Abstract Spatial-Temporal Reasoning via Probabilistic Abduction and Execution

Chi Zhang* Baoxiong Jia* Song-Chun Zhu Yixin Zhu
UCLA Center for Vision, Cognition, Learning, and Autonomy
{chi.zhang,baoxiongjia}@ucla.edu, sczhu@stat.ucla.edu, yixin.zhu@ucla.edu

Abstract

Spatial-temporal reasoning is a challenging task in Artificial Intelligence (AI) due to its demanding but unique nature: a theoretic requirement on representing *and* reasoning *based on spatial-temporal knowledge in mind, and an applied requirement on a high-level cognitive system capable of* navigating *and* acting *in space and time. Recent works have focused on an abstract reasoning task of this kind—Raven's Progressive Matrices (RPM). Despite the encouraging progress on RPM that achieves human-level performance in terms of accuracy, modern approaches have neither a treatment of human-like reasoning on generalization, nor a potential to generate answers. To fill in this gap, we propose a neuro-symbolic* ***Probabilistic Abduction and Execution (PrAE)*** *learner; central to the PrAE learner is the process of probabilistic abduction and execution on a probabilistic scene representation, akin to the mental manipulation of objects. Specifically, we disentangle perception and reasoning from a monolithic model. The neural visual perception frontend predicts objects' attributes, later aggregated by a scene inference engine to produce a probabilistic scene representation. In the symbolic logical reasoning backend, the PrAE learner uses the representation to* ***abduce*** *the hidden rules. An answer is predicted by* ***executing*** *the rules on the probabilistic representation. The entire system is trained end-to-end in an analysis-by-synthesis manner* ***without*** *any visual attribute annotations. Extensive experiments demonstrate that the PrAE learner improves cross-configuration generalization and is capable of rendering an answer, in contrast to prior works that merely make a categorical choice from candidates.*

1. Introduction

While "thinking in pictures" [13], *i.e.*, spatial-temporal reasoning, is effortless and instantaneous for humans, this significant ability has proven to be particularly challenging for current machine vision systems [27]. With the promising results [13] that show the very ability is strongly correlated with one's logical induction performance and a crucial factor for the intellectual history of technology development, recent computational studies on the problem focus on an abstract reasoning task relying heavily on "thinking in pictures"—Raven's Progressive Matrices (RPM) [3, 24, 51, 52]. In this task, a subject is asked to pick a correct answer that best fits an incomplete figure matrix to satisfy the hidden governing rules. The ability to solve RPM-like problems is believed to be critical for generating and conceptualizing solutions to multi-step problems, which requires mental manipulation of given images over a time-ordered sequence of spatial transformations. Such a task is also believed to be characteristic of relational and analogical reasoning and an indicator of one's *fluid intelligence* [6, 18, 26, 55].

State-of-the-art algorithms incorporating a contrasting mechanism and perceptual inference [17, 72] have achieved decent performance in terms of accuracy. Nevertheless, along with the improved accuracy from deep models come critiques on its transparency, interpretability, generalization, and difficulty to incorporate knowledge. Without explicitly distinguishing perception and reasoning, existing methods use a *monolithic* model to learn correlation, sacrificing transparency and interpretability in exchange for improved performance [17, 20, 53, 59, 70, 72, 75]. Furthermore, as shown in experiments, deep models nearly always overfit to the training regime and cannot properly generalize. Such a finding is consistent with Fodor [11] and Marcus's [43, 44] hypothesis that human-level systematic generalizability is hardly compatible with classic neural networks; Marcus postulates that a neuro-symbolic architecture should be recruited for human-level generalization [7, 8, 9, 41, 42, 66].

Another defect of prior methods is the lack of top-down and bottom-up reasoning [72]: Human reasoning applies a *generative* process to abduce rules and execute them to synthesize a possible solution in mind, and *discriminatively* selects the most similar answer from choices [19]. This bi-directional reasoning is in stark contrast to discriminative-only models, solely capable of making a categorical choice.

* indicates equal contribution.

Psychologists also call for weak attribute supervision in RPM. As isolated Amazonians, absent of schooling on primitive attributes, could still correctly solve RPM [5, 25], an ideal computational counterpart should be able to learn it *absent of visual attribute annotations*. This weakly-supervised setting introduces unique challenges: How to jointly learn these visual attributes given only ground-truth images? With uncertainties in perception, how to abduce hidden logic relations from it? How about executing the symbolic logic on inaccurate perception to derive answers?

To support cross-configuration generalization and answer generation, we move a step further towards a neuro-symbolic model with explicit logical reasoning and human-like generative problem-solving while addressing the challenges. Specifically, we propose the *Probabilistic Abduction and Execution (PrAE)* learner; central to it is the process of abduction and execution on the probabilistic scene representation. Inspired by Fodor, Marcus, and neuro-symbolic reasoning [15, 40, 67, 68], the PrAE learner disentangles the previous monolithic process into two separate modules: a neural visual perception frontend and a symbolic logical reasoning backend. The neural visual frontend operates on object-based representation [15, 29, 40, 67, 68] and predicts conditional probability distributions on its attributes. A scene inference engine then aggregates all object attribute distributions to produce a probabilistic scene representation for the backend. The symbolic logical backend abduces, from the representation, hidden rules that govern the time-ordered sequence via inverse dynamics. An execution engine executes the rules to *generate* an answer representation in a probabilistic planning manner [12, 21, 31], instead of directly making a categorical choice among the candidates. The final choice is selected based on the divergence between the generated prediction and the given candidates. The entire system is trained end-to-end with a cross-entropy loss and a curricular auxiliary loss [53, 70, 72] *without* any visual attribute annotations. Fig. 1 compares the proposed PrAE learner with prior methods.

The unique design in PrAE connects perception and reasoning and offers several advantages: (i) With an intermediate probabilistic scene representation, the neural visual perception frontend and the symbolic logical reasoning backend can be *swapped* for different task domains, enabling a greater extent of module reuse and combinatorial *generalization*. (ii) Instead of blending perception and reasoning into one monolithic model without any explicit reasoning, probabilistic abduction offers a more *interpretable* account for reasoning on a logical representation. It also affords a more detailed analysis into both perception and reasoning. (iii) Probabilistic execution permits a *generative* process to be integrated into the system. Symbolic logical constraints can be transformed by the execution engine into a forward model [28] and applied in a probabilistic manner to predict the final scene representation, such that the entire system can be trained by analysis-by-synthesis [4, 14, 16, 22, 23, 36, 62, 63, 64, 65, 69, 77]. (iv) Instead of making a deterministic decision or drawing limited samples, maintaining probabilistic distributions brings in extra robustness and fault tolerance and allows gradients to be easily propagated.

