V-PROM: A Benchmark for Visual Reasoning Using Visual Progressive Matrices

Damien Teney,[1*] **Peng Wang,**[2*] **Jiewei Cao,**[1] **Lingqiao Liu,**[1] **Chunhua Shen,**[1] **Anton van den Hengel**[1]
[1]Australian Institute for Machine Learning
The University of Adelaide
Adelaide, Australia
[2]University of Wollongong
Wollongong, Australia
[*]Authors with equal contribution, listed in alphabetical order

Abstract

Advances in machine learning have generated increasing enthusiasm for tasks that require high-level reasoning on top of perceptual capabilities, particularly over visual data. Such tasks include, for example, image captioning, visual question answering, and visual navigation. Their evaluation is however hindered by task-specific confounding factors and dataset biases. In parallel, the existing benchmarks for abstract reasoning are limited to synthetic stimuli (e.g. images of simple shapes) and do not capture the challenges of real-world data. We propose a new large-scale benchmark to evaluates abstract reasoning over real visual data. The test involves *visual questions* that require operations fundamental to many high-level vision tasks, such as comparisons of counts and logical operations on complex visual properties. The benchmark measures a method's ability to infer high-level relationships and to generalise them over image-based concepts. We provide multiple training/test splits that require controlled levels of generalization. We evaluate a range of deep learning architectures, and find that existing models, including those popular for vision-and-language tasks, are unable to solve seemingly-simple instances. Models using relational networks fare better but leave substantial room for improvement.

Introduction

Some of the most active research areas in computer vision are tackling increasingly complex tasks that require high-level reasoning. Some examples of this trend include visual question answering (VQA) (Antol et al. 2015), image captioning (Anderson et al. 2016), referring expressions (Yu et al. 2017), visual dialog (Das et al. 2017), and vision-and-language navigation (Anderson et al. 2018b). While deep learning helped make significant progress, these tasks expose the limitations of the pattern recognition methods that have proved successful on classical vision tasks such as object recognition. A key indicator of the shortcomings of deep learning methods is their tendency to respond to specific features or biases in the dataset, rather than generalising to an approach that is applicable more broadly (Agrawal et al. 2018; Devlin et al. 2015). In response, we propose a benchmark to *directly* measure a method's ability for high-level reasoning over real visual information, and in which we can control the level of generalisation required.

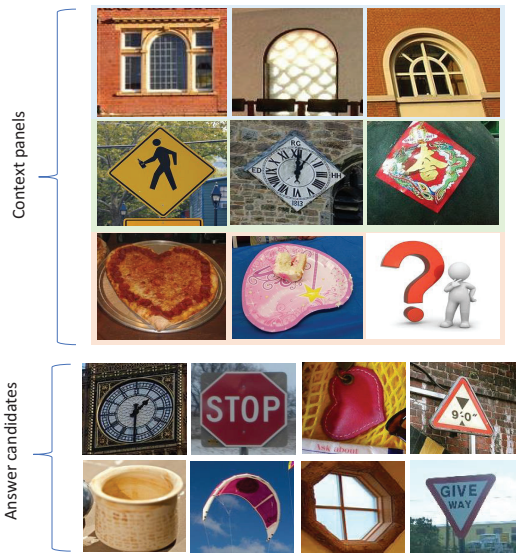

Figure 1: We propose a new task to evaluate a model's ability to perform abstract reasoning over complex visual stimuli. Each test instance is a matrix of 3×3 images, within which each row contains 3 images that exemplify the same relationship (in this case they have the same shape). The task is to identify the correct candidate for the missing $9^{\text{scriptsizeth}}$ image from a set of candidates. The correct answer above is the third candidate that represents a heart-shaped object.

Progress on the complex tasks mentioned above is typically evaluated on standardized benchmarks (Anderson et al. 2018b; Antol et al. 2015; Teney, Liu, and van den Hengel 2016). Methods are evaluated with metrics on task-specific objectives, *e.g.* predicting the correct answer in VQA, or producing a sentence matching the ground truth in image captioning. These tasks include a strong visual component,

and they are naturally assumed to lie on the path to semantic scene understanding, the overarching goal of computer vision. Unfortunately, non-visual aspects of these tasks – language in particular – act as major confounding factors. For example, in image captioning, the automated evaluation of generated language is itself an unsolved problem. In VQA, many questions are phrased such that their answers can be guessed without looking at the image.

We propose to take a step back with a task that directly evaluates abstract reasoning over realistic visual stimuli. Our setting is inspired by Raven's Progressive Matrices (RPMs) (Raven and for Educational Research. 1938), which are used in educational settings to measure human non-verbal visual reasoning abilities. Each instance of the task is a 3×3 matrix of images, where the last image is missing and is to be chosen from eight candidates. All rows of the completed matrix must represent a same relationship (logical relationships, counts and comparisons, etc.) over a visual property of their three images (Fig. 1). We use real photographs, such that the task requires strong visual capabilities, and we focus on visual, mostly non-semantic properties. This evaluation is thus designed to reflect the capabilities required by the complex tasks mentioned above, but in an abstract non-task-specific manner that might help guide general progress in the field.

