Cognitive Neuroscience
Current Debates, Research & Reports

ISSN: 1758-8928 (Print) 1758-8936 (Online) Journal homepage: www.tandfonline.com/journals/pcns20

A feature-based generalizable prediction model for both perceptual and abstract reasoning

Quan Do, Thomas M. Morin, Chantal E. Stern & Michael E. Hasselmo

To cite this article: Quan Do, Thomas M. Morin, Chantal E. Stern & Michael E. Hasselmo (2026) A feature-based generalizable prediction model for both perceptual and abstract reasoning, Cognitive Neuroscience, 17:1, 13-29, DOI: 10.1080/17588928.2025.2599784

To link to this article: https://doi.org/10.1080/17588928.2025.2599784

COGNITIVE NEUROSCIENCE
2026, VOL. 17, NO. 1, 13–29
https://doi.org/10.1080/17588928.2025.2599784

Routledge
Taylor & Francis Group

A feature-based generalizable prediction model for both perceptual and abstract reasoning

Quan Do[a,b], Thomas M. Morin[c,d], Chantal E. Stern[a,e,f] and Michael E. Hasselmo[a,b,e]

[a]Graduate Program for Neuroscience, Boston University, Boston, MA, USA; [b]Center for Systems Neuroscience, Boston University, Boston, MA, USA; [c]Department of Radiology, Mass. General Hospital, Boston, MA, USA; [d]Psychology Department, Brandeis University, Waltham, MA, USA; [e]Department of Psychological and Brain Sciences, Boston University, Boston, MA, USA; [f]Cognitive Neuroimaging Center, Boston University, Boston, MA, USA

ABSTRACT

A hallmark of human intelligence is the ability to infer abstract rules from limited experience and apply these rules to unfamiliar situations. This capacity is widely studied in the visual domain using the Raven's Progressive Matrices. Recent advances in deep learning have led to multiple artificial neural network models matching or even surpassing human performance. However, while humans can identify and express the rule underlying these tasks with little to no exposure, contemporary neural networks often rely on massive pattern-based training and cannot express or extrapolate the rule inferred from the task. Furthermore, most Raven's Progressive Matrices or Raven-like tasks used to train neural networks consist only of symbolic challenges, whereas humans can flexibly solve both symbolic and perceptual challenges. In this work, we present an algorithmic approach to rule detection and application using feature detection, affine transformation estimation and search. We applied our model to a simplified Raven's Progressives Matrices task, previously designed for behavioral testing and neuroimaging in humans. The model exhibited one-shot inference and achieved near human-level performance in the symbolic reasoning condition of the simplified task. Furthermore, the model can express the relationships discovered and generate multi-step predictions in accordance with the underlying rule. Finally, the model can handle perceptual challenges containing continuous patterns. We discuss our results and their relevance to studying abstract reasoning in humans, as well as their implications for improving intelligent machines.

ARTICLE HISTORY
Received 12 August 2025
Revised 3 November 2025

KEYWORDS
Reasoning; symbolic; perceptual; generalization; RPM

Introduction

Humans have a remarkable reasoning capacity. In addition to identifying and recognizing similar features to aid categorization, humans can also reason in novel contexts and situations by inferring an abstract rule that is generalizable. This capacity has been studied using the Raven's Progressive Matrices (RPM) (Raven, 1941), one of the most widely used tests of 'fluid' intelligence, or the ability to reason about and solve novel problems. In this task, human participants are presented with a 3×3 matrix in which one element, or a component is intentionally left blank. The components in the matrix are related by an underlying rule. The participant must deduce the rules of the matrix to choose the correct component from a given list of available items to fill in the blank. Humans can complete many RPM and RPM-like problems after little or no exposure (Stone & Day, 1981; Vodegel Matzen et al., 1994).

Many AI researchers believe that progress in human-level AI can be made by evaluating their AI systems on the RPM, and the RPM became an important benchmark for testing the reasoning abilities of deep neural networks. Deep neural networks trained to generate texts, classify images, and play video games (Krizhevsky et al., 2017; Mnih et al., 2015; Vaswani et al., 2017) are some of the biggest success stories for AI in the 21st century, but they have not achieved human level performance on the original RPM. Surprisingly, a pre-trained Large Language Model GPT-3 (Brown et al., 2020) surpassed human capabilities in a text-based version of the RPM. Results suggest that reasoning ability may emerge from the sheer quantity of data that the model is trained on – a highly unrealistic scenario for a human reasoner (Webb et al., 2023). The jury is still out on how these large language models are displaying these seemingly emergent capabilities, but deep neural networks embedded with inductive bias on problem solving can also achieve competitive performance on several RPM-inspired datasets (An & Cho, 2020; Barrett et al., 2018; Benny et al., 2021; Hahne et al., 2019; Hu et al., 2022; Jahrens & Martinetz,

CONTACT Quan Do qdo@bu.edu Graduate Program for Neuroscience, Boston University, Boston, MA 02215, USA
Supplemental data for this article can be accessed online at https://doi.org/10.1080/17588928.2025.2599784

2020; Kerg et al., 2022; Malkinski & Mandziuk, 2022; Sinha et al., 2020; Steenbrugge et al., 2018; van Steenkiste et al., 2020; Wu et al., 2020; Zheng et al., 2019). Still, most neural network models are trained to select a correct answer, without providing an explanation for their output, and often rely on biases in the dataset, therefore failing at out-of-distribution generalization (Hersche et al., 2023; Malkinski & Mandziuk, 2022; Sinha et al., 2020; Wang et al., 2020). Furthermore, neural network approaches to solving the RPM focused entirely on the discrete symbolic version of matrix reasoning, even though the Raven's Advanced Progressive Matrices also includes continuous perceptual stimuli (Raven, 1941; Yang & Kunda, 2023). It is unclear whether computational models trained to do discrete matrix reasoning can also solve perceptual reasoning task, or vice versa. It is also an open question whether humans use different strategies or representations under different task conditions. Investigating this topic could provide insights into how humans seamlessly transition between perceptual and symbolic reasoning with minimal exposure, offering a way to bridge the gap between artificial neural networks and human cognition.

It is an open question how humans solve these problems. Nevertheless, modeling efforts include a symbolic production system (Carpenter et al., 1990), geometric analogies (Lovett & Forbus, 2017; Lovett et al., 2009), affine and set transformations (Kunda et al., 2010, 2013; Yang et al., 2022), Bayesian rule induction (Little et al., 2012), role-filler variable bindings via circular convolutions (Rasmussen & Eliasmith, 2011, 2014), and reinforcement learning (Raudies & Hasselmo, 2017). Notably, Lovett and Forbus (2017) propose that the RPM tests individuals' abilities to make effective analogies. Specifically, RPM can be seen as a complex geometric analogy problem, where analogous relation between images must be inferred. Lovett and Forbus then proposed a geometrical analogy that is based on Structure Mapping Theory (Falkenhainer et al., 1989), which assumes independence between perceptual representations and subsequent abstract analogical structure. In this framework, perceptual similarity and analogical reasoning are distinct processes. A different viewpoint, the High-Level Perception (HLP) theory of analogy (Chalmers et al., 1992; Mitchell, 1993) views analogy as the outcome of closely intertwined perceptual and abstract reasoning processes. It suggests that the creation of stimulus representations and the alignment of those representations are interdependent. Consistent with this viewpoint, past models have explicitly modeled how matrix reasoning in the RPM could be equated to searching for possible computational transformations between perceptual representations (Kunda & Goel, 2011; Kunda et al., 2013). This approach is of particular interest, since it is largely unexplored in the AI community, even though evidence from psychology and neuroscience (DeShon et al., 1995; Kunda & Goel, 2008; Prabhakaran et al., 1997; Soulières et al., 2009) suggests that human participants frequently used affine transformation to solve RPM tasks. Intuitively, this process would involve inspecting the objects in the matrix, comparing the objects, and mentally transforming the objects and estimating their perceptual similarity. Therefore, computational models in this line of approach (Kunda et al., 2010, 2013; Yang et al., 2022) represent the matrix entries in the RPM by pixel images, apply predefined pixel-operations to, and calculate pixel-level similarities between these images. However, since the human eyes cannot differentiate individual pixels on a computer screen, it's unlikely that humans have the same pixel level representation as these computational models. Additionally, the major downside of these approaches is the computational power required to operate on all pixels in a large image, or the lack of sufficient abstraction that reduces the complexity of problem solving. In fact, evidence from neural imaging studies (Schendan & Stern, 2007, 2008) suggests that humans applied mental rotation to a more abstract representation like object.

Here we further explored the affine transformation approach, built upon our earlier ideas of multi-step future prediction (Hasselmo, 2018), and introduced a feature-based algorithmic framework for both perceptual and symbolic reasoning. Instead of representing and acting on pixels, explicit symbols, or bound variables, our model works with scale-invariant features detected with the SIFT algorithm (Lowe, 1999), which share several properties in common with the responses of neurons in the IT cortex in primate vision (Ito et al., 1995). These features become the common denominators shared among multiple reasoning demands, perceptual and symbolic. Reasoning in our model involves iteratively estimating and deriving a sequence of affine transformations acting on the detected features via a model fitting procedure called random sample consensus (RANSAC) (Fischler & Bolles, 1981). The sequence is fine-tuned by similarity metrics between the predicted and desired outputs to generate a set of interpretable and generalizable operations that can be used for extrapolation.