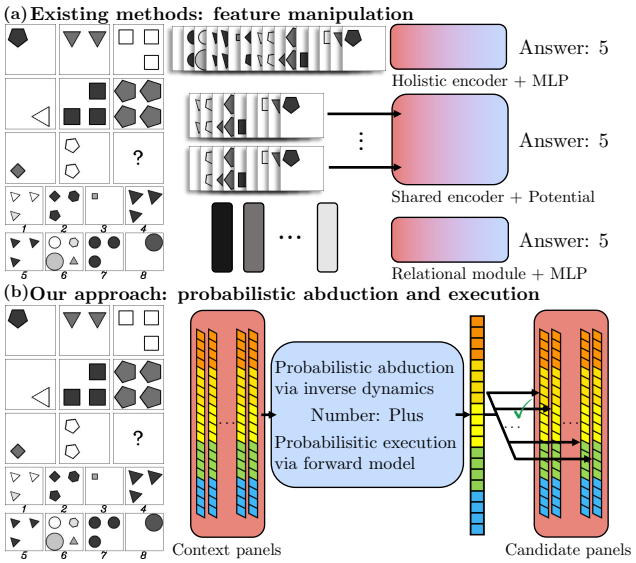

Figure 1. Differences between (a) prior methods and (b) the proposed approach. Prior methods do not explicitly distinguish perception and reasoning; instead, they use a monolithic model and only differ in how features are manipulated, lacking semantics and probabilistic interpretability. In contrast, the proposed approach disentangles this monolithic process: It perceives each panel of RPM as a set of probability distributions of attributes, performs logical reasoning to abduce the hidden rules that govern the time-ordered sequence, and executes the abduced rules to *generate* answer representations. A final choice is made based on the divergence between predicted answer distributions and each candidate's distributions; see Section 2 for a detailed comparison.

This paper makes three major contributions: (i) We propose the *Probabilistic Abduction and Execution (PrAE)* learner. Unlike previous methods, the PrAE learner disentangles perception and reasoning from a monolithic model with the reasoning process realized by abduction and execution on a probabilistic scene representation. The abduction process performs interpretable reasoning on perception results. The execution process adds to the learner a generative flavor, such that the system can be trained in an analysis-by-synthesis manner without any visual attribute annotations. (ii) Our experiments demonstrate the PrAE learner achieves better generalization results compared to existing methods in the cross-configuration generalization task of RPM. We also show that the PrAE learner is capable of generating answers for RPM questions via a renderer. (iii) We present analyses into the inner functioning of both perception and reasoning, providing an interpretable account of PrAE.

2. Related Work

Neuro-Symbolic Visual Reasoning Neuro-symbolic methods have shown promising potential in tasks involving an interplay between vision and language and vision and causality. Qi *et al*. [49, 50] showed that action recognition could be significantly improved with the help of grammar parsing, and Li *et al*. [33] integrated perception, parsing, and logics into a unified framework. Of particular relevance, Yi *et al*. [68] first demonstrated a prototype of a neuro-symbolic system to solve Visual Question Answering (VQA) [1], where the vision system and the language parsing system were separately trained with a final symbolic logic system applying the parsed program to deliver an answer. Mao *et al*. [40] improved such a system by making the symbolic component continuous and end-to-end trainable, despite sacrificing the semantics and interpretability of logics. Han *et al*. [15] built on [40] and studied the meta-concept problem by learning concept embeddings. A recent work investigated temporal and causal relations in collision events [67] and solved it in a way similar to [68]. The proposed PrAE learner is similar to but has fundamental differences from existing neuro-symbolic methods. Unlike the method proposed by Yi *et al*. [67, 68], our approach is end-to-end trainable and does not require intermediate visual annotations, such as ground-truth attributes. Compared to [40], our approach preserves logic semantics and interpretability by explicit logical reasoning involving probabilistic abduction and execution in a probabilistic planning manner [12, 21, 31].

Computational Approaches to RPM Initially proposed as an intelligence quotient test into general intelligence and fluid intelligence [51, 52], Raven's Progressive Matrices (RPM) has received notable attention from the research community of cognitive science. Psychologists have proposed reasoning systems based on symbolic representations and discrete logics [3, 37, 38, 39]. However, such logical systems cannot handle visual uncertainty arising from imperfect perception. Similar issues also pose challenges to methods based on image similarity [35, 45, 46, 47, 54]. Recent works approach this problem in a data-driven manner. The first automatic RPM generation method was proposed by Wang and Su [60]. Santoro *et al*. [53] extended it using procedural generation and introduced the Wild Relational Network (WReN) to solve the problem. Zhang *et al*. [70] and Hu *et al*. [20] used stochastic image grammar [76] and provided structural annotations to the dataset. Unanimously, existing methods do not explicitly distinguish perception and reasoning; instead, they use one monolithic neural model, sacrificing interpretability in exchange for better performance. The differences in previous methods lie in how features are manipulated: Santoro *et al*. [53] used the relational module to extract final features, Zhang *et al*. [70] stacked all panels into the channel dimension and fed them into a residual network, Hill *et al*. [17] prepared the data in a contrasting manner, Zhang *et al*. [72] composed the context with each candidate and compared their potentials, Wang *et al*. [59] modeled the features by a multiplex graph, and Hu *et al*. [20] integrated hierarchical features. Zheng *et al*. [75] studied a teacher-student setting in RPM, while Steenbrugge *et al*. [57] focused on a generative approach to improve learning. Concurrent to our work, Spratley *et al*. [56] unsupervisedly extracted object embeddings and conducted reasoning via a ResNet. In contrast, PrAE is designed to address cross-configuration generalization and disentangles perception and reasoning from a monolithic model, with symbolic logical reasoning implemented as probabilistic abduction and execution.

3. The PrAE Learner

Problem Setup In this section, we explain our approach to tackling the RPM problem. Each RPM instance consists of 16 panels: 8 context panels form an incomplete 3×3 matrix with a 9th missing entry, and 8 candidate panels for one to choose. The goal is to pick one candidate that best completes the matrix to satisfy the latent governing rules. Existing datasets [20, 53, 60, 70] assume fixed sets of object attributes, panel attributes, and rules, with each panel attribute governed by one rule. The value of a panel attribute constrains the value of the corresponding object attribute for each object in it.

Overview The proposed neuro-symbolic PrAE learner disentangles previous monolithic visual reasoning into two modules: the neural visual perception frontend and the symbolic logical reasoning backend. The frontend uses a CNN to extract object attribute distributions, later aggregated by a scene inference engine to produce panel attribute distributions. The set of all panel attribute distributions in a panel is referred to as its *probabilistic scene representation*. The backend retrieves this compact scene representation and performs logical abduction and execution in order to predict the answer representation in a generative manner. A final choice is made based on the divergence between the prediction and each candidate. Using REINFORCE [61], the entire system is trained *without attribute annotations* in a curricular manner; see Fig. 2 for an overview of PrAE.