Other recent efforts have proposed benchmarks for visual reasoning (Barrett et al. 2018; Suhr et al. 2017) and our key difference is to focus on real images, which are of greater interest to the computer vision community than 2D shapes and line drawings. This is a critical difference, because abstract reasoning is otherwise much easier to achieve when applied to a closed set of easily identified symbols such as simple geometrical shapes. A major contribution of this paper is the construction of a suitable dataset with real images on large scale (over 300,000 instances).

We have adapted and evaluated a range of deep learning models on our benchmark. Simple feed-forward networks achieve better than random results given enough depth, but recurrent neural networks and relational networks perform noticeably better. In the evaluation settings requiring strong generalization, *i.e.* applying relationships to visual properties in combinations not seen during training, all tested models clearly struggle. In most cases, small improvements are observed by using additional supervision, both on the visual features (using a bottom-up attention network (Anderson et al. 2018a) rather than a ResNet CNN), and on the type of relationship represented in the training examples. These results indicate the difficulty of the task while hinting at promising research directions.

Finally, **the proposed benchmark is not to be addressed as an end-goal, but should serve as a diagnostic test** of methods aiming at more complex tasks. In the spirit of the CLEVR dataset for VQA (Johnson et al. 2016) and the bAbI dataset for reading comprehension (Weston et al. 2015), our benchmark focuses on the fundamental operations common to multiple high-levels tasks. Crafting a solution specific to this benchmark is however not necessarily a path to actual solutions to these tasks. This guided the selection of general-purpose architectures evaluated in this paper.

The contributions of this paper are summarized as follows.

1. We define a new task to evaluate a model's ability for abstract reasoning over complex visual stimuli. The task is designed to require reasoning similar to complex tasks in computer vision, while allowing evaluation free of task-specific confounding factors such as natural language and dataset biases.
2. We describe a procedure to collect instances for this task at little cost, by mining images and annotations from the Visual Genome. We build a large-scale dataset of over 300,000 instances, over which we define multiple training and evaluation splits that require controlled amounts of generalization.
3. We evaluate a range of popular deep learning architectures on the benchmark. We identify elements that prove beneficial (*e.g.* relational reasoning and mid-level supervision), and we also show that all tested models struggle significantly when strong generalization is required.

The dataset will be publicly released to encourage the development of models with improved capabilities for abstract reasoning over visual data.

Related work

Evaluation of abstract visual reasoning Evaluating reasoning has a long history in the field of AI, but is typically based on pre-defined or easily identifiable symbols. Recent works include the task set of Fleuret *et al.* (Fleureta et al. 2011), in which they focus on the spatial arrangement of abstract elements in synthetic images. Their setting is reminiscent of the Bongard problems presented in (Bongard 1970) and further popularized by Hofstadter (Hofstadter 1979). Stabinger *et al.* (Stabinger, Rodríguez-Sánchez, and Piater 2016) tested whether state-of-the-art CNN architectures can compare visual properties of multiple abstract objects, *e.g.* to determine whether two shapes are of the same size. Although this involves high-level reasoning, it is over coarse characteristics of line-drawings.

V-PROM is inspired by Raven's Progressive Matrices (RPMs) (Raven and for Educational Research. 1938), a classic psychological test of a human's ability to interpret synthetic images. RPMs have been used previously to evaluate the reasoning abilities of neural networks (Barrett et al. 2018; Hoshen and Werman 2017; Wang and Su 2015). In (Hoshen and Werman 2017), the authors propose a CNN model to solve problems involving geometric operations such as rotations and reflections. Barrett *et al.* (Barrett et al. 2018) evaluated existing deep learning models on a large-scale dataset of RPMs, with a procedure similar to one previously proposed by Wang *et al.* (Wang and Su 2015). The benchmark of Barrett *et al.* (Barrett et al. 2018) is the most similar to our work. It uses synthetic images of simple 2D shapes, whereas ours uses much more complex images, at the cost of a less precise control of the visual stimuli. Recognizing the complementarity of the two settings, we purposefully model our evaluation setup after (Barrett et al. 2018) such that future methods can be evaluated and compared across the two settings. Since the synthetic images in (Barrett et al. 2018) do not reflect the complexity of real-world data, progress on this benchmark may not readily translate

Figure 2: Some challenging instances from our dataset. See the footnote[1] for the answer key.

to high-level vision tasks. Our work bridges the gap between these two extremes.

