Confident Reasoning on Raven's Progressive Matrices Tests

Keith McGreggor and Ashok Goel

Design & Intelligence Laboratory, School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA 30332, USA

keith.mcgreggor@gatech.edu, goel@cc.gatech.edu

Abstract

We report a novel approach to addressing the Raven's Progressive Matrices (RPM) tests, one based upon purely visual representations. Our technique introduces the calculation of confidence in an answer and the automatic adjustment of level of resolution if that confidence is insufficient. We first describe the nature of the visual analogies found on the RPM. We then exhibit our algorithm and work through a detailed example. Finally, we present the performance of our algorithm on the four major variants of the RPM tests, illustrating the impact of confidence. This is the first such account of any computational model against the entirety of the Raven's.

Introduction

The Raven's Progressive Matrices (RPM) test paradigm is intended to measure eductive ability, the ability to extract and process information from a novel situation (Raven, Raven, & Court, 2003). The problems from Raven's various tests are organized into sets. Each successive set is generally interpreted to be more difficult than the prior set. Some of the problem sets are 2x2 matrices of images with six possible answers; the remaining sets are 3x3 matrices of images with eight possible answers. The tests are purely visual: no verbal information accompanies the tests.

From Turing onward, researchers in AI have long had an affinity for challenging their systems with intelligence tests (e.g. Levesque, Davis, & Morgenstern, 2011), and the Raven's is no exception. Over the years, different computational accounts have proposed various representations and specific mechanisms for solving RPM problems. These we now briefly shall review.

Hunt (1974) gives a theoretical account of the information processing demands of certain problems from the Advanced Progressive Matrices (APM). He proposes two qualitatively different solution algorithms—"Gestalt," which uses visual operations on analogical representations, and "Analytic," which uses logical operations on conceptual representations.

Carpenter, Just, and Shell (1990) describe a computational model that simulates solving RPM problems using propositional representations. Their model is based on the traditional production system architecture, with a long-term memory containing a set of hand-authored productions and a working memory containing the current goals. Productions are based on the relations among the entities in a RPM problem.

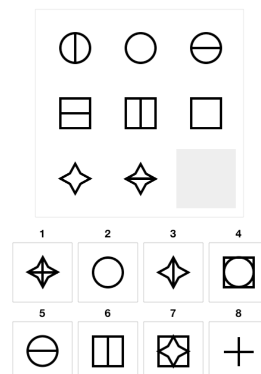


Figure 1. An example of a Raven's problem

Bringsjord and Schimanski (2003) used a theorem-prover to solve selected RPM problems stated in first-order logic.

Lovett, Forbus and Usher (2010) describe a model that extracts qualitative spatial representations from visually segmented representations of RPM problem inputs and then uses the analogy technique of structure mapping to find solutions and, where needed to achieve better analogies, to regroup or re-segment the initial inputs to form new problem representations.

Cirillo and Ström (2010) created a system for solving problems from the SPM that, like that of Lovett et al. (2010), takes as inputs vector graphics representations of test problems and automatically extracts hierarchical propositional problem representations. Then, like the work of Carpenter et al. (1990), the system draws from a set of predefined patterns, derived by the authors, to find the best-fit pattern for a given problem.

Kunda, McGreggor, and Goel (2011) have developed a model that operates directly on scanned image inputs from the test. This model uses operations based on mental imagery (rotations, translations, image composition, etc.) to induce image transformations between images in the problem matrix and then predicts an answer image based on the final induced transformation. McGreggor, Kunda, and Goel (2011) also report a model that employs fractal representations of the relationships between images.

Finally, Rasmussen and Eliasmith (2011) used a spiking neuron model to induce rules for solving RPM problems. Input images from the test were hand-coded into vectors of propositional attribute-value pairs, and then the spiking neuron model was used to derive transformations among these vectors and abstract over them to induce a general rule transformation for that particular problem.

The variety of approaches to solving RPM problems suggest that no one definitive account exists. Here, we develop a new method for addressing the RPM, based upon fractal representations. An important aspect of our method is that a desired confidence with which the problem is to be solved may be used as a method for automatically tuning the algorithm. In addition, we illustrate the application of our model against all of the available test suites of RPM problems, a first in the literature.

Ravens and Confidence

Let us illustrate our method for solving RPM problems. We shall use as an example the 3x3 matrix problem shown in Figure 1. The images and Java source code for this example may be found on our research group's website.