3.1. Neural Visual Perception

The neural visual perception frontend operates on each of the 16 panels *independently* to produce probabilistic scene representation. It has two sub-modules: object CNN and scene inference engine.

Object CNN Given an image panel I, a sliding window traverses its spatial domain and feeds each image region into a 4-branch CNN. The 4 CNN branches use the same LeNet-like architecture [32] and produce the probability distributions of object attributes, including objectiveness

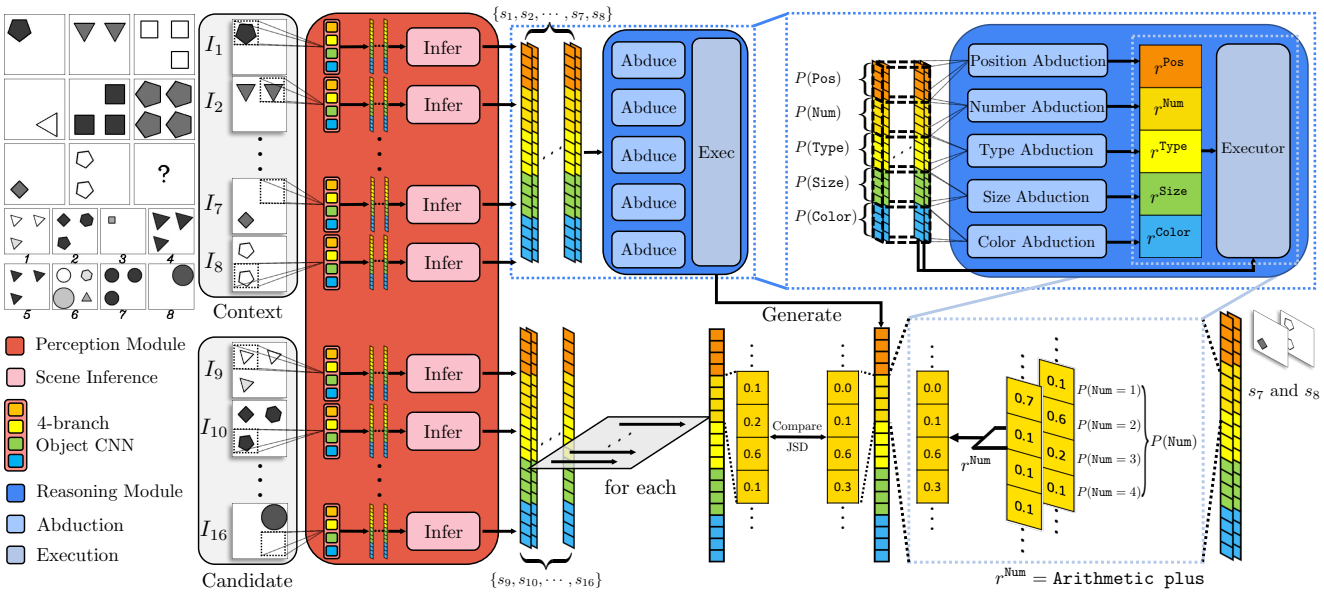

Figure 2. An overview of learning and reasoning of the proposed PrAE learner. Given an RPM instance, the neural perception frontend (in red) extracts probabilistic scene representation for each of the 16 panels (8 contexts + 8 candidates). The *Object CNN* sub-module takes in each image region returned by a sliding window to produce object attribute distributions (over objectiveness, type, size, and color). The *Scene Inference Engine* sub-module (in pink) aggregates object attribute distributions from all regions to produce panel attribute distributions (over position, number, type, size, and color). Probabilistic representation for context panels is fed into the symbolic reasoning backend (in blue), which abduces hidden rule distributions for all panel attributes (upper-right figure) and executes chosen rules on corresponding context panels to generate the answer representation (lower-right figure). The answer representation is compared with each candidate representation from the perception frontend; the candidate with minimum divergence from the prediction is chosen as the final answer. The lower-right figure is an example of probabilistic execution on the panel attribute of `Number`; see Section 3.2 for the exact computation process.

(whether the image region has an object), type, size, and color. Of note, the distributions of type, size, and color are conditioned on objectiveness being true. Attribute distributions of each image region are kept and sent to the scene inference engine to produce panel attribute distributions.

Scene Inference Engine The scene inference engine takes in the outputs of object CNN and produces panel attribute distributions (over position, number, type, size, and color) by marginalizing over the set of object attribute distributions (over objectiveness, type, size, and color). Take the panel attribute of `Number` as an example: Given N objectiveness probability distributions produced by the object CNN for N image regions, the probability of a panel having k objects can be computed as

$$P(\texttt{Number} = k) = \sum_{\substack{B^o \in \{0,1\}^N \\ |B^o| = k}} \prod_{j=1}^{N} P(b_j^o = B_j^o), \quad (1)$$

where B^o is an ordered binary sequence corresponding to objectiveness of the N regions, $|\cdot|$ the number of 1 in the sequence, and $P(b_j^o)$ the objectiveness distribution of the jth region. We assume $k \geqslant 1$ in each RPM panel, leave $P(\texttt{Number} = 0)$ out, and renormalize the probability to have a sum of 1. The panel attribute distributions for position, type, size, and color, can be computed similarly.

We refer to the set of all panel attribute distributions in a panel its *probabilistic scene representation*, denoted as s, with the distribution of panel attribute a denoted as $P(s^a)$.

3.2. Symbolic Logical Reasoning

The symbolic logical reasoning backend collects probabilistic scene representation from 8 context panels, abduces the probability distributions over hidden rules on each panel attribute, and executes them on corresponding panels of the context. Based on a prior study [3], we assume a set of symbolic logical constraints describing rules is available. For example, the `Arithmetic plus` rule on `Number` can be represented as: for each row (column), $\forall l, m \geqslant 1$

$$(\texttt{Number}_1 = m) \wedge (\texttt{Number}_2 = l) \wedge (\texttt{Number}_3 = m + l), \quad (2)$$

where $\texttt{Number}_i$ denotes the number of objects in the ith panel in a row (column). With access to such constraints, we use inverse dynamics to abduce the rules in an instance. They can also be transformed into a forward model and executed on discrete symbols: For instance, `Arithmetic plus` deterministically adds `Number` in the first two panels to obtain the `Number` of the last panel.