Evaluation of high-level tasks in computer vision The interest in high-level tasks is growing, as exemplified by the advent of VQA (Antol et al. 2015), referring expressions (Yu et al. 2017), and visual navigation (Anderson et al. 2018b), to name a few. Unbiased evaluations are notoriously difficult, and there is a growing trend toward evaluation on out-of-distribution data, *i.e.* where the test set is drawn from a different distribution than the training set (Agrawal et al. 2018; Anderson et al. 2018b; Teney and van den Hengel 2016; Tran et al. 2016). In this spirit, our benchmark includes multiple training/test splits drawn from different distributions to evaluate generalization under controlled conditions. Moreover, our task focuses on abstract relationships applied to visual (*i.e.* mostly non-semantic) properties, with the aim of minimizing the possibility of solving the task by exploiting non-visual factors.

Models for abstract reasoning with neural networks Various architectures have been proposed with the goal of moving beyond memorizing training examples, for example relation networks (Santoro et al. 2017), memory-augmented networks (Weston, Chopra, and Bordes 2014), and neural Turing machines (Graves, Wayne, and Danihelka 2014). Recent works on meta learning (*e.g.* (Finn, Abbeel, and Levine 2017)) address the same fundamental problem by focusing on generalization from few examples (*i.e.* few shot learning), and they have shown better generalization (Finn et al. 2017), including in VQA (Teney and van den Hengel 2018). Barrett *et al.* (Barrett et al. 2018) applied relation networks (RNs) with success to their dataset of RPMs. We evaluate RNs on our benchmark with equally encouraging results, although there remains large room for improvement, in particular when strong generalization is required.

[1] Denoting the candidate answers as 1–8, left-to-right, first then second row, the correct ones are 7, 2, 6.

A new task to evaluate visual reasoning

Our task is inspired by the classical Raven's Progressive Matrices (Raven and for Educational Research. 1938) used in human IQ tests (see Fig. 1) . Each instance is a matrix of 3×3 images, where the missing final image must be identified from among 8 candidates. The goal is to select an image such that all 3 rows represent a same relationship over some visual property (attribute, object category, or object count) of their 3 respective images. The definition of our task was guided by the following principles. First, it must require, but be not limited to, strong visual recognition ability. Second, it should measure a common set of capabilities required in high-level computer vision tasks. Third, it must be practical to construct a large-scale benchmark for this task, enabling an automatic and unambiguous evaluation. Finally, the task cannot be solvable through task-specific heuristics or relying on superficial statistics of the training examples. This points at a task that is compositional in nature and inherently requires strong generalization.

Our task can be seen as an extension to real images of recent benchmarks for reasoning on synthetic data (Barrett et al. 2018; Hoshen and Werman 2017). These works sacrifice visual realism for precise control over the contents of images which are limited to simple geometrical shapes. It is unclear whether reasoning under these conditions can transfer to realistic vision tasks. Our design is also intended to limit the extent to which semantic cues might be used to as "shortcuts" to avoid solving the task using the appropriate relationships. For example, a test to recognize the relation *above* could rely on the higher likelihood of *car above ground* than *ground above car*, rather than its actual spatial meaning. Therefore, our task focuses on fundamental visual properties and relationships such as logical and counting operations over *multiple* images (co-occurrence in a same photograph being likely biased).

The task requires identifying a plausible explanation for the provided triplets of images, *i.e.* a relation that could have generated them. The incomplete triplet serves as a "visual question", and the explanation must be applied generatively to identify the missing image. It is unavoidable that

	Object attributes	Human attributes	Object categories	Object counts
Nb. visual elements	84	38	346	10
Nb. images	36,750	12,249	82,905	11,730
Nb. task instances	45,000	45,000	45,000	100,000[2]

Table 1: Statistics of the V-PROM dataset.

more than one of the answer candidates constitute plausible completions. Indeed, a sufficiently-contrived explanation can justify any possible choice. The model has to identify the explanation with the strongest justification, which in practice tends to be the simplest one in the sense of Occam's razor. This is expected to be learned by the model from training examples.

Construction of the V-PROM dataset

We describe how to construct a large-scale dataset for our task semi-automatically. We call it *V-PROM* for *Visual PROgressive Matrices*.

Generating descriptions of task instances Each instance is a matrix of 3×3 images that we call a visual reasoning matrix (VRM). Each image I_i in the VRM depicts a visual element $a_i = \phi(I_i)$, where a_i denotes an element depicted in the image with $a_i \in \mathcal{A} \cup \mathcal{O} \cup \mathcal{C}$, where $\mathcal{A}$, $\mathcal{O}$, $\mathcal{C}$, respectively denote sets of possible attributes, objects, and object counts. We denote with $v(I_i) \in \{A, O, C\}$ the type of visual element a_i corresponds to. We also denote with $I_{i,j}$ the j-th image of the i-th row in a VRM. Each VRM represents one specific type v of visual elements, and one specific type of relationship $r \in \{\text{And}, \text{Or}, \text{Union}, \text{Progression}\}$. We define them as follows.