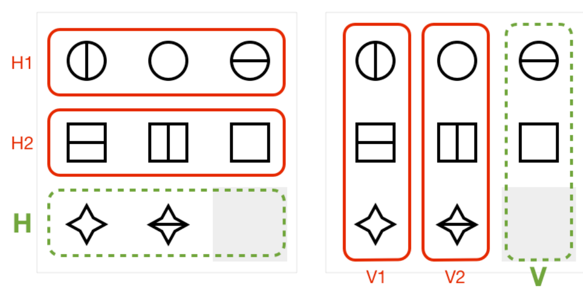


Figure 2. Simultaneous relationships

Simultaneous Relationships and Constraints

In any Raven's problem there exist simultaneous horizontal and vertical relationships which must be maintained. In Figure 2, we illustrate these relationships using our example problem. As shown, relationships H1 and H2 constrain relationship H, while relationships V1 and V2 constrain relationship V. While there may be other possible relationships suggested by this problem, we have chosen to focus on these particular relationships for clarity.

To solve a Raven's problem, one must select the image from the set of possible answers for which the similarity to each of the problem's relationships is maximal. For our example, this involves the calculation of a set of similarity values Θ_i for each answer A_i:

$$\Theta_i \leftarrow \{\, S(\, H1, H(A_i)\,),\ S(\, H2, H(A_i)\,),\ S(\, V1, V(A_i)\,),\ S(\, V2, V(A_i)\,)\,\}$$

where $H(A_i)$ and $V(A_i)$ denote the relationship formed when the answer image A_i is included. S(X,Y) is the Tversky featural similarity between two sets X and Y (Tversky, 1977):

$$S(X,Y) \leftarrow f(X \cap Y) \,/\, [\, f(X \cap Y) + \alpha f(X-Y) + \beta f(Y-X)\,]$$

Fractal Representation of Visual Relationships

We chose to use fractal representations here for their consistency under re-representation (McGreggor, 2013), and in particular for the mutual fractal representation, which expresses the relationship between sets of images.

In Figure 3, we illustrate how to construct a mutual fractal representation of the relationship H1.

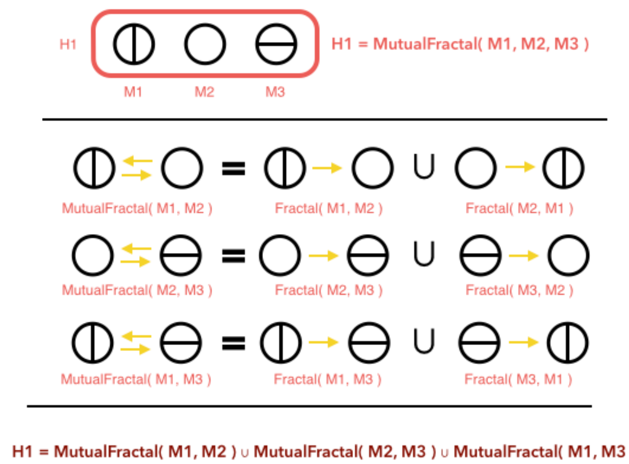


Figure 3. Mutual Fractal Representations

Confidence and Ambiguity

An answer to a Raven's problem may be found by choosing the one with the maximal featural similarity. But how confident is that answer? Given the variety of answer choices, even though an answer may be selected based on maximal similarity, how may that choice be contrasted with its peers as the designated answer?

We claim that the most probable answer would in a sense "stand apart" from the rest of the choices, and that distinction may be interpreted as a metric of confidence. Assuming a normal distribution, we may calculate a confidence interval based upon the standard deviation, and score each of these values along such a confidence scale. Thus, the problem of selecting the answer for a Raven's problem is transformed into a problem of distinguishing which of the possible choices is a statistical outlier.

The Confident Ravens Algorithm

To address Raven's problems, we developed the Confident Ravens algorithm. We present it here in pseudo-code form, in two parts: the preparatory stage and the execution stage.

Confident Ravens, Preparatory Stage

In the first stage of our Confident Ravens Algorithm, an image containing the entire problem is first segmented into its component images (the matrix of images, and the possible answers). Next, based upon the complexity of the matrix, the set of relationships to be evaluated is established. Then, a range of abstraction levels is determined. Throughout, we use MutualFractal() to indicate the mutual fractal representation of the input images (McGreggor & Goel, 2012).

Given an image P containing a Raven's problem, prepare to determine an answer with confidence.

PROBLEM SEGMENTATION

By examination, divide P into two images, one containing the matrix and the other containing the possible answers. Further divide the matrix image into an ordered set of either 3 or 8 matrix element images, for 2x2 or 3x3 matrices respectively. Likewise, divide the answer image into an ordered set of its constituent individual answer choices.

Let $M \leftarrow \{ m_1, m_2, \ldots \}$ be the set of matrix element images.
Let $C \leftarrow \{ c_1, c_2, c_3, \ldots \}$ be the set of answer choices.
Let η be an integer denoting the order of the matrix image (either 2 or 3, for 2x2 or 3x3 matrices respectively).