To evaluate this framework, we applied it to a modified version of the Raven's Progressive Matrices (RPM) task comprising four task-defined conditions: Perceptual

Matching, Perceptual Reasoning, Symbolic Matching, and Symbolic Reasoning. These conditions distinguish between continuous (perceptual) and discrete (symbolic) stimulus formats, and between constant (matching) and progressive (reasoning) rules. Specifically, Perceptual Matching involves identical stimuli that are continuously repeated, while Perceptual Reasoning presents continuously varying stimuli that are progressively altered. In Symbolic Matching, identical stimuli are displayed in three discrete boxes, whereas in Symbolic Reasoning, the three boxes contain stimuli that are progressively altered. These distinctions are experimentally imposed to manipulate visual and conceptual demands rather than to reflect fundamentally different cognitive systems.

This version of the RPM was originally designed for a neuroimaging study of both symbolic and perceptual reasoning in humans (Morin et al., 2023), which revealed both shared and distinct neural signatures across task conditions. Dynamic network analyses demonstrated that reasoning-related activation in the frontoparietal control network was facilitated by a reconfiguration of functional connectivity into a 'task-ready' state – a finding interpreted as evidence for a domain-general reasoning mechanism supported by a flexible frontoparietal architecture. In that study, all four conditions engaged a common frontoparietal – visual network supporting visuospatial integration but also showed graded differences in activation depending on task demands: reasoning conditions recruited frontoparietal regions more strongly than matching, and perceptual reasoning elicited greater inferior temporal activity than symbolic reasoning, reflecting heavier reliance on low-level visual features. These results suggest a shared visuospatial mechanism modulated by task context rather than entirely distinct systems.

Building on this neural framework, the present computational model was designed to capture this organization – to reproduce both the shared visuospatial processing observed across all conditions and the subtle variations that arise from differences in feature complexity and relational learning. Affine transformations fitted to scale-invariant features implement a domain-general visual comparison process, consistent with the fMRI finding that a strong frontoparietal – visual community facilitates visuospatial integration across tasks. The greater frontoparietal activation during reasoning relative to matching mirrors the model's need to estimate multiple transformation steps rather than a simple identity or translation operation, while the increased inferior temporal involvement during perceptual reasoning corresponds to the model's reliance on a larger number of detected visual features. Under this shared task structure, our model achieved near-human-level performance on the symbolic conditions and well above chance on the perceptual conditions, returning interpretable and generalizable relationships from only a small number of examples – unlike existing deep neural network models. Together, this work provides a biologically grounded interpretation of the human neuroimaging results, suggesting that the functional reconfiguration of the frontoparietal network during abstract reasoning may reflect an iterative process of applying and refining affine transformations to capture underlying relational rules.

Methods/approach

Task design

In our modeling approach, the artificial agents experience the same task structure that human participants experienced in a modified and simplified version of the RPM (Morin et al., 2023). There are four task conditions in this modified RPM to probe four cognitive abilities: Perceptual Matching, Perceptual Reasoning, Symbolic Matching and Symbolic Reasoning. Each task condition contains 24 unique cue stimuli. Each cue stimulus was displayed four times: twice normally and twice left/right flipped (reflected over the y-axis), resulting in 384 unique trials total for the task. Along with the presentation of the cue stimuli, two answer choices are displayed on each side of the screen. The agents must make a response (0 for left and 1 for right) to indicate their answers. The four task conditions are interleaved on a trial-by-trial basis, and the agents completed all 384 trials of the task. To illustrate the diversity and visual characteristics of these stimuli, representative examples from each condition are shown in Supplemental Figures S6–S9, which depict sample tasks from the Perceptual Matching, Perceptual Reasoning, Symbolic Matching, and Symbolic Reasoning sets, respectively. These examples provide a clearer view of the types of visual patterns and relational structures used to evaluate the model's reasoning performance across different cognitive domains.

Model overview

The four task conditions – Perceptual Matching, Perceptual Reasoning, Symbolic Matching, and Symbolic Reasoning – can be reframed as a visual analogical challenge, where the relationship between A and B must be inferred to predict what follows C (Figure 2). The model identifies corresponding perceptual features between A and B, determines the

necessary affine transformations to map A onto B, and applies these transformations to B to predict C. It then refines its predictions by comparing the generated C to the correct C, adjusting the transformations accordingly.

Trial identification and areas of interest detection

In the Symbolic Matching and Symbolic Reasoning conditions, the different cues are located inside three horizontally and evenly spaced rectangles. A fourth rectangle, the last one on the right, is left blank, indicating where the answer would be. The two answer choices displayed below the cues on each side of the screen are also presented within rectangles of equal dimensions (Figure 1(c, d)). From the artificial agent's perspective, the rectangles containing the answer choice have the exact same dimensions as the blank rectangle. The artificial agent runs a contour detection procedure on the cue stimulus to get polygons that best approximate the detected contour. By checking for the number of vertices as well as the angles between the edges of the polygons, the artificial agent can reliably find all the cue rectangles as well as the blank rectangle. The number of detected rectangles also tells the artificial agent which trial conditions it is in. To verify that the detected rectangles indeed contain the cues, the artificial agent also compared the dimensions of each cue rectangle with the dimensions of the rectangles containing the choice options and only kept the ones that matched. The image content inside the detected rectangles is the only piece of information the artificial agent pays attention to. This procedure results in three areas of interest, one for each presented cue, labeled cue A, B and C.

In the Perceptual Matching and Perceptual Reasoning conditions, there is only one rectangle inside the cue stimulus that is left blank (Figure 1(a, b)), indicating where the answer would be for the human participants to fill in. The two answer choices displayed below the cues on each side of the screen are presented within rectangles of equal dimensions (Figure 1(a, b)). The artificial agent in these conditions also runs a contour detection procedure on the cue stimulus to reliably find the blank rectangle and calculate its width and height. The artificial agent can check that the detected rectangle is correct by comparing its dimension to that of the rectangle containing the answer choice. The number of detected rectangles also tells the artificial agent which trial conditions it is in. For the Perceptual Matching and Perceptual Reasoning conditions, only one rectangle should be detected. The artificial agent subsequently divides the cue stimulus into 4 equally sized rectangles that span the height of the original cue stimulus, which we thereby labeled Cue A, B, C, D. Only cue D contains

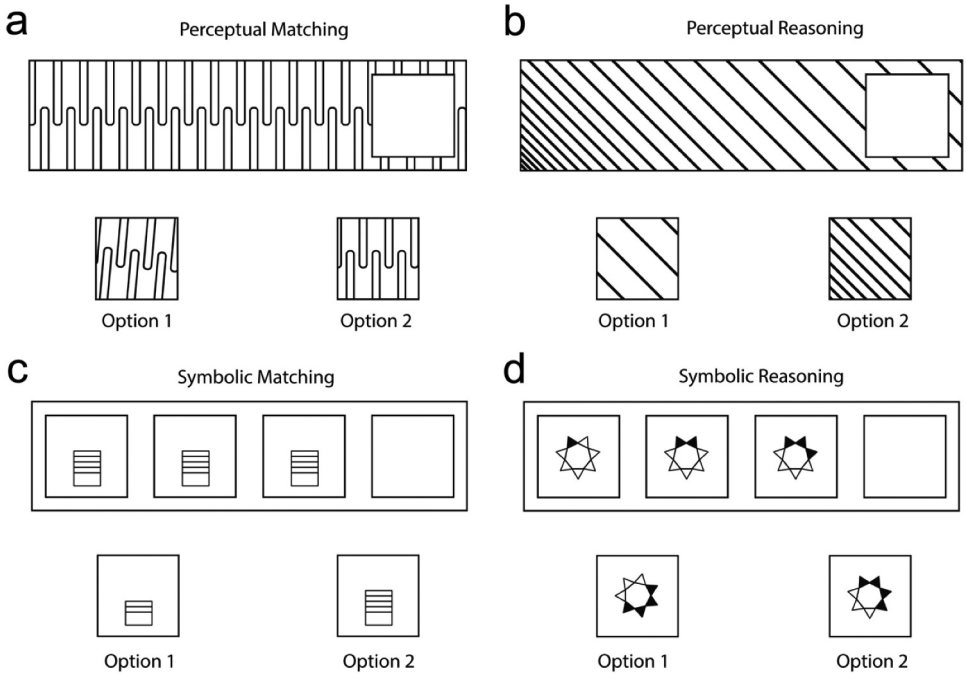

Figure 1. Task design. (a) a condition to test perceptual Matching ability. A continuous pattern is presented with two answer choices. Agents pick one of the two options to fill in the blank. Option 2 is the correct answer. (b) a condition to test perceptual reasoning. Option 1 is the correct answer. (c) a condition to test Symbolic Matching. A discrete set of symbols appears in series with two options to fill in the blank. Option 2 is the correct answer. (d) a condition to test Symbolic reasoning. Option 2 is the correct answer.