- *And*: $\phi(I_{i,3}) = \phi(I_{i,j}),\ \forall j \in 1, 2$. The last image of each row has the same visual element as the other two.
- *Or*: $\phi(I_{i,3}) = \phi(I_{i,1})$ or $\phi(I_{i,3}) = \phi(I_{i,2})$. The last image in each row has the same visual element as the first or the second.
- *Union*: $\{\phi(I_{1,j})\ \forall\ j\} = \{\phi(I_{2,j})\ \forall\ j\} = \{\phi(I_{3,j})\ \forall\ j\}$. All rows contain the same three visual elements, possibly in different orders.
- *Progression*: $v(I_{i,j}) = C,\ \forall i, j$; and $\phi(I_{i,t+1}) - \phi(I_{i,t}) = \phi(I_{j,t+1}) - \phi(I_{i,t})\ \forall i, j, t \in 1, 2$. The numbers of objects in a row follow an arithmetic progression.

We randomly sample a visual element v and relationship r to generate the definition of a VRM. Seven additional incorrect answer candidates are obtained by sampling seven different visual elements of the same type as v. The following section describes how to obtain images that fulfills a definition (v, r) of a VRM by mining annotations from the Visual Genome (VG) (Krishna et al. 2016).

Mining images from the Visual Genome To select suitable images, we impose five desired principles: *richness*, *purity*, *image quality*, *visual relatedness*, and *independence*. Richness requires the diversity of visual elements, and of the images representing each visual element. Purity constrains the complexity of the image, as we want images that depict the visual element of interest fairly clearly. Visual relatedness guides us toward properties that have a clear visual depiction. As a counterexample, the attribute *open* appears very differently when a door is open and a bottle is open. Such semantic attributes are not desirable for our task. Finally, independence excludes the objects that frequently co-occur with other objects (*e.g.* "sky","road","water", *etc.*) and could lead to ambiguous VRMs.

We obtain images that fulfill the above principles using VG's region-level annotations of categories, attributes, and natural language description (Table 1). We first preselect categories and attributes with large numbers of instances to guarantee sufficient representations of each in our dataset. We manually exclude unsuitable labels such as semantic attributes, and objects likely to cause ambiguity. We crop the annotated regions to obtain *pure* images. We discard those smaller than 100 px in either dimension. The annotations of *object counts* are extracted from numbers 1–10 appearing in natural language descriptions (*e.g.* "five bowls of oatmeal"), manually excluding those unrelated to counts (*e.g.* "five o'clock" or "a 10 years old boy").

Data splits to measure generalization

In order to evaluate a method's capabilities for generalization, we define several training/evaluation splits that require different levels of generalization. Training and evaluating a method in each of these settings will provide an overall picture of its capabilities beyond the basic fitting of training examples. To define these different settings, we follow the nomenclature proposed by Barrett *et al.* (Barrett et al. 2018).

1. **Neutral** – The training and test sets are both sampled from the whole set of relationships and visual elements. Training to testing ratio is $2:1$.
2. **Interpolation / extrapolation** – These two splits evaluate generalization for counting. In the interpolation split, odd counts (1,3,5,7,9) are used for training and even counts (2,4,6,8,10) are used for testing. In the extrapolation split, the first five counts (1–5) are used for training and the remaining (6–10) are used for testing.
3. **Held-out attributes** – The object attributes are divided into 7 super-attributes[3]: color, material, scene, plant condition, action, shape, texture. The human attributes are divided into 6 super-attributes: age, hair style, clothing style, gender, action, clothing color. The super-attributes *shape, texture, action* are held-out for testing only.
4. **Held-out objects** – A subset of object categories $(1/3)$ are held-out for testing only.
5. **Held-out pairs of relationships/attributes** – A subset of relationship/super-attribute combinations are held-out for testing only. Three combinations are held-out for both object attributes and human attributes. The held-out super-attributes vary with each type of relationship.

[2]We generate more task instances with *object counts* than with *attributes* and *categories* because counts are the only ones involved in the relationship *progression*, in addition to the three others (*and*, *or*, *union*).

[3]The attributes within each super-attribute are mutually exclusive.

6. **Held-out pairs of relationships/objects** – For each type of relationship, $1/3$ of objects are held-out. The held-out objects are different for each relationship.

We report a model's performance with the accuracy, *i.e.* the fraction of test instances for which the predicted answer (among the eight candidates) is correct. Random guessing gives an accuracy of 12.5%.

Task complexity and human evaluation

Solving an instance of our task requires to recognize the visual elements depicted in all images, and to identify the relation that applies to triplets of images. This basically amounts to inferring the abstract description (Section) $\mathcal{S} = \{[r, v] : r \in \mathcal{R}, v \in \mathcal{V}\}$ of the instance. Our dataset contains 4 types of relations, applied over 478 types of visual elements (Table 1), giving in the order of 2,000 different combinations.