RELATIONSHIP DESIGNATIONS

Let R be a set of relationships, determined by the value of η as follows:
If $\eta = 2$:
  $R \leftarrow \{ H_1, V_1 \}$ where
  $H_1 \leftarrow$ MutualFractal(m_1, m_2)
  $V_1 \leftarrow$ MutualFractal(m_1, m_3)
Else: (because $\eta = 3$)
  $R \leftarrow \{ H_1, H_2, V_1, V_2 \}$ where
  $H_1 \leftarrow$ MutualFractal(m_1, m_2, m_3)
  $H_2 \leftarrow$ MutualFractal(m_4, m_5, m_6)
  $V_1 \leftarrow$ MutualFractal(m_1, m_4, m_7)
  $V_2 \leftarrow$ MutualFractal(m_2, m_5, m_8)

ABSTRACTION LEVEL PREPARATION

Let d be the largest dimension for any image in $M \cup C$.

Let $A := \{ a_1, a_2, \ldots \}$ represent an ordered range of abstraction values where
  $a_1 \leftarrow d$, and $a_i \leftarrow ½\, a_{i-1}$
  $\forall\, i,\ 2 \leq i \leq \text{floor}(\log2\ d)$ and $a_i \geq 2$

The values within A constitute the grid values to be used when partitioning the problem's images.

Algorithm 1. Confident Ravens Preparatory Stage

In the present implementation, the abstraction levels are determined to be a partitioning of the given images into gridded sections at a prescribed size and regularity.

Confident Ravens, Execution Stage

The algorithm concludes by calculating similarity values for each of the possible answer choices. It uses the deviation of these values from their mean to determine the confidence in the answers at each level.

Given M, C, R, A, and η as determined in the preparatory stage, determine an answer and its confidence.

Let E be a real number which represents the number of standard deviations beyond which a value's answer may be judged as "confident"

Let S(X,Y) be the Tversky similarity metric for sets X and Y

EXECUTION

For each abstraction $a \in A$:

- Re-represent each representation $r \in R$ according to abstraction a
- $S \leftarrow$ □
- For each answer image $c \in C$:
  - If $\eta = 2$:
    $H \leftarrow$ MutualFractal(m_3, c)
    $V \leftarrow$ MutualFractal(m_2, c)
    $\Theta \leftarrow \{ S(H_1, H), S(V_1, V) \}$
  - Else: (because $\eta = 3$)
    $H \leftarrow$ MutualFractal(m_7, m_8, c)
    $V \leftarrow$ MutualFractal(m_3, m_6, c)
    $\Theta \leftarrow \{ S(H_1, H), S(H_2, H),$
      $S(V_1, V), S(V_2, V) \}$
  - Calculate a single similarity metric from vector Θ:
    $t \leftarrow \sqrt{\Sigma\, \theta^2} \quad \forall\, \theta \in \Theta$
    $S \leftarrow S \cup \{ t \}$
- Set $\mu \leftarrow$ mean (S)
- Set $\sigma_\mu \leftarrow$ stdev (S) / $\sqrt{n}$
- Set $D \leftarrow \{ D_1, D_2, D_3, D_4, \ldots D_n \}$ where $D_i = (S_i - \mu) / \sigma_\mu$
- Generate the set $Z \leftarrow \{ Z_i \ldots \}\ \forall\, Z_i \in D$ and $Z_i > E$
- If $|Z| = 1$, return the answer image $c_i \in C$ which corresponds to Z_i
- otherwise there exists ambiguity, and further refinement must occur.

If no answer has been returned, then no answer may be given unambiguously.

Algorithm 2. Confident Ravens Execution Stage

Thus, for each level of abstraction, the relationships implied by the kind of Raven's problem (2x2 or 3x3) are re-represented into that partitioning. Then, for each of the candidate images, a potentially analogous relationship is determined for each of the existing relationships and a similarity value calculated. The vector of similarity values is reduced via a simple Euclidean distance formula to a single similarity. The balance of the algorithm, using the deviation from the mean of these similarities, continues through a variety of levels of abstraction, looking for an unambiguous answer that meets the specified confidence constraint.

The Example, Solved

Table 1 shows the results of running the Confident Ravens algorithm on the example problem, starting at an original gridded partitioning of 200x200 pixels (the maximal pixel dimension of the images), and then refining the partitioning down to a grid of 6x6 pixels, using a subdivision by half scheme, yielding 6 levels of abstraction.