Probabilistic Abduction Given the probabilistic scene representation of 8 context panels, the probabilistic abduction engine calculates the probability of rules for each panel attribute via inverse dynamics. Formally, for each rule r on a panel attribute a,

$$P(r^a \mid I_1, \ldots, I_8) = P(r^a \mid I_1^a, \ldots, I_8^a), \tag{3}$$

where I_i denotes the ith context panel, and I_i^a the component of context panel I_i corresponding to a. Note Eq. (3) generalizes inverse dynamics [28] to 8 states, in contrast to that of a conventional MDP.

To model $P(r^a \mid I_1^a, \ldots, I_8^a)$, we leverage the compact probabilistic scene representation with respect to attribute a and logical constraints:

$$P(r^a \mid I_1^a, \ldots, I_8^a) \propto \sum_{S^a \in \texttt{valid}(r^a)} \prod_{i=1}^{8} P(s_i^a = S_i^a), \tag{4}$$

where $\texttt{valid}(\cdot)$ returns a set of attribute value assignments of the context panels that satisfy the logical constraints of r^a, and i indexes into context panels. By going over all panel attributes, we have the distribution of hidden rules for each of them.

Take `Arithmetic plus` on `Number` as an example. A row-major assignment for context panels can be $[1, 2, 3, 1, 3, 4, 1, 2]$ (as in Fig. 2), whose probability is computed as the product of each panel having k objects as in Eq. (1). Summing it with other assignment probabilities gives an unnormalized rule probability.

We note that the set of valid states for each r^a is a product space of valid states on each row (column). Therefore, we can perform partial marginalization on each row (column) first and aggregate them later to avoid directly marginalizing over the entire space. This decomposition will help reduce computation and mitigate numerical instability.

Probabilistic Execution For each panel attribute a, the probabilistic execution engine chooses a rule from the abduced rule distribution and executes it on corresponding context panels to predict, in a generative fashion, the panel attribute distribution of an answer. While traditionally, a logical forward model only works on discrete symbols, we follow a generalized notion of probabilistic execution as done in probabilistic planning [21, 31]. The probabilistic execution could be treated as a distribution transformation that redistributes the probability mass based on logical rules. For a binary rule r on a,

$$P(s_3^a = S_3^a) \propto \sum_{\substack{(S_2^a, S_1^a) \in \texttt{pre}(r^a) \\ S_3^a = f(S_2^a, S_1^a; r^a)}} P(s_2^a = S_2^a) P(s_1^a = S_1^a), \tag{5}$$

where f is the forward model transformed from logical constraints and $\texttt{pre}(\cdot)$ the rule precondition set. Predicted distributions of panel attributes compose the final probabilistic scene representation s_f.

As an example of `Arithmetic plus` on `Number`, 4 objects result from the addition of $(1, 3)$, $(2, 2)$, and $(3, 1)$. The probability of an answer having 4 objects is the sum of the instances' probabilities.

During training, the execution engine samples a rule from the abduced probability. During testing, the most probable rule is chosen.

Candidate Selection With a set of predicted panel attribute distributions, we compare it with that from each candidate answer. We use the Jensen–Shannon Divergence (JSD) [34] to quantify the divergence between the prediction and the candidate, *i.e.*,

$$d(s_f, s_i) = \sum_a \mathbb{D}_{\text{JSD}}(P(s_f^a) \mid\mid P(s_i^a)), \tag{6}$$

where the summation is over panel attributes and i indexes into the candidate panels. The candidate with minimum divergence will be chosen as the final answer.

Discussion The design of reasoning as probabilistic abduction and execution is a computational and interpretable counterpart to human-like reasoning in RPM [3]. By abduction, one infers the hidden rules from context panels. By executing the abduced rules, one obtains a probabilistic answer representation. Such a probabilistic representation is compared with all candidates available; the most similar one in terms of divergence is picked as the final answer. Note that the probabilistic execution adds the generative flavor into reasoning: Eq. (5) depicts the predicted panel attribute distribution, which can be sampled and sent to a rendering engine for panel generation. The entire process resembles bi-directional inference and combines both top-down and bottom-up reasoning missing in prior works. In the meantime, the design addresses challenges mentioned in Section 1 by marginalizing over perception and abducing and executing rules probabilistically.

3.3. Learning Objective

During training, we transform the divergence in Eq. (6) into a probability distribution by

$$P(\text{Answer} = i) \propto \exp(-d(s_f, s_i)) \tag{7}$$

and minimize the cross-entropy loss. Note that the learning procedure follows a general paradigm of analysis-by-synthesis [4, 14, 16, 22, 23, 36, 62, 63, 64, 65, 69, 77]: The learner synthesizes a result and measures difference analytically.

As the reasoning process involves rule selection, we use REINFORCE [61] to optimize:

$$\min_\theta \mathbb{E}_{P(r)}[\ell(P(\text{Answer}; r), y)], \tag{8}$$

where θ denotes the trainable parameters in the object CNN, $P(r)$ packs the rule distributions over all panel attributes,

ℓ is the cross-entropy loss, and y is the ground-truth answer. Note that here we make explicit the dependency of the answer distribution on rules, as the predicted probabilistic scene representation s_f is dependent on the rules chosen.

In practice, the PrAE learner experiences difficulty in convergence with cross-entropy loss only, as the object CNN fails to produce meaningful object attribute predictions at the early stage of training. To resolve this issue, we jointly train the PrAE learner to optimize the auxiliary loss, as discussed in recent literature [53, 70, 72]. The auxiliary loss regularizes the perception module such that the learner produces the correct rule prediction. The final objective is

$$\min_{\theta} \mathbb{E}_{P(r)}[\ell(P(\text{Answer}; r), y)] + \sum_{a} \lambda^a \ell(P(r^a), y^a), \quad (9)$$

where λ^a is the weight coefficient, $P(r^a)$ the distribution of the abduced rule on a, and y^a the ground-truth rule. In reinforcement learning terminology, one can treat the cross-entropy loss as the negative reward and the auxiliary loss as behavior cloning [58].

3.4. Curriculum Learning

In preliminary experiments, we notice that accurate objectiveness prediction at the early stage is essential to the success of the learner, while learning without auxiliary will reinforce the perception system to produce more accurate object attribute predictions in the later stage when all branches of the object CNN are already warm-started. This observation is consistent with human learning: One learns object attributes only after they can correctly distinguish objects from the scene, and their perception will be enhanced with positive signals from the task.

Based on this observation, we train our PrAE learner in a 3-stage curriculum [2]. In the first stage, only parameters corresponding to objectiveness are trained. In the second stage, objectiveness parameters are frozen while weights responsible for type, size, and color prediction are learned. In the third stage, we perform joint fine-tuning for the entire model via REINFORCE [61].

4. Experiments

We demonstrate the efficacy of the proposed PrAE learner in RPM. In particular, we show that the PrAE learner achieves the best performance among all baselines in the cross-configuration generalization task of RPM. In addition, the modularized perception and reasoning process allows us to probe into how each module performs in the RPM task and analyze the PrAE learner's strengths and weaknesses. Furthermore, we show that probabilistic scene representation learned by the PrAE learner can be used to generate an answer when equipped with a rendering engine.