We performed a human study to assess the difficulty of our benchmark. We presented human subjects with a random selection of task instances, sampled evenly across the four types of relations. The testees can skip an instance if they find it too difficult or ambiguous. The accuracy was of 77.8% with a *skip* rate of 4.5%. This accuracy is not an upper bound for the task however. The two main reasons for non-perfect human performance are (1) counting errors with >5 objects, cluttered background, or scale variations and (2) a tendency to use prior knowledge and favor higher-level (semantic) concepts/attributes than those used to generate the dataset.

Models and experimental setup

We evaluated a range of models on our benchmark. These models are based on popular deep learning architectures that have proven successful on various task-specific benchmarks. The models are summarized in Fig. 3.

Input data

For each instance of our task, the input data consists of 8 context panels and 8 candidate answers. These 16 RGB images are passed through a pretrained CNN to extract visual features. Our experiments compare features from a ResNet101 and from a Bottom-Up Attention Network[4] (Anderson et al. 2018a), which is popular for image captioning and VQA (Teney et al. 2018). The feature maps from either of these CNNs are average-pooled, and the resulting vector is L2-normalized. The vector of each of the 16 images is concatenated with a one-hot representation of an index: the 8 context panels are assigned indices 1–8 and the candidate answers 9–16. The resulting vectors are referred to as $x_1, x_2, ...x_{16} \in \mathbb{R}^{2048+16}$.

The vectors x_i serve as input to the models described below, which are trained with supervision to predict a score for each of the 8 candidate answers, *i.e.* $\hat{\boldsymbol{s}} \in \mathbb{R}^8$. Each model is trained with a softmax cross-entropy loss over $\hat{\boldsymbol{s}}$, standard backpropagation and SGD, using AdaDelta as the optimizer. Suitable hyperparameters for each model were coarsely selected by grid search (details in supplementary material). We held out 8,000 instances from the training set to serve as a validation set, to select the hyperparameters and to monitor for convergence and early-stopping. Unless noted, the non-linear transformations within the networks below refer to a linear layer followed by a ReLU.

[4]The network of *et al.* (Anderson et al. 2018a) was pretrained with annotations from the Visual Genome. Our dataset only uses cropped images from VG, and we use layer activations rather than explicit class predictions, but the possible overlap in the label space used to pretrain (Anderson et al. 2018a) and to generate our benchmark must be kept in mind.

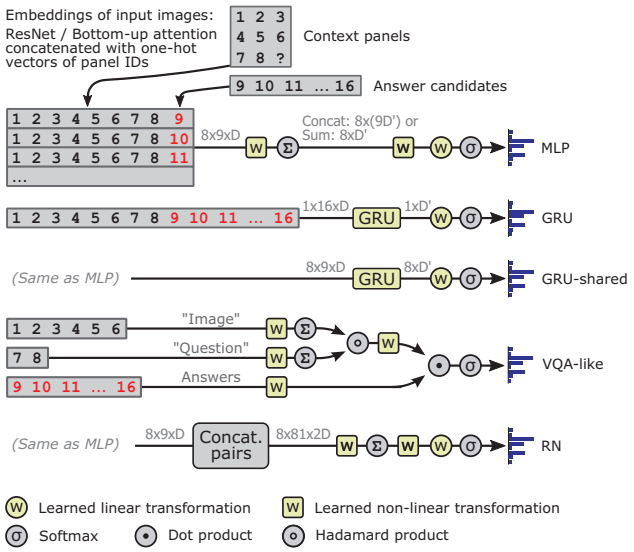

Figure 3: Overview of the models evaluated in our experiments. These are based on popular deep learning architectures.

MLP

Our simplest model is a multilayer perceptron (see Fig. 3). The features of every image are passed through a non-linear transformation $f_1(\cdot)$. The model is then applied so as to share the parameters used to score each candidate answer. The features of each candidate answer (x_i for i=9, …, 16) are concatenated with the context panels ($x_1, \ldots, x_8$). The features are then passed through another non-linear transformation $f_2(\cdot)$, and a final linear transformation $\boldsymbol{w}$ to produce a scalar score for each candidate answer. That is, $\forall\, i = 1 \ldots 8$:

$$\hat{s}_i \;=\; \boldsymbol{w}\; f_2\big([f_1(x_1); f_1(x_2); \ldots; f_1(x_8); f_1(x_{8+i})]\big) \quad (1)$$

where the semicolumn represents the concatenation of vectors. A variant of this model replaces the concatenation with a sum-pooling over the nine panels. This reduces the number of parameters by sharing the weights within f_2 across the panels. This gives

$$\hat{s}_i \;=\; \boldsymbol{w}\; f_2\big(\; \Sigma_{i=1,2,\ldots,x_8,8+i}\; f_1(x_i)\;\big)\;. \quad (2)$$

We will refer to these two models as *MLP-cat-k* and *MLP-sum-k*, in which f_1 and f_2 are both implemented with $k/2$ linear layers, all followed by a ReLU.