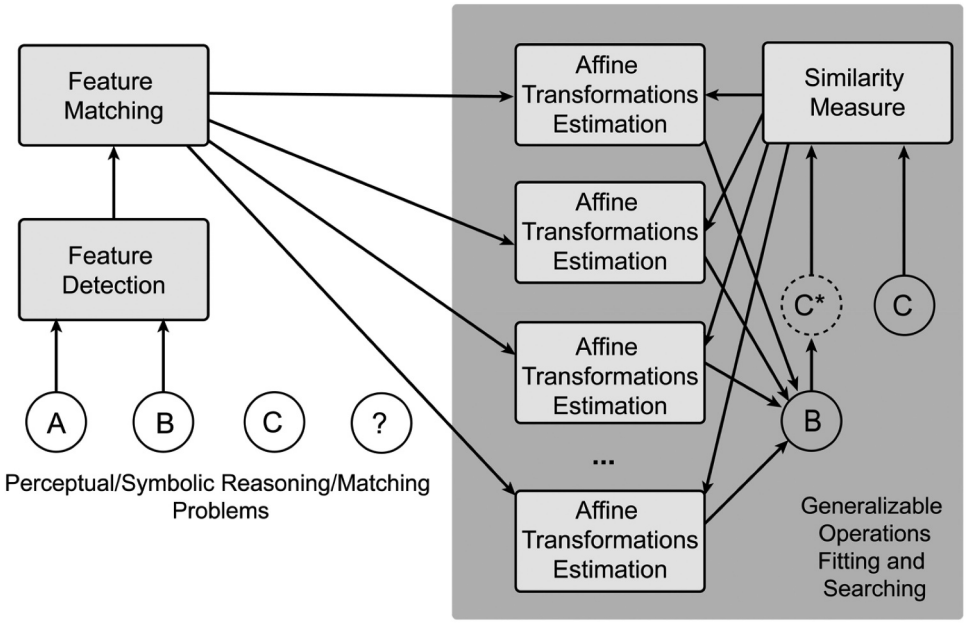

Figure 2. Model schematic. The model first identifies perceptual correspondences between A and B, determines the set of affine transformations required to map a onto B, and then applies these transformations to B to generate a prediction for C. The predicted C is compared to the correct C, and the transformations are refined iteratively to improve accuracy.

the blank square where the answer would be filled in. The cues A, B, C are the areas of interest that the artificial agent focuses on.

Feature detection and Matching

Once an area of interest is defined, the agent looks for SIFT features based on the content inside the window (Figure 3(a)). We also tried ORB (Oriented FAST and Rotated BRIEF) feature detection (Rublee et al., 2011) since it provides a free and efficient alternative to the popular SIFT algorithm. SIFT detects and describes local features in images, making it robust to scale, rotation, and illumination changes. It uses a Difference of Gaussians (DoG) for keypoint detection and computes distinctive descriptors using histograms of gradient orientations. SIFT is highly accurate but computationally expensive. ORB is a more efficient alternative to SIFT, designed for real-time applications. It combines the FAST (Features from Accelerated Segment Test) detector with the BRIEF (Binary Robust Independent Elementary Features) descriptor. ORB introduces orientation compensation to make BRIEF rotation invariant. It is faster than SIFT and suitable for embedded systems but may be less accurate in some cases. SIFT and ORB are interchangeable for our purposes. We however noticed that SIFT returns more features than ORBs and is better for the Perceptual conditions (Supplemental Figure S1). ORB can fail to find features in these conditions.

In all four conditions (Symbolic Reasoning, Perceptual Reasoning, Symbolic Matching, Perceptual Matching) of the task, there are three areas of interest for the three distinct cues. The detected features from these three windows are compared and matched using a brute force matching procedure, to locate the repeating features across the three cues. A visual feature consisted of a descriptor and a key point (that defines the location on the screen in x-y coordinates). One descriptor for a feature from cue A was compared with all the descriptors of features from cue B, using the Hamming distance, until a best match was found. Then the next descriptor for a feature from cue A was matched to all those in cue B, and so on. After all the matches are determined, only the closer descriptors are kept for quality control. This was done with a nearest neighbor distance ratio of 0.8 (arbitrary choice). Finally, the x-y coordinates of the detected, matched and chosen descriptors between cue A and B, provided by the key points, were then used as input into the Transformation Estimation step.

Transformation Estimation/RANSAC outlier detection and local similarity

In this step, after the matched features or correspondences across A, B and C are detected, we assume that there exists a repeatable sequence of affine transformations between these features, and we try to estimate the

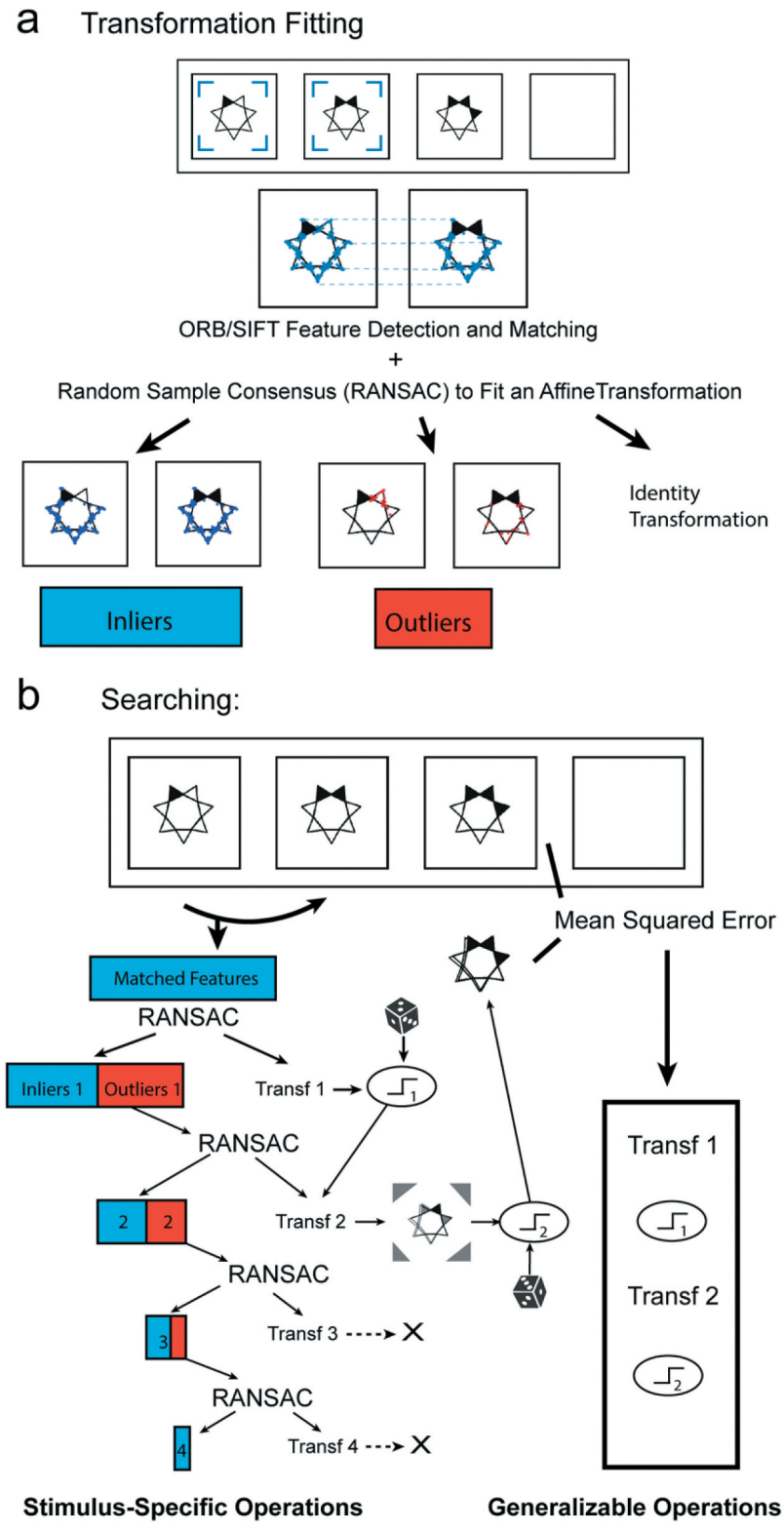

Figure 3. Operations fitting and searching procedure. a) matched ORB/SIFT features between the source and target image were used to fit an affine transformation with RANSAC. Example of an identity transformation returned by RANSAC. The outliers (red dots) are points that cannot be fitted by the identity transformation while the inliers (blue dots) are points that are identical between the source and target. b) after the first run of RANSAC, the outliers can be fitted to another RANSAC procedure, and so on, to generate a list of affine transformations that will fully transform all the matched features from the target to the source for a specific set of input-output pair. Each transformation is followed by a thresholding function with a threshold value randomly drawn from a list of prechosen thresholds. The transformation and thresholding function are then iteratively applied to the target to predict the next stimulus in the sequence. Mean squared errors between the predicted stimulus and the actual stimulus determines whether a particular transformation should be kept, and whether the chosen thresholding value would lead to a better output. The model arrives at a final list of operations that can act on each of the input images in the sequence.

parameters of these transformations as well as the number of steps in the sequence.

Random sample consensus (RANSAC) was used to robustly estimate parameters of the affine transformation between Cue A and Cue B. This ensures that we are not making errors due to faulty matched features if we solved for the transformation using the full set of detected features.

Given the initial set of feature correspondences, RANSAC was used to estimate the affine transformation while minimizing the impact of outliers. The procedure is as follows:

(1) **Random Sampling**: A minimal subset of three matched key points was randomly selected.
(2) **Model Estimation**: Using the selected points, the affine transformation matrix T was computed in homogeneous coordinates. The transformation is represented as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a11 & a12 & tx \\ a21 & a22 & ty \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where the upper-left 2×2 submatrix A (a11, a12, a21, a22) represents scaling, rotation, and shearing, and T (tx, ty) accounts for translation.

(1) **Consensus Evaluation**: The estimated transformation was applied to all matched key points, and the number of inliers was determined based on a predefined Euclidean distance threshold between the transformed points and their corresponding matches.
(2) **Iteration**: Steps 1–3 were repeated for a fixed number of iterations or until a satisfactory number of inliers was obtained.
(3) **Model Refinement**: The final affine transformation was recomputed using all inliers to improve accuracy.