4.1. Experimental Setup

We evaluate the proposed PrAE learner on RAVEN [70] and I-RAVEN [20]. Both datasets consist of 7 distinct RPM configurations, each of which contains $10,000$ samples, equally divided into 6 folds for training, 2 folds for validation, and 2 folds for testing. We compare our PrAE learner with simple baselines of LSTM, CNN, and ResNet, and strong baselines of WReN [53], ResNet+DRT [70], LEN [75], CoPINet [72], MXGNet [59], and SRAN [20]. To measure cross-configuration generalization, we train all models using the 2x2Grid configuration due to its proper complexity for probability marginalization and a sufficient number of rules on each panel attribute. We test the models on all *other* configurations. All models are implemented in PyTorch [48] and optimized using ADAM [30] on an Nvidia Titan Xp GPU. For numerical stability, we use log probability in PrAE.

4.2. Cross-Configuration Generalization

Table 1 shows the cross-configuration generalization performance of different models. While advanced models like WReN, LEN, MXGNet, and SRAN have fairly good fitting performance on the training regime, these models fail to learn transferable representation for other configurations, which suggests that they do not learn logics or any forms of abstraction but visual appearance only. Simpler baselines like LSTM, CNNs, ResNet, and ResNet+DRT show less severe overfitting, but neither do they demonstrate satisfactory performance. This effect indicates that using only deep models in abstract visual reasoning makes it very difficult to acquire the generalization capability required in situations with similar inner mechanisms but distinctive appearances. By leveraging the notion of contrast, CoPINet improves generalization performance by a notable margin.

Equipped with symbolic reasoning and neural perception, not only does the PrAE learner achieve the best performance among all models, but it also shows performance better than humans on three configurations. Compared to baselines trained on the full dataset (see supplementary material), the PrAE learner surpasses all other models on the 2x2Grid domain, despite other models seeing 6 times more data. The PrAE learner does not exhibit strong overfitting either, achieving comparable and sometimes better performance on Center, L-R, and U-D. However, limitations of the PrAE learner do exist. In cases with overlap (O-IC and O-IG), the performance decreases, and a devastating result is observed on 3x3Grid. The first failure is due to the domain shift in the region appearance that neural models cannot handle, and the second could be attributed to marginalization over probability distributions of multiple objects in 3x3Grid, where uncertainties from all objects accumulate, leading to inaccurate abduced rule distributions. These observations are echoed in our analysis shown next.

Method	Acc	Center	2x2Grid	3x3Grid	L-R	U-D	O-IC	O-IG
WReN	9.86/14.87	8.65/14.25	29.60/20.50	9.75/15.70	4.40/13.75	5.00/13.50	5.70/14.15	5.90/12.25
LSTM	12.81/12.52	12.70/12.55	13.80/13.50	12.90/11.35	12.40/14.30	12.10/11.35	12.45/11.55	13.30/13.05
LEN	12.29/13.60	11.85/14.85	41.40/18.20	12.95/13.35	3.95/12.55	3.95/12.75	5.55/11.15	6.35/12.35
CNN	14.78/12.69	13.80/11.30	18.25/14.60	14.55/11.95	13.35/13.00	15.40/13.30	14.35/11.80	13.75/12.85
MXGNet	20.78/13.07	12.95/13.65	37.05/13.95	24.80/12.50	17.45/12.50	16.80/12.05	18.05/12.95	18.35/13.90
ResNet	24.79/13.19	24.30/14.50	25.05/14.30	25.80/12.95	23.80/12.35	27.40/13.55	25.05/13.40	22.15/11.30
ResNet+DRT	31.56/13.26	31.65/13.20	39.55/14.30	35.55/13.25	25.65/12.15	32.05/13.10	31.40/13.70	25.05/13.15
SRAN	15.56/29.06	18.35/37.55	38.80/38.30	17.40/29.30	9.45/29.55	11.35/28.65	5.50/21.15	8.05/18.95
CoPINet	52.96/22.84	49.45/24.50	61.55/31.10	**52.15**/25.35	68.10/20.60	65.40/19.85	39.55/19.00	34.55/19.45
PrAE Learner	**65.03/77.02**	**76.50/90.45**	**78.60/85.35**	28.55/**45.60**	**90.05/96.25**	**90.85/97.35**	**48.05/63.45**	**42.60/60.70**
Human	84.41	95.45	81.82	79.55	86.36	81.81	86.36	81.81

Table 1. Model performance (%) on RAVEN / I-RAVEN. All models are trained on 2x2Grid only. Acc denotes the mean accuracy. Following Zhang *et al*. [70], L-R is short for the Left-Right configuration, U-D Up-Down, O-IC Out-InCenter, and O-IG Out-InGrid.

Object Attribute	Acc	Center	2x2Grid	3x3Grid	L-R	U-D	O-IC	O-IG
Objectiveness	93.81/95.41	96.13/96.07	99.79/99.99	99.71/97.98	99.56/95.00	99.86/94.84	71.73/88.05	82.07/95.97
Type	86.29/89.24	89.89/89.33	99.95/95.93	83.49/85.96	99.92/92.90	99.85/97.84	91.55/91.86	66.68/70.85
Size	64.72/66.63	68.45/69.11	71.26/73.20	71.42/62.02	73.00/85.08	73.41/73.45	53.54/62.63	44.36/40.95
Color	75.26/79.45	75.15/75.65	85.15/87.81	62.69/69.94	85.27/83.24	84.45/81.38	84.91/75.32	78.48/82.84

Table 2. Accuracy (%) of the object CNN on each attribute, reported as RAVEN / I-RAVEN. The CNN module is trained with the PrAE learner on 2x2Grid only without any visual attribute annotations. Acc denotes the mean accuracy on each attribute.

Panel Attribute	Acc	Center	2x2Grid	3x3Grid	L-R	U-D	O-IC	O-IG
Pos/Num	90.53/91.67	-	90.55/90.05	92.80/94.10	-	-	-	88.25/90.85
Type	94.17/92.15	100.00/95.00	99.75/95.30	63.95/68.40	100.00/99.90	100.00/100.00	100.00/100.00	86.08/77.60
Size	90.06/88.33	98.95/99.00	90.45/89.90	65.30/70.45	98.15/96.78	99.45/92.45	93.08/96.13	77.35/70.78
Color	87.38/87.25	97.60/93.75	88.10/85.35	37.45/45.65	98.90/92.38	99.40/98.43	92.90/97.23	73.75/79.48

Table 3. Accuracy (%) of the probabilistic abduction engine on each attribute, reported as RAVEN / I-RAVEN. The PrAE learner is trained on 2x2Grid only. Acc denotes the mean accuracy on each attribute.