GRU

We consider two variants of a recurrent neural network, implemented with a gated recurrent unit (GRU (Cho et al. 2014)). The first naive version takes each of the feature vectors x_1 to x_16 over 16 time steps. The final hidden state of the GRU is then passed through a linear transformation $\boldsymbol{w}$ to map it to a vector of 8 scores $\hat{\boldsymbol{s}} \in \mathbb{R}^8$.

$$\hat{\boldsymbol{s}} \;=\; \boldsymbol{w}\ \text{GRU}\big(x_1, x_2, ..., x_8, x_9, x_{10}, ..., x_{16}\big)\ . \quad (3)$$

The second version shares the parameters of the model over the 8 candidate answers. The GRU takes, in parallel, 8 sequences, each consisting of the context panels with one of the 8 candidate answers. The final state of each GRU is then mapped to a single score for the corresponding candidate answer. That is, $\forall\ i = 1...8$:

$$\hat{s}_i \;=\; \boldsymbol{w}\ \text{GRU}\big(x_1, x_2, ..., x_8, x_{8+i}\big)\ . \quad (4)$$

VQA-like architecture

We consider an architecture that mimics a state-of-the-art model in VQA (Teney et al. 2018) based on a "joint embedding" approach (Wu et al. 2017; Teney, Wu, and van den Hengel 2017). In our case, the context panels $x_1, ..., x_6$ serve as the input "image", and the panels x_7, x_8 serve as the "question". They are passed through non-linear transformations, then combined with an elementwise product into a joint embedding $\boldsymbol{h}$. The score for each answer is obtained as the dot product between $\boldsymbol{h}$ and the embedding of each candidate answer (see Fig. 3). Formally, we have

$$\boldsymbol{h} \;=\; \Sigma_{i=1...6}\ f_1(x_i) \;\circ\; \Sigma_{i=7,8}\ f_2(x_i) \quad (5)$$

$$\hat{s}_i \;=\; \boldsymbol{h}.f_3(x_{8+i})\ . \quad (6)$$

where f_1, f_2 and f_3 are non-linear transformations, and $\circ$ represents the Hadamard product.

Relation networks

We finally evaluate a relation network (RN). RNs were specifically proposed to model relationships between visual elements, such as in VQA when questions refer to multiple parts of the image (Santoro et al. 2017). Our model is applied, again, such that its parameters are shared across answer candidates. The basic idea of an RN is to consider all pairwise combinations of input elements (9^2 in our case), pass them through a non-linear transformation, sum-pool over these 9^2 representations, then pass the pooled representation through another non-linear transformation. Formally, we have, $\forall\ i = 1...8$:

$$\boldsymbol{h}_i \;=\; \Sigma_{(i,j)\in\{1,2,...,8,8+i\}} f_1\big([x_i; x_j]\big) \quad (7)$$

$$\hat{s}_i \;=\; \boldsymbol{w}\ f_2(\boldsymbol{h}_i) \quad (8)$$

where f_1 and f_2 are non-linear transformations, and $\boldsymbol{w}$ a linear transformation.

Auxiliary objective

We experimented with an auxiliary objective that encourages the network to predict the type of the relationship involved in the given matrix. This objective is trained with a softmax cross-entropy and the ground truth type of relationship in the training example. This value is a index among the seven possible relations, *i.e. and*, *or*, *progression*, *attribute*, *object*, *union*, and *counting* (see Section). This prediction is made from a linear projection of the final activations of the network in Eq. 8, that is:

$$\hat{s}_i \;=\; \boldsymbol{w}'\ f_2(\boldsymbol{h}) \quad (9)$$

where $\boldsymbol{w}'$ is an additional learned linear transformation. At test time, this prediction is not used, and the auxiliary objective serves only to provide an inductive bias during the training of the network such that its internal representation captures the type of relationship (which should then help the model to generalize). Note that we also experimented with an auxiliary objective for predicting labels such as object class and visual attributes, but this did not prove beneficial.

	ResNet	ResNet +aux.loss	B.-up	B.-up +aux.loss
Human evaluation		77.8		
RN with shuffled inputs	12.5	12.5	12.5	12.5
MLP-sum-6 layers	40.7	44.5	50.4	55.7
GRU-shared	43.4	48.2	46.7	52.7
VQA-like	36.7	39.7	37.9	41.0
Relational network (RN)	**51.2**	**55.8**	**55.4**	**61.3**

Table 2: Summary of the best models in the neutral setting, on all question types (also see the supplementary material).

Experiments

We conducted numerous experiments to establish reference baselines and to shed light on the capabilities of popular architectures. As a sanity check, we trained our best model with randomly-shuffled context panels. This verified that the task could not be solved by exploiting superficial regularities of the data. All models trained in this way perform around the "chance" level of 12.5%.