By iteratively refining the transformation while rejecting mismatched correspondences, RANSAC provided a robust estimation of the affine model. The returned transformations can include identity, rotation, translation, scaling and sheering.

The features that are effectively fitted to the transformations are referred to as the 'inliers.' RANSAC also returns the × and y coordinates of the 'outliers' that do not fit the transformations (Bottom of Figure 3(a)). We call the initial inlier set I and the initial outlier set O. We then randomly sample 3 inliers from the set I and run RANSAC again on the outliers as well as the 3 recently sampled inliers (minimum of three points are needed to fit a geometric transformation) to detect whether another distinct transformation exists across a different set of point combinations. Rotation and translation only become detectable at this stage when computing transformations on outliers, whereas the previous step had identified either identity or similarity transformations. Upon finding a new transformation, we update the outlier set O with the reduced number of outliers. We continue running this procedure until there are no more outliers (Figure 3(b)). At each iteration, we computed a local similarity index. Since there is a stochastic component to the sampling of inliers at each step, resulting in different transformations being found, we do this step several times (10 times). A local similarity index is computed by applying the local transformation to cue B to predict cue C and calculating the mean squared error (MSE) between the prediction and the desired output. Note that the main difference here is instead of applying a transformation to the detected features, we are applying the transformation to every pixel in the image using an image warping function. We are distorting a part of cue B to match a part of cue C. We only kept the local transformation that results in the lowest MSE.

In summary, given cue A and B, we followed a random sampling and greedy strategy to look for the sequence of actions or transformations that returns the best local prediction of cue C (a local transformation transforms a part of cue B to match a part of C) at each step. We note that there are other strategies to sample and update the set of inliers and outliers that are yet to be explored. It's an open question what strategy a human observer would follow.

Image combination, thresholding and global similarity

To verify that the transformed output matches the desired output, we apply all the transformations in the detected action sequence to the input, add up the resulting outputs, and threshold the final output. Input and output refer to one pair of images being tested, in this case cue A and B, respectively. At each local transformation step, a non-biased coin is tossed to decide whether to round up or down the outputs generated by the transformations, introducing stochastic variability in how the transformed components are combined (Figure 3(b); bottom of Figure 4 (a)). Essentially, a threshold value is randomly selected from a predefined list (half, two-thirds, or one-third of the maximum pixel intensity) and applied to the elementwise sum of the transformed images. This operation acts as an overlap gating mechanism: areas with high summed intensity –

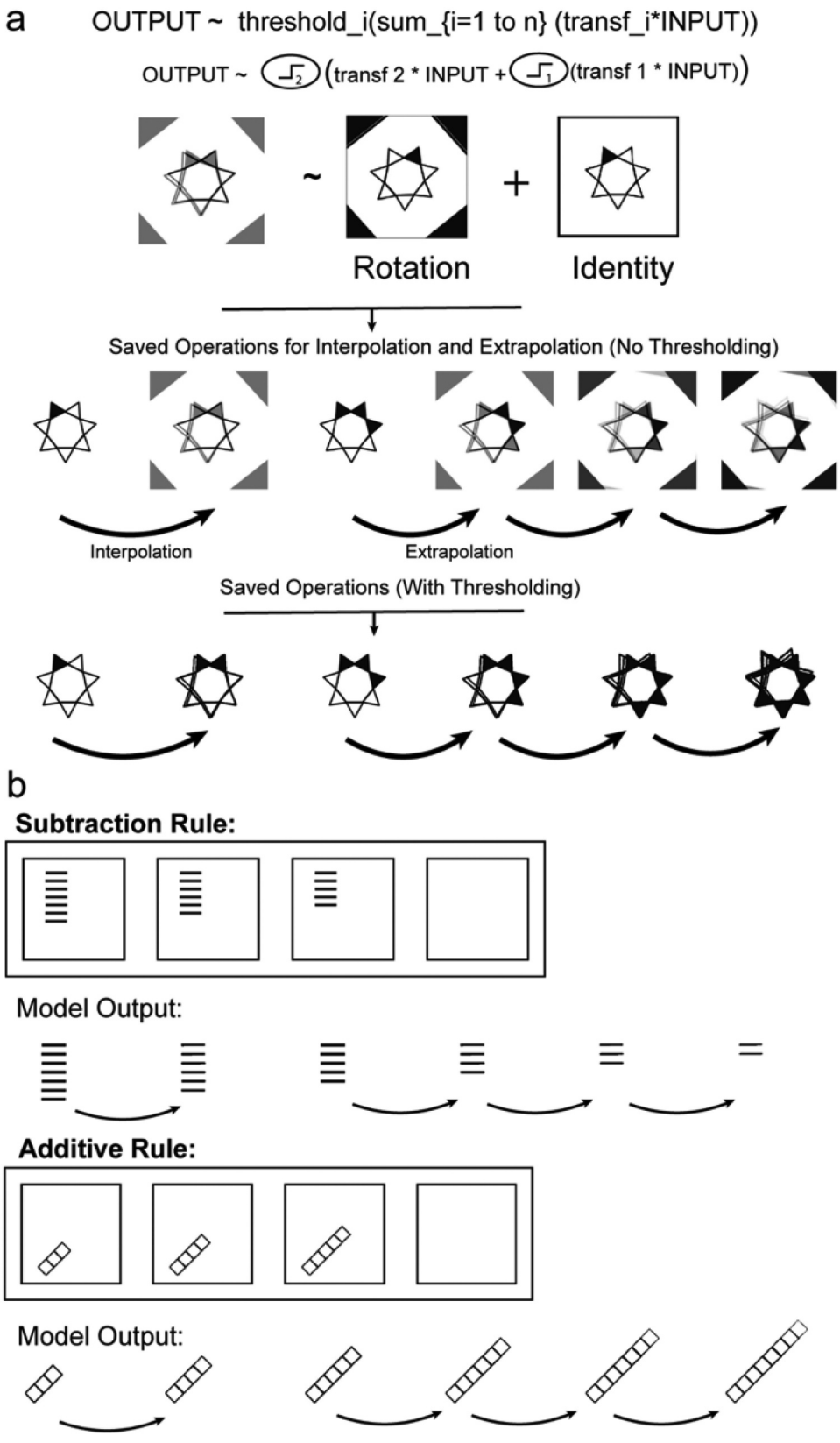

Figure 4. Model finds generalizable solutions that allow for extrapolation. a) a general formulation for the operations that can be discovered by the model. The formulation can be understood to a composition of multiple affine transformations with the output going to different thresholding functions at each step. An example of a rotation and identity operation being combined and applied to an input image. Applying the combined operations multiple times enables multi-step predictions that are in accordance with the underlying rule. Adding the thresholding functions allow for removal of the background. b) examples of the model working on stimulus sets governed by subtraction rule and an additive rule. While the affine transformation acted on the entire image, the thresholding functions allow for robust subtraction and addition of parts of the image. Specifically, overlapping parts are enhanced resulting in addition in the first example, whereas non-overlapping parts are removed, resulting in subtraction in the second example.

corresponding to overlapping or consistently transformed areas – are preserved, while non-overlapping or weakly corresponding areas are suppressed. This random thresholding scheme allows the model to flexibly combine or suppress transformed outputs at each iteration.

To further evaluate the influence of this step, we conducted a sensitivity analysis to systematically examine how threshold selection affects model performance. The original random draw mechanism was preserved, but the threshold list was reduced to two active parameters (threshold1 and threshold2), whose numerical values were systematically varied between 0.10 and 0.90 in a 17×17 grid (289 total combinations). Each configuration was evaluated across 96 trials in the Symbolic Reasoning condition, and the resulting performance landscape was visualized as a heatmap (Supplemental Figure S2).

We compute a similarity index between the transformed output to the desired output. We again used the mean squared error (mse) between two images as a measure of global similarity. A lower mean squared error indicates a better fit.

We ran the greedy local search procedure described above multiple times (2 or 3 times), each time with different thresholding values randomly sampled from the preselected values, computed the global similarity index, and selected the search thread that resulted in the lowest MSE. Thresholding is important because at each step, we are distorting a part of a cue to match it to another part of another cue, and we want to be able to subtract the parts that aren't relevant after the distortion process or highlight the parts that might be important. This process would effectively let us manipulate and combine parts of an image to transform it to a different image. We apply the sequence of steps from the selected search thread to cue C to predict the answer in the blank square. The full algorithm is described and summarized in Box 1. The implementation can be found at https://github.com/qdo1010/SRPM_GPM

Task difficulty analysis

To quantify relative task difficulty across conditions, we computed several visual and geometric metrics for each cue image in the dataset. These measures characterized both low-level visual complexity and transformation consistency.

(1) **Entropy**: For each cue image, a 256-bin intensity histogram was computed using cv2.calcHist, normalized to probabilities, and converted to Shannon entropy using $H=-\sum p*\log(p)$, over non-zero bins.
(2) **Edge Density**: The proportion of Canny edge pixels relative to total image area was calculated to capture overall visual texture.
(3) **Keypoint Counts**: The total number of SIFT and ORB keypoints detected across Boxes A and B was used as an estimate of visual feature richness.
(4) **Geometric Diagnostics**: For each trial, SIFT and ORB feature matches were computed and refined using RANSAC. From this, we recorded both the total number of matches and the inlier ratio, defined as the number of geometrically consistent correspondences divided by total matches.