4.3. Analysis on Perception and Reasoning

RAVEN and I-RAVEN provide multiple levels of annotations for us to analyze our modularized PrAE learner. Specifically, we use the region-based attribute annotations to evaluate our object CNN in perception. Note that the object CNN is not trained using any attribute annotations. We also use the ground-truth rule annotations to evaluate the accuracy of the probabilistic abduction engine.

Table 2 details the analysis of perception using the object CNN: It achieves reasonable performance on object attribute prediction, though not trained with any visual attribute annotations. The model shows a relatively accurate prediction of objectiveness in order to solve an RPM instance. Compared to the size prediction accuracy, the object CNN is better at predicting texture-related attributes of type and color. The object CNN has similar results on 2x2Grid, L-R, and U-D. However, referencing Table 1, we notice that 2x2Grid requires marginalization over more objects, resulting in an inferior performance. Accuracy further drops on configurations with overlap, leading to unsatisfactory results on O-IC and O-IG. For 3x3Grid, more accurate predictions are necessary as uncertainties accumulate from probabilities over multiple objects.

Table 3 details the analysis on reasoning, showing how the probabilistic abduction engine performs on rule prediction for each attribute across different configurations. Since rules on position and number are exclusive, we merge their performance as Pos/Num. As Center, L-R, U-D, and O-IC do not involve rules on Pos/Num, we do not measure the abduction performance on them. We note that, in general, the abduction engine shows good performance on all panel attributes, with a perfect prediction on type in certain configurations. However, the design of abduction as probability marginalization is a double-edged sword. While the object CNN's performance on size prediction is only marginally different on 2x2Grid and 3x3Grid in RAVEN, their abduction accuracies drastically vary. The difference occurs because uncertainties on object attributes accumulate during marginalization as the number of objects increases, eventually leading to poor performance on rule prediction and answer selection. However, on configurations with fewer objects, unsatisfactory object attribute predictions can still produce accurate rule predictions. Note there is no guarantee that a correct rule will necessarily lead to a correct final choice, as the selected rule still operates on panel attribute distributions inferred from object attribute distributions.

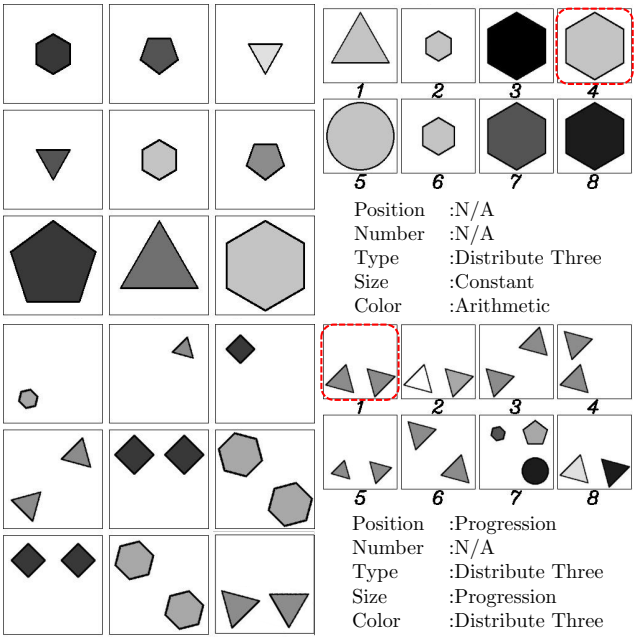

Figure 3. Two RPM instances with the final 9th panels filled by our generation results. The ground-truth selections are highlighted in red squares, and the ground-truth rules in each instance are listed. There are no rules on position and number in the first instance of the Center configuration, and the rules on position and number are exclusive in the second instance of 2x2Grid.

4.4. Generation Ability

One unique property of the proposed PrAE learner is its ability to directly generate a panel from the predicted representation when a rendering engine is given. The ability resembles the bi-directional top-down and bottom-up reasoning, adding a generative flavor commonly ignored in prior discriminative-only approaches [17, 20, 53, 59, 70, 72, 75]. As the PrAE learner predicts final panel attribute distributions and is trained in an analysis-by-synthesis manner, we can sample panel attribute values from the predicted distributions and render the final answer using a rendering engine. Here, we use the rendering program released with RAVEN [70] to show the generation ability of the PrAE learner. Fig. 3 shows examples of the generation results. Note that one of our generations is slightly different from the ground-truth answer due to random sampling of rotations during rendering. However, it still follows the rules in the problem and should be considered as a correct answer.

5. Conclusion and Discussion

We propose the *Probabilistic Abduction and Execution (PrAE)* learner for spatial-temporal reasoning in Raven's Progressive Matrices (RPM) that decomposes the problem-solving process into neural perception and logical reasoning. While existing methods on RPM are merely discriminative, the proposed PrAE learner is a hybrid of generative models and discriminative models, closing the loop in a human-like, top-down bottom-up bi-directional reasoning process. In the experiments, we show that the PrAE learner achieves the best performance on the cross-configuration generalization task on RAVEN and I-RAVEN. The modularized design of the PrAE learner also permits us to probe into how perception and reasoning work independently during problem-solving. Finally, we show the unique generative property of the PrAE learner by filling in the missing panel with an image produced by the values sampled from the probabilistic scene representation.

However, the proposed PrAE learner also has limits. As shown in our experiments, probabilistic abduction can be a double-edged sword in the sense that when the number of objects increases, uncertainties over multiple objects will accumulate, making the entire process sensitive to perception performance. Also, complete probability marginalization introduces a challenge for computational scalability; it prevents us from training the PrAE learner on more complex configurations such as 3x3Grid. One possible solution might be a discrete abduction process. However, jointly learning such a system is non-trivial. It is also difficult for the learner to perceive and reason based on lower-level primitives, such as lines and corners. While, in theory, a generic detector of lines and corners should be able to resolve this issue, no well-performing systems exist in practice, except those with strict handcrafted detection rules, which would miss the critical probabilistic interpretations in the entire framework. The PrAE learner also requires strong prior knowledge about the underlying logical relations to work, while an ideal method should be able to induce the hidden rules by itself. Though a precise induction mechanism is still unknown for humans, an emerging computational technique of bi-level optimization [10, 73] may be able to house perception and induction together into a general optimization framework.

While we answer questions about generalization and generation in RPM, one crucial question remains to be addressed: How perception learned from other domains can be transferred and used to solve this abstract reasoning task. Unlike humans that arguably apply knowledge learned from elsewhere to solve RPM, current systems still need training on the same task to acquire the capability. While feature transfer is still challenging for computer vision, we anticipate that progress in answering transferability in RPM will help address similar questions [71, 74, 78] and further advance the field.

