-related data. There is no risk for harm.

Are there tasks for which the dataset should not be used? If so, please provide a description.

The dataset is intended for neural network architecture evaluation only.

11.6 Distribution

Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created? If so, please provide a description.

Yes, the dataset is publicly available.

How will the dataset will be distributed (e.g., tarball on website, API, GitHub) Does the dataset have a digital object identifier (DOI)?

The dataset is available in a Github repository.

When will the dataset be distributed?

The dataset is distributed in 2022.

Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)? If so, please describe this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.

The dataset is released under the Apache License, Version 2.0.

Have any third parties imposed IP-based or other restrictions on the data associated with the instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms, as well as any fees associated with these restrictions.

N/A

Do any export controls or other regulatory restrictions apply to the dataset or to individual instances? If so, please describe these restrictions, and provide a link or other access point to, or otherwise reproduce, any supporting documentation.

N/A

11.7 Maintenance

Who will be supporting/hosting/maintaining the dataset?

Aimen Zerroug is supporting and maintaining the dataset.

How can the owner/curator/manager of the dataset be contacted (e.g., email address)?

aimen_zerroug@brown.edu

Is there an erratum? If so, please provide a link or other access point.

All fixes and updates to the dataset will be visible in reflected in the Github repository.

Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)? If so, please describe how often, by whom, and how updates will be communicated to users (e.g., mailing list, GitHub)?

This information can be found on the github repository.

If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were individuals in question told that their data would be retained for a fixed period of time and then deleted)? If so, please describe these limits and explain how they will be enforced.

N/A

Will older versions of the dataset continue to be supported/hosted/maintained? If so, please describe how. If not, please describe how its obsolescence will be communicated to users.

N/A

If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so? If so, please provide a description. Will these contributions be validated/verified? If so, please describe how. If not, why not? Is there a process for communicating/distributing these contributions to other users? If so, please provide a description.

Contributors are allowed and welcome to do so and should contact the authors about incorporating changes.

References

[1] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 39–48, 2016.

[2] David Barrett, Felix Hill, Adam Santoro, Ari Morcos, and Timothy Lillicrap. Measuring abstract reasoning in neural networks. In *International conference on machine learning*, pages 511–520. PMLR, 2018.

[3] Christopher P Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *arXiv preprint arXiv:1901.11390*, 2019.

[4] Tina Chen, Renran Tian, and Zhengming Ding. Visual reasoning using graph convolutional networks for predicting pedestrian crossing intention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3103–3109, 2021.

[5] Wenhu Chen, Zhe Gan, Linjie Li, Yu Cheng, William Wang, and Jingjing Liu. Meta module network for compositional visual reasoning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 655–664, 2021.

[6] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[8] David Ding, Felix Hill, Adam Santoro, Malcolm Reynolds, and Matt Botvinick. Attention over learned object embeddings enables complex visual reasoning. *Advances in neural information processing systems*, 34:9112–9124, 2021.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[10] Martin Engelcke, Adam R Kosiorek, Oiwi Parker Jones, and Ingmar Posner. Genesis: Generative scene inference and sampling with object-centric latent representations. *arXiv preprint arXiv:1907.13052*, 2019.

[11] François Fleuret, Ting Li, Charles Dubout, Emma K Wampler, Steven Yantis, and Donald Geman. Comparing machines and humans on a visual categorization test. *Proc. Natl. Acad. Sci. U. S. A.*, 108(43):17621–17625, October 2011.

[12] Anirudh Goyal, Alex Lamb, Jordan Hoffmann, Shagun Sodhani, Sergey Levine, Yoshua Bengio, and Bernhard Schölkopf. Recurrent independent mechanisms. *arXiv preprint arXiv:1909.10893*, 2019.

[13] Kaiming He, X Zhang, S Ren, and J Sun. Deep residual learning for image recognition." computer vision and pattern recognition (2015). *Google Scholar There is no corresponding record for this reference*, pages 770–778, 2015.

[14] Drew Hudson and Christopher D Manning. Learning by abstraction: The neural state machine. *Advances in Neural Information Processing Systems*, 32, 2019.

[15] Drew A Hudson and Christopher D Manning. Compositional attention networks for machine reasoning. *arXiv preprint arXiv:1803.03067*, 2018.

[16] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Judy Hoffman, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Inferring and executing programs for visual reasoning. In *Proceedings of the IEEE international conference on computer vision*, pages 2989–2998, 2017.

[17] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[18] Nanbo Li, Cian Eastwood, and Robert Fisher. Learning object-centric representations of multi-object scenes from multiple views. *Advances in Neural Information Processing Systems*, 33:5656–5666, 2020.

[19] Sarthak Mittal, Sharath Chandra Raparthy, Irina Rish, Yoshua Bengio, and Guillaume Lajoie. Compositional attention: Disentangling search and retrieval. *arXiv preprint arXiv:2110.09419*, 2021.

[20] Nasim Rahaman, Muhammad Waleed Gondal, Shruti Joshi, Peter Gehler, Yoshua Bengio, Francesco Locatello, and Bernhard Schölkopf. Dynamic inference with neural interpreters. *Advances in Neural Information Processing Systems*, 34:10985–10998, 2021.

[21] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[22] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. *Advances in neural information processing systems*, 30, 2017.

[23] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[24] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[25] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[26] Yuhuai Wu, Honghua Dong, Roger Grosse, and Jimmy Ba. The scattering compositional learner: Discovering objects, attributes, relationships in analogical reasoning. *arXiv preprint arXiv:2007.04212*, 2020.

[27] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

[28] Chi Zhang, Feng Gao, Baoxiong Jia, Yixin Zhu, and Song-Chun Zhu. Raven: A dataset for relational and analogical visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5317–5327, 2019.