Bootstrap resampling (2,000 iterations, fixed random seed) was used to estimate 95% confidence intervals for each metric across the four experimental conditions. The resulting summary statistics were visualized as bar plots (Supplemental Figure S3).

Results

The model found generalizable operations from stimulus-specific operations, allowing for interpolation and extrapolation under different rules

The model after the fitting and searching procedures can arrive at a list of operations. We formulate the structure of these operations (Figure 4(a)) to describe how these are iteratively applied to the input. Essentially, after a transformation, an output is added to the previous output and a thresholding function is applied to either highlight the important parts or remove the extraneous parts of the image.

For instance, a rotation can be combined with an identity transformation to color an additional part of the hexagram (Figure 4(a)). The part of the hexagram that is essential for highlighting successful rule application, can then be highlighted after applying the thresholding function (Figure 4(a)). With the combination of the transformation and thresholding function, the model can therefore manipulate parts of the image instead of acting on the entire image, allowing for a more robust image manipulation scheme. Two examples further highlight how the model can either subtract or add parts to an image (Figure 4(b)). Finally, after discovering these operations (combinations of transformations and thresholding functions), the model can apply them iteratively to an image and its subsequent outputs to generate predictions multiple steps into the future (Figure 4(a, b)).

These examples also illustrate the interpretability of the proposed approach. Each affine transformation learned by

Algorithm: Multi-Stage RANSAC for Affine Transformation Estimation.

Input:

- Matched keypoints between two images: {(xi,yi)↔(xi′,yi′)}
- RANSAC parameters:
  - Number of iterations N
  - Inlier threshold τ
  - Similarity computation iterations $t = 10$
  - Preselected threshold values: {0.5 M,0.67 M,0.33 M} (where M is the max pixel value)

Output:

- Set of detected affine transformations
- Final inlier and outlier set
- Optimal local transformation with the lowest mean squared error (MSE)
- Final transformed image with thresholding applied

1. Initial RANSAC Estimation:
a. Apply RANSAC to estimate an initial affine transformation
b. Classify matched key points into:
–Inliers I (points that fit)
–Outliers O (points that do not fit)

2. Iterative Outlier Refinement: While O:
a. Randomly sample three inliers from I.
b. Run RANSAC on O ∪ (sampled inliers) to detect a new transformation
c. If a new transformation is found, update the outlier set O with the remaining unmatched points.

3. Transformation Selection via Similarity Index:
a. Repeat steps 2a-2c for $t = 10$ iterations due to the stochastic nature of inlier sampling.
b. For each transformation compute a local similarity index by:
–Applying to all pixels in image cue B using image warping.
–Predicting image cue C.
–Calculating the Mean Squared Error (MSE) between prediction and ground truth.
c. Store the transformation with the lowest MSE as the final local transformation.

4. Global Verification and Thresholding:
a. Apply all detected transformations sequentially to cue A to generate transformed outputs. b. Sum all the transformed outputs.
c. Apply a thresholding operation:
–Randomly select a threshold from {0.5 M, 0.67 M, 0.33 M}
–Toss a fair (non-biased) coin at each step to decide whether to round up or down the transformed pixel values.
–Enhance overlapping components, and remove non-overlapping components.
d. Compute the global similarity index by calculating the MSE between the final transformed output and the desired output.

5. Greedy Local Search for Optimal Transformation:
a. Repeat steps 1–4 multiple times (2 or 3 times), each time with a different randomly sampled thresholding value.
b. Compute the global similarity index for each run.
c. Select the transformation sequence that resulted in the lowest MSE as the final optimal transformation.

6. Output:
–Final set of transformations
–Refined inlier/outlier sets
–Best local and global transformation
–Transformed image with thresholding applied

Box 1. Multi-Stage RANSAC for Affine Transformation Estimation.

the model can be directly inspected through its transformation matrix, which encodes a transparent and human-interpretable geometric rule (e.g., rotation, translation, scaling, or reflection). Because these matrices are explicitly recovered during inference, we can read out how the model is manipulating the image at each step. For example, in the *star (hexagram)* task, the model first identifies the identity matrix, then discovers a rotation matrix, which together explain how the system aligns and colors an additional segment of the hexagram (Figure 4(a)). In other cases, the model learns translation matrices that shift object components to complete a missing pattern or reflection matrices that generate mirror-symmetric structures. When combined with the thresholding function, these geometric transformations act as an overlap-gating mechanism that selectively preserves overlapping regions and suppresses non-overlapping ones (Figure 4(b)). In effect, this process can add or subtract specific image

components, effectively implementing symbolic-like manipulations through geometric operations. Because these transformations act directly on pixel coordinates and are compositional, they yield a transparent mapping between the model's internal parameters and its reasoning behavior. In this way, the iterative application of affine matrices constitutes a sequence of explicit and interpretable geometric reasoning steps.

The same model can be applied to continuous stimuli, demonstrating complex pattern completion and prediction

Humans can generalize using continuous perceptual stimuli. Here we demonstrated that our model can be applied to continuous patterns in addition to discrete symbols. We made no significant changes to the model as described in Figure 3. The only change is that instead of looking at the three areas of interest as defined by the three squares in the Symbolic Reasoning Condition (Figure 1(d)), the model divides the continuous patterns into 4 equal slides, and operates on the three slides containing the patterns, while trying to predict the slide containing the blank square (See Methods). Going through the same fitting and searching procedure as described in Figure 3, the model can generate complex patterns to fill in the blank and extrapolate the patterns beyond what was given (Figure 5).

The model prediction is noisy and declines in quality over multiple future steps, but for the task of filling in the blank, the first prediction from the model is sufficient, and can be directly compared to the list of available answer choices to make the correct decision.

The model achieved comparable performance to humans on most of the task conditions

We ran the model on the 384 trials that humans are given in the modified Raven task (Morin et al., 2023). The task has 4 conditions that tests different perceptual and symbolic reasoning skill (See Methods). The Symbolic Rule and Matching condition tests reasoning with discrete symbols while the Perceptual Rule and Matching conditions test reasoning ability on continuous patterns. On a single best run, the model solved 86/96 trials in the Symbolic Reasoning condition, 96/96 trials in the Symbolic Matching condition, 78/96 trials in the Perceptual Matching condition and 63/96 trials in the Perceptual Reasoning condition. For each condition, the model generates a prediction and selects between two answer options. We quantified the difference between the two answer choices using two complementary image-based similarity measures: mean squared error (MSE) and structural similarity index (SSIM). We analyzed model accuracy as a function of the MSE and SSIM between the two answer options and found a nonlinear relationship (Supplemental Figure S4).

Since there is a stochastic component to the model, we ran the model 5 times on the full task to simulate 5 artificial agents (arbitrary choice) and to compare with the data collected from 27 human participants tested on the task (behavior data from Morin et al., 2023, excluding subjects ($n = 1$) who failed to achieve at least 80% accuracy on each of the four conditions). We plotted the human data and the model's performance side by side, grouped by the 4 conditions of the task (Figure 6).

We noted that the models performed well on the Symbolic Reasoning (88.75% accuracy, std. = 31.63%) and Symbolic Matching (100% accuracy, std. = 0%) condition. Humans achieved a 95.6% (std. = 20.5%) and 96.36% (std. = 18.73%) accuracy rates on these conditions, respectively. The model also works with the continuous texture task conditions and can achieve an 82.29% accuracy rate (std. = 38.21%) on the Perceptual Matching condition, compared to human level of 93.54% (std. = 24.58%). However, the model struggled on the Perceptual Reasoning condition when compared to human performance (mean = 94.96%, std. = 21.86%), despite performing above chance (63.33% accuracy rate, std = 48.24%).

To further evaluate the contribution of the transformation-based reasoning component, we compared our model's performance to a simple similarity baseline that skips the affine transformation step. In this baseline, each answer choice was compared directly to the three discrete cue boxes using the structural similarity index (SSIM), and the total similarity across boxes determined the chosen answer. This approach effectively implements a template-matching strategy. As expected, it achieves perfect accuracy (100%) on the Symbolic Matching condition but performs substantially worse on Symbolic Reasoning (71.9%), where the solution requires applying a transformation rather than matching surface features (Supplemental Figure S5). These results confirm that the proposed method captures relational reasoning beyond direct visual similarity.