Acknowledgement: The authors thank Sirui Xie, Prof. Ying Nian Wu, and Prof. Hongjing Lu at UCLA for helpful discussions. The work reported herein was supported by ONR MURI grant N00014-16-1-2007, DARPA XAI grant N66001-17-2-4029, and ONR grant N00014-19-1-2153.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 3

[2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2009. 6

[3] Patricia A Carpenter, Marcel A Just, and Peter Shell. What one intelligence test measures: a theoretical account of the processing in the raven progressive matrices test. *Psychological Review*, 97(3):404, 1990. 1, 3, 4, 5

[4] Yixin Chen, Siyuan Huang, Tao Yuan, Yixin Zhu, Siyuan Qi, and Song-Chun Zhu. Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2019. 2, 5

[5] Stanislas Dehaene, Véronique Izard, Pierre Pica, and Elizabeth Spelke. Core knowledge of geometry in an amazonian indigene group. *Science*, 311(5759):381–384, 2006. 2

[6] R E Snow, Patrick Kyllonen, and B Marshalek. The topography of ability and learning correlations. *Advances in the psychology of human intelligence*, pages 47–103, 1984. 1

[7] Mark Edmonds, Feng Kubricht, James, Colin Summers, Yixin Zhu, Brandon Rothrock, Song-Chun Zhu, and Hongjing Lu. Human causal transfer: Challenges for deep reinforcement learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society (CogSci)*, 2018. 1

[8] Mark Edmonds, Xiaojian Ma, Siyuan Qi, Yixin Zhu, Hongjing Lu, and Song-Chun Zhu. Theory-based causal transfer: Integrating instance-level induction and abstract-level structure learning. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2020. 1

[9] Mark Edmonds, Siyuan Qi, Yixin Zhu, James Kubricht, Song-Chun Zhu, and Hongjing Lu. Decomposing human causal learning: Bottom-up associative learning and top-down schema reasoning. In *Proceedings of the Annual Meeting of the Cognitive Science Society (CogSci)*, 2019. 1

[10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of International Conference on Machine Learning (ICML)*, 2017. 8

[11] Jerry A Fodor, Zenon W Pylyshyn, et al. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71, 1988. 1

[12] Malik Ghallab, Dana Nau, and Paolo Traverso. *Automated Planning: theory and practice*. Elsevier, 2004. 2, 3

[13] Temple Grandin. *Thinking in pictures: And other reports from my life with autism*. Vintage, 2006. 1

[14] Ulf Grenander. Lectures in pattern theory i, ii and iii: Pattern analysis, pattern synthesis and regular structures, 1976. 2, 5

[15] Chi Han, Jiayuan Mao, Chuang Gan, Josh Tenenbaum, and Jiajun Wu. Visual concept-metaconcept learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 2, 3

[16] Tian Han, Erik Nijkamp, Xiaolin Fang, Mitch Hill, Song-Chun Zhu, and Ying Nian Wu. Divergence triangle for joint training of generator model, energy-based model, and inferential model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 5

[17] Felix Hill, Adam Santoro, David GT Barrett, Ari S Morcos, and Timothy Lillicrap. Learning to make analogies by contrasting abstract relational structure. In *International Conference on Learning Representations (ICLR)*, 2019. 1, 3, 8

[18] Douglas R Hofstadter. *Fluid concepts and creative analogies: Computer models of the fundamental mechanisms of thought*. Basic books, 1995. 1

[19] Keith James Holyoak and Robert G Morrison. *The Oxford handbook of thinking and reasoning*. Oxford University Press, 2012. 1

[20] Sheng Hu, Yuqing Ma, Xianglong Liu, Yanlu Wei, and Shihao Bai. Stratified rule-aware network for abstract visual reasoning. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2021. 1, 3, 6, 8

[21] De-An Huang, Danfei Xu, Yuke Zhu, Animesh Garg, Silvio Savarese, Li Fei-Fei, and Juan Carlos Niebles. Continuous relaxation of symbolic planner for one-shot imitation learning. In *Proceedings of International Conference on Intelligent Robots and Systems (IROS)*, 2019. 2, 3, 5

[22] Siyuan Huang, Siyuan Qi, Yinxue Xiao, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Cooperative holistic scene understanding: Unifying 3d object, layout and camera pose estimation. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 2, 5

[23] Siyuan Huang, Siyuan Qi, Yixin Zhu, Yinxue Xiao, Yuanlu Xu, and Song-Chun Zhu. Holistic 3d scene parsing and reconstruction from a single rgb image. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2018. 2, 5

[24] Earl Hunt. *Quote the Raven? Nevermore.* Lawrence Erlbaum, 1974. 1

[25] Véronique Izard, Pierre Pica, Elizabeth S Spelke, and Stanislas Dehaene. Flexible intuitions of euclidean geometry in an amazonian indigene group. *Proceedings of the National Academy of Sciences (PNAS)*, 108(24):9782–9787, 2011. 2

[26] Susanne M Jaeggi, Martin Buschkuehl, John Jonides, and Walter J Perrig. Improving fluid intelligence with training on working memory. *Proceedings of the National Academy of Sciences (PNAS)*, 105(19):6829–6833, 2008. 1

[27] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[28] Michael I Jordan and David E Rumelhart. Forward models: Supervised learning with a distal teacher. *Cognitive Science*, 16(3):307–354, 1992. 2, 5

[29] Ken Kansky, Tom Silver, David A Mély, Mohamed Eldawy, Miguel Lázaro-Gredilla, Xinghua Lou, Nimrod Dorfman, Szymon Sidor, Scott Phoenix, and Dileep George. Schema networks: Zero-shot transfer with a generative causal model

of intuitive physics. In *Proceedings of International Conference on Machine Learning (ICML)*, 2017. 2

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2014. 6

[31] George Konidaris, Leslie Kaelbling, and Tomas Lozano-Perez. Symbol acquisition for probabilistic high-level planning. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 2015. 2, 3, 5

[32] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 3

[33] Qing Li, Siyuan Huang, Yining Hong, Yixin Chen, Ying Nian Wu, and Song-Chun Zhu. Closed loop neural-symbolic learning via integrating neural perception, grammar parsing, and symbolic reasoning. In *Proceedings of International Conference on Machine Learning (ICML)*, 2020. 3

[34] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991. 5

[35] Daniel R Little, Stephan Lewandowsky, and Thomas L Griffiths. A bayesian model of rule induction in raven's progressive matrices. In *Proceedings of the Annual Meeting of the Cognitive Science Society (CogSci)*, 2012. 3

[36] Matthew M Loper and Michael J Black. Opendr: An approximate differentiable renderer. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014. 2, 5