Neutral training/test splits

We first examine all models on the neutral training/test splits (see supplementary material for full results). In this setting, training and test data are drawn from the same distribution, and supervised models are expected to perform well, given sufficient capacity and training examples. We observe that a **simple MLP** can indeed fit the data relatively well if it has enough layers, but a network with only 2 non-linear layers performs quite badly. The two models based on a **GRU** have very different performance. The *GRU-shared* model performs best. It shares its parameters over the candidate answers (processed in parallel rather than across the recurrent steps). This result was not obviously predictable, since this model does not get to consider all candidate answers in relation with each other. The alternate model (*GRU*) receives every candidate answer in succession. It could therefore perform additional reasoning steps over the candidates, but this does not seem to be the case in practice. The **VQA-like** model obtains a performance comparable to a deep MLP, but it proved more difficult to train than an MLP. In some of our experiments, the optimization this model was slow or

simply failed to converge. We found it best to use, as non-linear transformations, "gated tanh" layers as in (Teney et al. 2018). Overall, we obtained the best performance with a **relation network** (RN) model. While this is basically an MLP on top of pairwise combinations of features, these combinations prove much more informative than the individual features. We experimented with an RN without the one-hot representations of panel IDs concatenated with the input ("*RN without panel IDs*"), and this version performed very poorly. It is worth noting that RNs come at the cost of processing N^2 feature vectors rather than N (with N=9 in our case). The number of parameters is the same, since they are shared across the N^2 combinations, but the computation time increases.

We break down performance along two axes (see figure in the supplementary material). The following two groups of question types are mutually exclusive: and/or/progression/union, and attribute/object/counting. The former reflects the type of relationship across the nine images of a test instance, while the latter corresponds to the type of visual properties to which the relationship applies. We observe that some types are much more easily solved than others. Instances involving object identity are easier than those involving attributes and counts, presumable because the image features are obtained with a CNN pretrained for object classification. The bottom-up image features performs remarkably well, most likely because the set of labels used for pretraining was richer than the ImageNet labels used to train the ResNet. The instances that require counting are particularly difficult; this corroborates the struggle of vision systems with counting, already reported *e.g.* in (Kafle and Kanan 2017).

Splits requiring generalization

We now look at the performance with respect to splits that specifically require generalization (see supplementary material for full results). As expected, accuracy drops significantly as the need for generalization increases. This confirms our hypothesis that naive end-to-end training cannot guarantee generalization beyond training examples, and that this is easily masked when the test and training data come from the same distribution (as in the neutral split). This drop is particularly visible with the simple MLP and GRU models. The RN model suffers a smaller drop in performance in some of the generalization settings. This indicates that learning over combinations of features provides a useful inductive bias for our task.

Image features from bottom-up attention We tested all models with features from a ResNet, as well as features from the "bottom-up attention" model of Anderson *et al*. (Anderson et al. 2018a). These improve the performance of all tested models over ResNet features, in the neutral and all generalization splits. The bottom-up attention model is pretrained with a richer set of annotations than the ImageNet labels used to pretrain the ResNet. This likely provides features that better capture fine visual properties of the input images. Note that the visual features used by our models do not contain explicit predictions of such labels and visual properties, as they are vectors of continuous values. We experimented with alternative schemes (not reported in the plots), including an auxiliary loss within our models for predicting visual attributes, but these did not prove helpful.

Auxiliary prediction of relationship type We experimented with success with an auxiliary loss on the prediction of the type of relationship in the given instance. This is provided during training as a label among seven. All models trained with this additional loss gained in accuracy in the neutral and most generalization settings. The relative importance of the main and auxiliary losses did not seem critical, and all reported experiments use an equal weight on both.

Overall, the performance of our best models remains well below that of human performance leaving substantial room for improvement. This dataset should be a valuable tool to evaluate future approaches to visual reasoning.

Conclusions

We have introduced a new benchmark to measure a method's ability to carry out abstract reasoning over complex visual data. The task addresses a central issue in deep learning, being the degree to which methods learn to reason over their inputs. This issue is critical because reasoning can generalise to new classes of data, whereas memorising incidental relationships between signal and label does not. This issue lies at the core of many of the current challenges in deep learning, including zero-shot learning, domain adaptation, and generalisation, more broadly.

Our benchmark serves to evaluate capabilities similar to some of those required in high-level tasks in computer vision, without task-specific confounding factors such as natural language or dataset biases. Moreover, the benchmark includes multiple evaluation settings that demand controllable levels of generalization. Our experiments with popular deep learning models demonstrate that they struggle when strong generalization is required, in particular for applying known relationships to combinations of visual properties not seen during training. We identified a number of promising directions for future research, and we hope that this setting will encourage the development of models with improved capabilities for abstract reasoning over visual data.

References

Agrawal, A.; Batra, D.; Parikh, D.; and Kembhavi, A. 2018. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*

Anderson, P.; Fernando, B.; Johnson, M.; and Gould, S. 2016. Guided open vocabulary image captioning with constrained beam search. *arXiv preprint arXiv:1612.00576*.

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018a. Bottom-up and top-down attention for image captioning and vqa. *CVPR*.

Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sünderhauf, N.; Reid, I.; Gould, S.; and van den Hengel,

A. 2018b. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Zitnick, C. L.; and Parikh, D. 2015. VQA: Visual Question Answering. In *Proc. IEEE Int. Conf. Comp. Vis.*

Barrett, D.; Hill, F.; Santoro, A.; Morcos, A.; and Lillicrap, T. 2018. Measuring abstract reasoning in neural networks. In *Proc. Int. Conf. Mach. Learn.*

Bongard, M. M. 1970. *Pattern recognition*. Spartan Books.

Cho, K.; van Merrienboer, B.; Gulcehre, C.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proc. Conf. Empirical Methods in Natural Language Processing*.

Das, A.; Kottur, S.; Gupta, K.; Singh, A.; Yadav, D.; Moura, J. M.; Parikh, D.; and Batra, D. 2017. Visual Dialog. In *CVPR*.

Devlin, J.; Gupta, S.; Girshick, R. B.; Mitchell, M.; and Zitnick, C. L. 2015. Exploring nearest neighbor approaches for image captioning. *arXiv preprint arXiv:1505.04467*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*.

Finn, C.; Yu, T.; Zhang, T.; Abbeel, P.; and Levine, S. 2017. One-shot visual imitation learning via meta-learning. In *Conference on Robot Learning (CoRL)*, 357–368.

Fleureta, F.; Lic, T.; Dubouta, C.; Wamplerd, E. K.; Yantisd, S.; and Gemanc, D. 2011. Comparing machines and humans on a visual categorization test. In *Proceedings of the National Academy of Sciences*.

Graves, A.; Wayne, G.; and Danihelka, I. 2014. Neural turing machines. *arXiv preprint arXiv:1410.5401*.

Hofstadter, D. R. 1979. *Godel, Escher, Bach: An Eternal Golden Braid*. Basic Books, Inc.

Hoshen, D., and Werman, M. 2017. Iq of neural networks. *arXiv preprint arXiv:1710.01692*.

Johnson, J.; Hariharan, B.; van der Maaten, L.; Fei-Fei, L.; Zitnick, C. L.; and Girshick, R. B. 2016. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. *arXiv preprint arXiv:1612.06890*.

Kafle, K., and Kanan, C. 2017. An analysis of visual question answering algorithms. In *Proc. IEEE Int. Conf. Comp. Vis.*

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; Bernstein, M.; and Fei-Fei, L. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *arXiv preprint arXiv:1602.07332*.

Raven, J. C., and for Educational Research., A. C. 1938. *Raven's progressive matrices (1938) : sets A, B, C, D, E*. Australian Council for Educational Research Melbourne.

Santoro, A.; Raposo, D.; Barrett, D. G.; Malinowski, M.; Pascanu, R.; Battaglia, P.; and Lillicrap, T. 2017. A simple neural network module for relational reasoning. In *Proc. Advances in Neural Inf. Process. Syst.*

Stabinger, S.; Rodríguez-Sánchez, A.; and Piater, J. 2016. 25 years of cnns: Can we compare to human abstraction capabilities? In Villa, A. E.; Masulli, P.; and Pons Rivero, A. J., eds., *Artificial Neural Networks and Machine Learning*.

Suhr, A.; Lewis, M.; Yeh, J.; and Artzi, Y. 2017. A corpus of natural language for visual reasoning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, 217–223.

Teney, D., and van den Hengel, A. 2016. Zero-shot visual question answering. *CoRR* abs/1611.05546.

Teney, D., and van den Hengel, A. 2018. Visual question answering as a meta learning task. In *Proc. Eur. Conf. Comp. Vis.*

Teney, D.; Anderson, P.; He, X.; and van den Hengel, A. 2018. Tips and tricks for visual question answering: Learnings from the 2017 challenge. *CVPR*.

Teney, D.; Liu, L.; and van den Hengel, A. 2016. Graph-structured representations for visual question answering. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*

Teney, D.; Wu, Q.; and van den Hengel, A. 2017. Visual question answering: A tutorial. *IEEE Signal Processing Magazine* 34:63–75.

Tran, K.; He, X.; Zhang, L.; Sun, J.; Carapcea, C.; Thrasher, C.; Buehler, C.; and Sienkiewicz, C. 2016. Rich image captioning in the wild. *arXiv preprint arXiv:1603.09016*.

Wang, K., and Su, Z. 2015. Automatic generation of raven's progressive matrices. In *Proc. Int. Joint Conf. Artificial Intell.*

Weston, J.; Bordes, A.; Chopra, S.; and Mikolov, T. 2015. Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*.

Weston, J.; Chopra, S.; and Bordes, A. 2014. Memory networks. *arXiv preprint arXiv:1410.3916*.

Wu, Q.; Teney, D.; Wang, P.; Shen, C.; Dick, A.; and van den Hengel, A. 2017. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*.

Yu, L.; Tan, H.; Bansal, M.; and Berg, T. L. 2017. A joint speaker-listener-reinforcer model for referring expressions. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*