We further explored the relationship between the model and humans by comparing the number of scale-invariant features detected and matched by the model (Figure 3(a)) with the performance of both the model and humans on different task conditions. We fitted linear models and found that only in the Perceptual Reasoning condition, both humans (intercept = 0.94, slope = 7.344e-5, p-value = 0.0046, R-squared = 0.003)

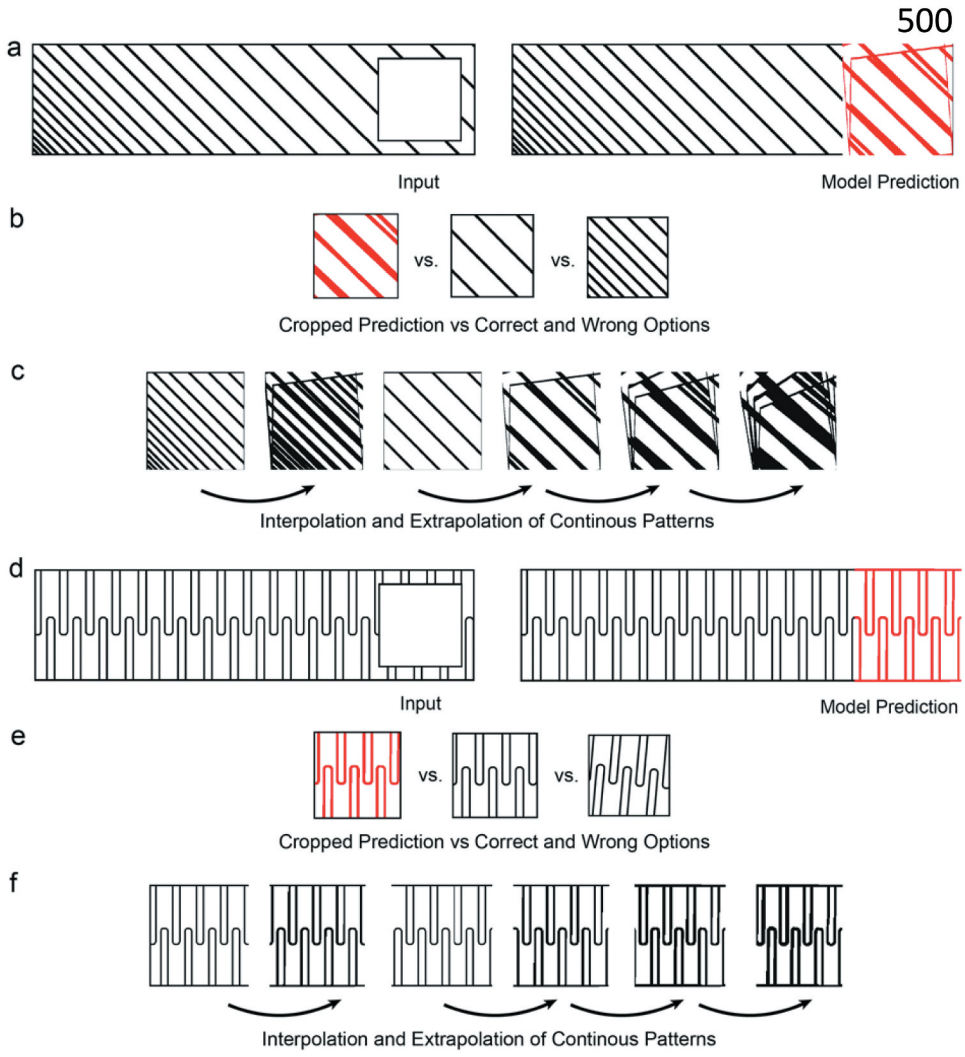

Figure 5. Model works on continuous perceptual stimuli to perform pattern completion and extrapolation. Two examples of the model generating continuous patterns to fill in the blank and to extrapolate, beyond what was given. a, d) given a pattern with a blank square, the model can predict the missing patterns. b, e) the full prediction can be cropped to the size of the blank square and compared to the different answer options. c, f) the model demonstrates interpolation and extrapolation with continuous patterns.

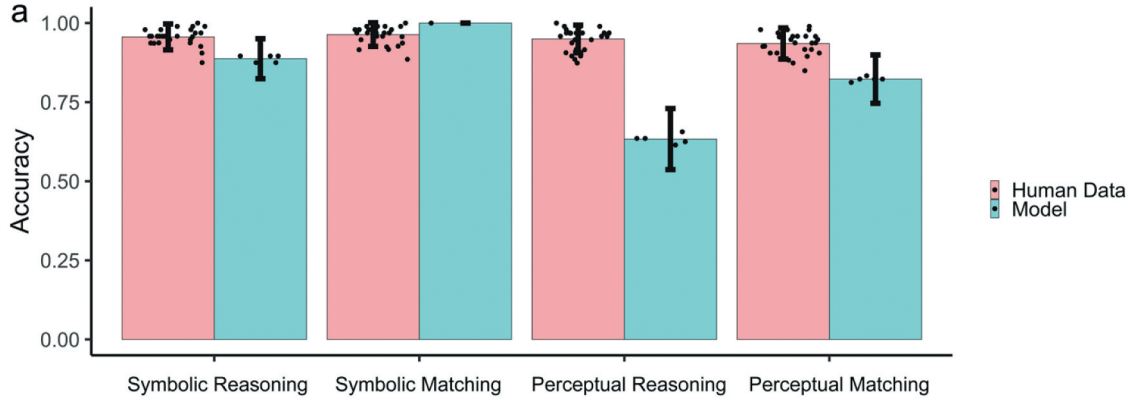

Figure 6. Model versus humans on different task conditions. a) comparing the performance of the model with human participants performing the task. The model performs well above chance and reaches close to human level in several task conditions. Error bar indicates trial-to-trial variability (384 trials). Scatter dots indicate individual variability (27 humans and 5 artificial agents).

and the model (intercept = 0.59, slope = 0.0004, p-value = 0.0023, R-squared = 0.019) perform better when there are more scale-invariant features found. This is somewhat surprising since one would suspect that humans might perform better on visual inputs with less features. The model however would predict that more features mean better chance of finding and fitting better affine transformations with RANSAC. Our findings suggest that this prediction might hold true for humans.

Discussion

In this work, we introduced a generalizable prediction model that can reason with both continuous patterns and discrete symbols. Given a series of three observations, continuous or discrete, our model returns an interpretable and generalizable sequence of operations that can be used to predict multiple future items in the series. The operations that the model discovered are reflective of the multiple underlying relationships that governed the series of observations. In a sense, the model has begun to capture some element of inductive reasoning, or the process of drawing general conclusions from a few specific observations, which is what the original RPM task was designed to test (Kunda et al, 2013). We discuss the applicability of our model as well as some future directions.

Relation inference with RANSAC

Perhaps the most important part of our model is the Random Sample Consensus (RANSAC) approach to fitting an affine transformation. Typically, the standard way to estimate an affine transformation is by using linear least squares (Zikan, 1991). This is essentially finding a best fit linear function to the data points. However, if we were to be fitting a function using a linear regression to infer a relationship, and if there were multiple relationships between the datapoints, this fitted function would be skewed away from the true relationships. The function fitting process would output an average relationship for all the datapoints. This is not desirable for a model that wants to capture all the underlying relationships. RANSAC however splits the data into inliers and outliers, and the fitted function is fitted only to the inliers, giving us one specific relationship that governs the inlier sets. We can then move on to the outlier set to infer the next relationship. This is beneficial in all the Raven tasks because the different parts within a matrix element have different relationships with their corresponding parts in another matrix element. The rule usually involves identifying and inferring these distinct relationships and applying them accordingly.

Part-whole image manipulation with adaptive thresholding

Once all the individual relationships are inferred, we need to be able to identify and manipulate the different parts of the matrix element. This is a challenging step, and we came up with an ad-hoc solution where we still apply the transformation to the entire matrix element instead of searching for individual parts. We rely on the thresholding function to add or subtract parts of the image. Overlapping parts are typically important and are retained with the thresholding function whereas non-overlapping parts are typically deleted. This is however not an ideal solution since in a task with complex patterns, we notice that the predicted pattern often gets corrupted, therefore hindering the model performance. This was noticeable especially in the Perceptual Reasoning condition, where our model underperformed. Ideally, we should crop different regions of the image and apply the appropriate transformation to only the parts that were cropped. To do this, we however would need to be able to identify which parts of the image correspond to which feature sets. Object or shape detection, as well as the understanding of part-whole relations within an image would be beneficial here to constraint the mapping process.

Continuous patterns, scale-invariant features and discrete symbols

Our model works on both continuous patterns and discrete symbols despite using scale-invariant features as the underlying representations. It is not difficult to imagine that scale-invariant feature is a bridge that links our continuous perceptual experience of the world to our symbolic representations. This point is illustrated in the field of computer vision, where the standard workflow involves detecting scale invariant features from a complex visual scene (a continuous perceptual experience) to identify a category or concept (a symbolic representation). Here we show that for most of the reasoning demands in the simplified Raven task, scale invariant feature is the right level of abstraction. Humans can flexibly switch between the different levels of abstraction to suit the reasoning demands, but this is beyond the scope of our work described here.

Simplified Raven task and applicability to other Raven-like tasks

Most RPM and RPM-like tasks usually have a 3×3 matrix structure whereas the modified Raven task we are using has a 1-dimensional structure. This raises the question of how applicable our model is to the traditional RPM as

well as other Raven-like tasks. We argue that the core of the RPM, problem solving requires inferring the relationship between the different matrix elements. The spatial structure of a task is arbitrary. The spatial structure may guide the rule inference process, but it is not crucial that the relations being inferred must strictly obey the spatial arrangement. Therefore, our model, though tested on a fundamental 1-dimensional series structure, is directly applicable to other Raven-like tasks, since its core process is relational inference. However, it is interesting to note that in most RPM and RPM-like tasks, including the task we are using, the relations being inferred often follow a left-to-right, top-to-bottom spatial directions. This spatial bias is hard coded into our model. In a 3×3 task where the blank matrix element is in the top left corner instead of the bottom right, our model would fail to generalize, whereas a human participant would most likely still be able to figure out the rule. There would need to be an additional process in the model that could flexibly infer how the spatial structure of the task relates to the relations being inferred. We have yet to implement this process since it is not necessary for the task we are using, but we think it involves testing our relation inference process along different vectors pointing in different directions (left to right, right to left, top to bottom, bottom to top, etc.) by trial-and-error.

Another key difference between the modified RPM task we are using, and the full RPM task is that ours only includes 'sequence' items. The full task also includes items that require an 'addition/subtraction' or 'set membership' operation (see Rasmussen & Eliasmith, 2014 for a good conceptualization of this). These are operations that could be included in the model that we proposed. However, we suspect that we would need explicit symbols, or representations more abstract than scale-invariant features to work with these operations.