[37] Andrew Lovett and Kenneth Forbus. Modeling visual problem solving as analogical reasoning. *Psychological Review*, 124(1):60, 2017. 3

[38] Andrew Lovett, Kenneth Forbus, and Jeffrey Usher. A structure-mapping model of raven's progressive matrices. In *Proceedings of the Annual Meeting of the Cognitive Science Society (CogSci)*, 2010. 3

[39] Andrew Lovett, Emmett Tomai, Kenneth Forbus, and Jeffrey Usher. Solving geometric analogy problems through two-stage analogical mapping. *Cognitive Science*, 33(7):1192–1231, 2009. 3

[40] Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B Tenenbaum, and Jiajun Wu. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. In *International Conference on Learning Representations (ICLR)*, 2019. 2, 3

[41] Gary Marcus and Ernest Davis. *Rebooting AI: building artificial intelligence we can trust*. Pantheon, 2019. 1

[42] Gary Marcus and Ernest Davis. Insights for ai from the human mind. *Communications of the ACM*, 64(1):38–41, 2020. 1

[43] Gary F Marcus. Rethinking eliminative connectionism. *Cognitive psychology*, 37(3):243–282, 1998. 1

[44] Gary F Marcus. *The algebraic mind: Integrating connectionism and cognitive science*. MIT press, 2018. 1

[45] Keith McGreggor and Ashok Goel. Confident reasoning on raven's progressive matrices tests. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2014. 3

[46] Keith McGreggor, Maithilee Kunda, and Ashok Goel. Fractals and ravens. *Artificial Intelligence*, 215:1–23, 2014. 3

[47] Can Serif Mekik, Ron Sun, and David Yun Dai. Similarity-based reasoning, raven's matrices, and general intelligence. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 2018. 3

[48] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*, 2017. 6

[49] Siyuan Qi, Baoxiong Jia, Siyuan Huang, Ping Wei, and Song-Chun Zhu. A generalized earley parser for human activity parsing and prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 3

[50] Siyuan Qi, Baoxiong Jia, and Song-Chun Zhu. Generalized earley parser: Bridging symbolic grammars and sequence data for future prediction. In *Proceedings of International Conference on Machine Learning (ICML)*, 2018. 3

[51] James C Raven. Mental tests used in genetic studies: The performance of related individuals on tests mainly educative and mainly reproductive. Master's thesis, University of London, 1936. 1, 3

[52] John C Raven and John Hugh Court. *Raven's progressive matrices and vocabulary scales*. Oxford pyschologists Press, 1998. 1, 3

[53] Adam Santoro, Felix Hill, David Barrett, Ari Morcos, and Timothy Lillicrap. Measuring abstract reasoning in neural networks. In *Proceedings of International Conference on Machine Learning (ICML)*, 2018. 1, 2, 3, 6, 8

[54] Snejana Shegheva and Ashok Goel. The structural affinity method for solving the raven's progressive matrices test for intelligence. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2018. 3

[55] Charles Spearman. *The abilities of man*. Macmillan, 1927. 1

[56] Steven Spratley, Krista Ehinger, and Tim Miller. A closer look at generalisation in raven. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020. 3

[57] Xander Steenbrugge, Sam Leroux, Tim Verbelen, and Bart Dhoedt. Improving generalization for abstract reasoning tasks using disentangled feature representations. *arXiv preprint arXiv:1811.04784*, 2018. 3

[58] Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*. MIT press Cambridge, 1998. 6

[59] Duo Wang, Mateja Jamnik, and Pietro Lio. Abstract diagrammatic reasoning with multiplex graph networks. In *International Conference on Learning Representations (ICLR)*, 2020. 1, 3, 6, 8

[60] Ke Wang and Zhendong Su. Automatic generation of raven's progressive matrices. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, 2015. 3

[61] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992. 3, 5, 6

[62] Jiajun Wu, Joshua B Tenenbaum, and Pushmeet Kohli. Neural scene de-rendering. In *Proceedings of the IEEE Confer-*

ence on Computer Vision and Pattern Recognition (CVPR), 2017. 2, 5

[63] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. Marrnet: 3d shape reconstruction via 2.5 d sketches. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 2, 5

[64] Jianwen Xie, Yang Lu, Song-Chun Zhu, and Yingnian Wu. A theory of generative convnet. In *Proceedings of International Conference on Machine Learning (ICML)*, 2016. 2, 5

[65] Jianwen Xie, Song-Chun Zhu, and Ying Nian Wu. Learning energy-based spatial-temporal generative convnets for dynamic patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019. 2, 5

[66] Sirui Xie, Xiaojian Ma, Peiyu Yu, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Halma: Humanlike abstraction learning meets affordance in rapid problem solving. *arXiv preprint arXiv:2102.11344*, 2021. 1

[67] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua Tenenbaum. Clevrer: Collision events for video representation and reasoning. In *International Conference on Learning Representations (ICLR)*, 2020. 2, 3

[68] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 2, 3

[69] Alan Yuille and Daniel Kersten. Vision as bayesian inference: analysis by synthesis? *Trends in cognitive sciences*, 2006. 2, 5

[70] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 3, 6, 7, 8

[71] Chi Zhang, Baoxiong Jia, Mark Edmonds, Song-Chun Zhu, and Yixin Zhu. Acre: Abstract causal reasoning beyond covariation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 8

[72] Chi Zhang, Baoxiong Jia, Feng Gao, Yixin Zhu, Hongjing Lu, and Song-Chun Zhu. Learning perceptual inference by contrasting. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1, 2, 3, 6, 8

[73] Chi Zhang, Yixin Zhu, and Song-Chun Zhu. Metastyle: Three-way trade-off among speed, flexibility, and quality in neural style transfer. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2019. 8

[74] Wenhe Zhang, Chi Zhang, Yixin Zhu, and Song-Chun Zhu. Machine number sense: A dataset of visual arithmetic problems for abstract and relational reasoning. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2020. 8

[75] Kecheng Zheng, Zheng-Jun Zha, and Wei Wei. Abstract reasoning with distracting features. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1, 3, 6, 8

[76] Song-Chun Zhu and David Mumford. A stochastic grammar of images. *Foundations and Trends® in Computer Graphics and Vision*, 2(4):259–362, 2007. 3

[77] Song-Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision (IJCV)*, 27(2):107–126, 1998. 2, 5

[78] Yixin Zhu, Tao Gao, Lifeng Fan, Siyuan Huang, Mark Edmonds, Hangxin Liu, Feng Gao, Chi Zhang, Siyuan Qi, Ying Nian Wu, et al. Dark, beyond deep: A paradigm shift to cognitive ai with humanlike common sense. *Engineering*, 6(3):310–345, 2020. 8