Relationship to functional reconfiguration of frontoparietal and visual networks

Using network analysis on fMRI data, work from our lab (Morin et al., 2023) found a stable community structure among frontoparietal and visual brain regions that formed during the simplified Raven task. This community was maintained across all four task conditions, namely Perceptual Reasoning, Perceptual Matching, Symbolic Reasoning, and Symbolic Matching. The four task conditions differ along two orthogonal dimensions: perceptual vs. symbolic and matching vs. reasoning. The perceptual conditions involve continuous visual patterns, where relational information is embedded in low-level image features such as texture, orientation, or gradient. In contrast, the symbolic conditions involve discrete, well-bounded elements (e.g., geometric shapes), where relations are expressed between objects rather than across continuous space. The matching conditions require the identification of a constant relation (typically an identity or direct correspondence), whereas the reasoning conditions require the inference of a progressive or transformational rule that links items across space or sequence. These distinctions are task-defined and were introduced to systematically vary visual and conceptual demands. Importantly, they are not assumed to reflect different underlying mechanisms but rather to probe how humans adapt to differences in stimulus structure and rule complexity. Yet, despite being experimentally imposed, we observed both commonalities and distinctions in the neural data. In Morin et al. (2023), we showed that all four task conditions engage a shared frontoparietal – visual network that supports visuospatial integration but also exhibit graded differences in activation depending on task demands. Specifically, reasoning conditions recruited frontoparietal control regions more strongly than matching, and perceptual reasoning elicited greater inferior temporal activity than symbolic reasoning, reflecting increased reliance on low-level visual features. These findings highlight a core visuospatial mechanism modulated by task context rather than entirely distinct systems. The computational model presented here was designed to capture this organization – to reproduce both the shared visuospatial processing observed across all conditions and the subtle variations that arise from differences in feature complexity and relational learning.

We postulated that the formation of a strong frontoparietal-visual community facilitates the integration of visuospatial information. This is consistent with our model that works on all task conditions, and consistent with the process of fitting affine transformation to scale invariant features. Furthermore, we found that the frontoparietal cortex was significantly more active for the reasoning conditions compared to the matching conditions. This is also consistent with the model since in the matching conditions, typically only an identity relation is sufficient for the discrete symbols, and sometimes a simple translation operation for the continuous patterns. Humans also showed increased activity in inferior temporal cortex on the perceptual reasoning condition compared to the perceptual matching condition (and overall, on the perceptual conditions compared to the symbolic conditions). This is consistent with the idea that there are many more relevant scale-invariant features that need to be detected in the perceptual conditions, especially in the perceptual reasoning condition. It also supports our finding of a linear relationship between human performance in the perceptual reasoning

condition and the number of scale-invariant features detected by our proposed model. One finding in the fMRI study that was not consistent with our model is the observation that the cognitive control network has greater activation during the symbolic reasoning condition in comparison to the perceptual reasoning condition. This may suggest that human participants are using more abstract representations like shape and object in the symbolic reasoning condition, instead of the scale invariant features employed by our model. As previously discussed, our model would greatly benefit from the incorporation of an additional process that clusters and categorizes features into more abstract representations.

Algorithms, out-of-distribution generalization, and neural networks

Humans have a remarkable ability to generalize and extrapolate under various contexts even with a limited number of observations. In this paper we present an algorithmic framework that performs single-example inference, generalizes, and extrapolates across a wide range of task demands. It is hardly surprising that an algorithm hand-crafted to a specific task can have good task performance and generalize to various instances of the task. However, to solve a novel task with varying level of complexity (P vs NP, for instance) would require significant redesigning of the algorithm. The real challenge is in designing an algorithm that can rewrite itself to adapt to the changing task demands. Our algorithmic framework illustrates how simple operations can be flexibly composed to perform different tasks. We argue that the tasks we are using, while simple to perform, vary in their reasoning demands, as shown by the activations of different brain networks under different task conditions by Morin et al. (2023).

One limitation of our work, and the limitation of algorithmic design in general, is the data preprocessing and feature selection that constrains the algorithm to work only on our chosen task domain (image sequence). Deep Neural Networks, on the other hand, have been successful in solving tasks of many domains including but not limited to texts, images, discrete actions, and proprioceptive inputs (Reed et al., 2022). It's natural to then wonder whether neural networks can mimic algorithms, to get the best of both worlds. This is the major focus of an emerging field called Neural Algorithmic Reasoning (Veličković & Blundell, 2021). Unfortunately, the out-of-distribution generalization capability of deep neural networks is lacking and not well-understood. For example, going back to Raven-inspired tasks, the state-of-the-art performance of neural networks on the extrapolation regime of the Procedurally Generated Matrices (Barrett et al., 2018), in which test problems contain feature values outside the range of those observed in the training set, is currently at 25.9% (Malkinski & Mandziuk, 2022; Sinha et al., 2020; Wang et al., 2020). It's therefore unclear what classes of algorithms a neural network can mimic, and whether neural networks can discover novel algorithms beyond the training set, though progress is being made (Bevilacqua et al., 2023; Ibarz et al., 2022; Xu et al., 2020, 2021).

Out-of-distribution generalization is a daunting yet exciting challenge. In future work, a major focus of ours will be to learn from how humans reason, build upon our algorithmic framework, and create neural circuits that generalize. Ultimately, human problem solving, regardless of how complex, is reducible to biological neural networks performing computations.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work is supported by the Office of Naval Research [ONR MURI N00014-19-1-2571, ONR MURI N00014-16-1-2832, NSF 1625552, ONR DURIP N00014-17-1-2304], and Kilachand Fund Award.

ORCID

Quan Do http://orcid.org/0000-0002-2723-2579

References

An, J., & Cho, S. (2020). Hierarchical transformer encoder with structured representation for abstract reasoning. *IEEE Access* (8, 200229– 00236). IEEE Access. https://doi.org/10.1109/ACCESS.2020.3035463

Barrett, D., Hill, F., Santoro, A., Morcos, A., & Lillicrap, T. (2018). Measuring abstract reasoning in neural networks. In *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Sweden (pp. 511–520). https://proceedings.mlr.press/v80/barrett18a.html

Benny, Y., Pekar, N., & Wolf, L. (2021). Scale-localized abstract reasoning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 12557-12565). PMLR. https://doi.org/10.48550/arXiv.2009.09405

Bevilacqua, B., Nikiforou, K., Ibarz, B., Bica, I., Paganini, M., Blundell, C., Mitrovic, J., & Veličković, P. (2023). Neural algorithmic reasoning with causal regularisation. In International Conference on Machine Learning (pp. 2272-2288). PMLR. https://doi.org/10.48550/arXiv.2302.10258

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G.,

Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901. https://doi.org/10.48550/arXiv.2005.14165

Carpenter, P. A., Just, M. A., & Shell, P. (1990). What one intelligence test measures: A theoretical account of the processing in the Raven progressive matrices test. *Psychological Review*, *97*(3), 404–431. https://doi.org/10.1037/0033-295X.97.3.404

Chalmers, D. J., French, R. M., & Hofstadter, D. R. (1992). High-level perception, representation, and analogy: A critique of artificial intelligence methodology. *Journal of Experimental and Theoretical Artificial Intellige*, *4*(3), 185–211. https://doi.org/10.1080/09528139208953747

DeShon, R. P., Chan, D., & Weissbein, D. A. (1995). Verbal overshadowing effects on Raven's advanced progressive matrices: Evidence for multidimensional performance determinants. *Intelligence*, *21*(2), 135–155. https://doi.org/10.1016/0160-2896(95)90023-3

Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, *41*(1), 1–63. https://doi.org/10.1016/0004-3702(89)90077-5

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, *24*(6), 381–395. https://doi.org/10.1145/358669.358692

Hahne, L., Lüddecke, T., Wörgötter, F., & Kappel, D. (2019). Attention on abstract visual reasoning (arXiv: 1911.05990). arXiv preprint. https://doi.org/10.48550/arXiv.1911.05990

Hasselmo, M. E. (2018). A model of cortical cognitive function using hierarchical interactions of gating matrices in internal agents coding relational representations. *arXiv: 1809.08203 [q-Bio]*. http://arxiv.org/abs/1809.08203

Hersche, M., Zeqiri, M., Benini, L., Sebastian, A., & Rahimi, A. (2023). A neuro-vector-symbolic architecture for solving Raven's progressive matrices. *Nature Machine Intelligence*, *5*(4), Article 4. 363–375. https://doi.org/10.1038/s42256-023-00630-8

Hu, S., Ma, Y., Liu, X., Wei, Y., & Bai, S. (2022). *Stratified rule-aware network for abstract visual reasoning* (arXiv: 2002.06838). arXiv. https://doi.org/10.48550/arXiv.2002.06838

Ibarz, B., Kurin, V., Papamakarios, G., Nikiforou, K., Bennani, M., Csordás, R., Dudzik, A. J., Bošnjak, M., Vitvitskyi, A., Rubanova, Y., Deac, A., Bevilacqua, B., Ganin, Y., Blundell, C., & Veličković, P. (2022). A generalist neural algorithmic learner. Proceedings of the first learning on graphs conference, 2. pp. 1–2. 23. https://proceedings.mlr.press/v198/ibarz22a.html

Ito, M., Tamura, H., Fujita, I., & Tanaka, K. (1995). Size and position invariance of neuronal responses in monkey inferotemporal cortex. *Journal of Neurophysiology*, *73*(1), 218–226. https://doi.org/10.1152/jn.1995.73.1.218

Jahrens, M., & Martinetz, T. (2020). *Solving Raven's progressive matrices with multi-layer relation networks* (arXiv: 2003.11608). arXiv. https://doi.org/10.48550/arXiv.2003.11608

Kerg, G., Mittal, S., Rolnick, D., Bengio, Y., Richards, B., & Lajoie, G. (2022). On neural architecture inductive biases for relational tasks. *(arXiv: 2206.05056) arXiv*. https://doi.org/10.48550/arXiv.2206.05056

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

Kunda, M., & Goel, A. K. (2008). How thinking in pictures can explain many characteristic behaviors of autism. In *2008 7th IEEE International Conference on Development and Learning*, Monterey, CA (pp. 304–309). IEEE. https://doi.org/10.1109/DEVLRN.2008.4640847

Kunda, M., & Goel, A. K. (2011). Thinking in pictures as a cognitive account of autism. *Journal of Autism and Developmental Disorders*, *41*(9), 1157–1177. https://doi.org/10.1007/s10803-010-1137-1

Kunda, M., McGreggor, K., & Goel, A. (2010). Taking a look (literally!) at the Raven's intelligence test: Two visual solution strategies. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, Portland, OR (Vol. 32, No. 32). https://escholarship.org/uc/item/2qf752cs

Kunda, M., McGreggor, K., & Goel, A. K. (2013). A computational model for solving problems from the Raven's progressive matrices intelligence test using iconic visual representations. *Cognitive Systems Research*, *22-23*, 47–66. https://doi.org/10.1016/j.cogsys.2012.08.001

Little, D. R., Lewandowsky, S., & Griffiths, T. L. (2012). A Bayesian model of rule induction in Raven's progressive matrices. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, Sapporo, Japan (Vol. 34, No. 34). https://escholarship.org/uc/item/0227t8z1

Lovett, A., & Forbus, K. (2017). Modeling visual problem solving as analogical reasoning. *Psychological Review*, *124*(1), 60–90. https://doi.org/10.1037/rev0000039

Lovett, A., Tomai, E., Forbus, K., & Usher, J. (2009). Solving geometric analogy problems through two-stage analogical mapping. *Cognitive Science*, *33*(7), 1192–1231. https://doi.org/10.1111/j.1551-6709.2009.01052.x

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, Kerkyra, Greece (pp. 1150.

Malkinski, M., & Mandziuk, J. (2022). Multi-label contrastive learning for abstract visual reasoning. *IEEE Transactions on Neural Networks and Learning Systems*, *PP*. https://doi.org/10.1109/tnnls.2022.3185949

Mitchell, M. (1993). *Analogy-making as perception: A computer model*. MIT Press.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), Article 7540. 529–533. https://doi.org/10.1038/nature14236

Morin, T. M., Moore, K. N., Isenburg, K., Ma, W., & Stern, C. E. (2023). Functional reconfiguration of task-active frontoparietal control network facilitates abstract reasoning. *Cerebral Cortex*, *33*(10), 5761–5773. https://doi.org/10.1093/cercor/bhac457

Prabhakaran, V., Smith, J. A., Desmond, J. E., Glover, G. H., & Gabrieli, J. D. (1997). Neural substrates of fluid reasoning: An fMRI study of neocortical activation during performance of the Raven's progressive matrices test. *Cognitive Psychology*, *33*(1), 43–63. https://doi.org/10.1006/cogp.1997.0659

Rasmussen, D., & Eliasmith, C. (2011). A neural model of rule generation in inductive reasoning. *Topics in Cognitive Science*, *3*(1), 140–153. https://doi.org/10.1111/j.1756-8765.2010.01127.x

Rasmussen, D., & Eliasmith, C. (2014). A spiking neural model applied to the study of human performance and cognitive decline on Raven's advanced progressive matrices. *Intelligence*, *42*, 53–82. https://doi.org/10.1016/j.intell.2013.10.003

Raudies, F., & Hasselmo, M. E. (2017). A model of symbolic processing in Raven's progressive matrices. *Biologically Inspired Cognitive Architectures*, *21*, 47–58. https://doi.org/10.1016/j.bica.2017.07.003

Raven, J. C. (1941). Standardization of progressive matrices, 1938. *British Journal of Medical Psychology*, *19*(1), 137–150. https://doi.org/10.1111/j.2044-8341.1941.tb00316.x

Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S. G., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J. T., Eccles, T., Bruce, J., Razavi, A., Edwards, A., Heess, N., Chen, Y., Hadsell, R., Vinyals, O., Bordbar, M., & de Freitas, N. (2022). A generalist agent. arXiv preprint (arXiv: 2205.06175). arXiv. https://doi.org/10.48550/arXiv.2205.06175

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). Orb: An efficient alternative to SIFT or SURF. In *Proceedings of the IEEE International Conference on Computer Vision*, Barcelona, Spain (pp. 2571). https://doi.org/10.1109/ICCV.2011.6126544

Schendan, H. E., & Stern, C. E. (2007). Mental rotation and object categorization share a common network of prefrontal and dorsal and ventral regions of posterior cortex. *NeuroImage*, *35*(3), 1264–1277. https://doi.org/10.1016/j.neuroimage.2007.01.012

Schendan, H. E., & Stern, C. E. (2008). Where vision meets memory: Prefrontal-posterior networks for visual object constancy during categorization and recognition. *Cerebral Cortex (New York, NY 1991)*, *18*(7), 1695–1711. https://doi.org/10.1093/cercor/bhm197

Sinha, I., Webb, T. W., & Cohen, J. D. (2020). *A memory-augmented neural network model of abstract rule learning* (arXiv: 2012.07172). arXiv. https://doi.org/10.48550/arXiv.2012.07172

Soulières, I., Dawson, M., Samson, F., Barbeau, E. B., Sahyoun, C. P., Strangman, G. E., Zeffiro, T. A., & Mottron, L. (2009). Enhanced visual processing contributes to matrix reasoning in autism. *Human Brain Mapping*, *30*(12), 4082–4107. https://doi.org/10.1002/hbm.20831

Steenbrugge, X., Leroux, S., Verbelen, T., & Dhoedt, B. (2018). *Improving generalization for abstract reasoning tasks using disentangled feature representations* (arXiv: 1811.04784). arXiv. https://doi.org/10.48550/arXiv.1811.04784

Stone, B., & Day, M. C. (1981). A developmental study of the processes underlying solution of figural matrices. *Child Development*, *52*(1), 359–362. https://doi.org/10.2307/1129251

van Steenkiste, S., Locatello, F., Schmidhuber, J., & Bachem, O. (2020). *Are disentangled representations helpful for abstract visual reasoning?*. Advances in neural information processing systems, 32. https://doi.org/10.48550/arXiv.1905.12506

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems, 30*. https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

Veličković, P., & Blundell, C. (2021). Neural algorithmic reasoning. *Patterns*, *2*(7), 100273. https://doi.org/10.1016/j.patter.2021.100273

Vodegel Matzen, L. B. L., van der Molen, M. W., & Dudink, A. C. M. (1994). Error analysis of Raven test performance. *Personality and Individual Differences*, *16*(3), 433–445. https://doi.org/10.1016/0191-8869(94)90070-1

Wang, D., Jamnik, M., & Liò, P. (2020). *Extrapolatable relational reasoning with comparators in low-dimensional manifolds*. https://openreview.net/forum?id=A993YzEUKB7

Webb, T., Holyoak, K. J., & Lu, H. (2023). Emergent analogical reasoning in large language models. *Nature Human Behaviour*, *7*(9), Article 9. 1526–1541. https://doi.org/10.1038/s41562-023-01659-w

Wu, Y., Dong, H., Grosse, R. B., & Ba, J. (2020). *The scattering compositional learner: Discovering objects, attributes, relationships in analogical reasoning*. https://openreview.net/forum?id=2oci5kFXE0o&referrer=%5Bthe%20profile%20of%20Honghua%20Dong%5D(%2Fprofile%3Fid%3D~Honghua_Dong1)

Xu, K., Li, J., Zhang, M., Du, S. S., Kawarabayashi, K., & Jegelka, S. (2020). What can neural networks reason about? (*arXiv: 1905.13211)*. *arXiv*. https://doi.org/10.48550/arXiv.1905.13211

Xu, K., Zhang, M., Li, J., Du, S. S., Kawarabayashi, K., & Jegelka, S. (2021). How neural networks extrapolate: From feedforward to graph neural networks. *(arXiv: 2009.11848)*. *arXiv*. https://doi.org/10.48550/arXiv.2009.11848

Yang, Y., & Kunda, M. (2023). Computational models of solving Raven's progressive matrices: A comprehensive introduction. arXiv preprint (arXiv: 2302.04238). arXiv. https://doi.org/10.48550/arXiv.2302.04238

Yang, Y., Sanyal, D., Michelson, J., Ainooson, J., & Kunda, M. (2022). An end-to-end imagery-based modeling of solving geometric analogy problems. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, Toronto, Canada (Vol. 44, No. 44). https://escholarship.org/uc/item/9gh2885c

Zheng, K., Zha, Z., & Wei, W. (2019). Abstract reasoning with distracting features (arXiv: 1912.00569). arXiv. https://doi.org/10.48550/arXiv.1912.00569

Zikan, K. (1991). Technical note-least-squares image registration. *ORSA Journal on Computing*, *3*(2), 169–172. https://doi.org/10.1287/ijoc.3.2.169