Let us suppose that a confidence level of 95% is desired. The table gives the mean (μ), standard deviation (σ_μ), and number of features (f) for each level of abstraction (grid). The deviation and confidence for each candidate answer are given for each level of abstraction as well.

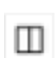
image	deviations & confidences					
	0.175 13.8%	-2.035 -95.8%	-1.861 -93.72%	0.698 51.47%	-0.760 -55.29%	-0.610 -45.79%
	-0.321 -25.17%	4.166 100%	2.783 99.46%	2.179 97.07%	0.681 50.4%	1.106 73.12%
	6.390 100%	3.484 99.95%	2.930 99.66%	4.487 100%	3.961 99.99%	4.100 100%
	0.495 37.97%	-3.384 -99.93%	-3.841 -99.99%	-4.848 -100%	-4.958 -100%	-5.454 -100%
	-1.741 -91.84%	-1.678 -90.67%	-2.148 -96.83%	-0.591 -44.56%	-2.825 -99.53%	-1.921 -94.52%
	-0.321 -25.17%	1.560 88.12%	2.444 98.55%	-1.361 -82.64%	0.896 62.96%	0.643 47.99%
	-1.741 -91.84%	0.254 20.02%	2.172 97.02%	-1.826 -93.22%	0.668 49.58%	0.213 16.85%
	-2.935 -99.67%	-2.366 -98.20%	-2.479 -98.68%	1.262 79.31%	2.338 98.06%	1.922 94.54%
grid	200	100	50	25	12	6
μ	0.589	0.31	0.432	0.69	0.872	0.915
σ_μ	0.031	0.019	0.028	0.015	0.007	0.005
f	378	1512	6048	24192	109242	436968

Table 1. Image Deviations and Confidences
Yellow indicates ambiguous results, red indicates that the result is unambiguous

The deviations presented in table 1 appear to suggest that if one starts at the very coarsest level of abstraction, the answer is apparent (image choice 3). Indeed, the confidence in that answer never dips below 99.66%.

We see evidence that operating with either too sparse a data set (at the coarsest) or with too homogeneous a data set (at the finest) may be problematic. The coarsest abstraction (200 pixel grid size) offers 378 features, whereas the finest abstraction (6 pixel grid size) offers more than 400,000 features for consideration.

The data in the table suggests the possibility of automatically detecting these boundary situations. We note that the average similarity measurement at the coarsest abstraction is 0.589, but then falls, at the next level of abstraction, to 0.310, only to thereafter generally increase. This constitutes further evidence for an emergent boundary for the maximum coarse abstraction.

We surmise that ambiguity exists for ranges of abstraction, only to vanish at some appropriate levels of abstraction, and then reemerges once those levels are surpassed. The example here offers evidence of such behavior, where there exists ambiguity at grid sizes 100, 50, 25, and 12, then the ambiguity vanishes for grid size 6. Though we omit the values in Table 1 for clarity of presentation, our calculations show that ambiguity reemerges for grid size 3. This suggests that there are discriminatory features within the images exist only at certain levels of abstraction.

Results

We have tested the Confident Ravens algorithm against the four primary variants of the RPM: the 60 problems of the Standard Progressive Matrices (SPM) test, the 48 problems of the Advanced Progressive Matrices (APM) test, the 36 problems of the Coloured Progressive Matrices (CPM) test, and the 60 problems of the SPM Plus test. Insofar as we know, this research represents the first published computational account of any model against the entire suite of the Raven Progressive Matrices.

To create inputs for the algorithm, each page from the various Raven test booklets were scanned, and the resulting greyscale images were rotated to roughly correct for page alignment issues. Then, the images were sliced up to create separate image files for each entry in the problem matrix and for each answer choice. These separate images were the inputs to the technique for each problem. No further image processing or cleanup was performed, despite the presence of numerous pixel-level artifacts introduced by the scanning and minor inter-problem image alignment issues. Additionally, each problem was solved independently: no information was carried over from problem to problem, nor from test variant to test variant.

The code used to conduct these tests was precisely the same code as used in the presented example, and is available for download from our lab website. The Raven test images as scanned, however, are copyrighted and thus are not available for download.

Abstractions, Metrics, and Calculations

The images associated with each problem, in general, had a maximum pixel dimension of between 150 and 250 pixels. We chose a partitioning scheme which started at the maximum dimension, then descended in steps of 10, until it reached a minimum size of no smaller than 4 pixels, yielding 14 to 22 levels of abstraction for each problem.

At each level of abstraction, we calculated the similarity value for each possible answer, as proscribed by the Confident Ravens algorithm. For those calculations, we used the Tversky contrast ratio formula (1977), and set α to 1.0 and β equal to 0.0, conforming to values used in the coincidence model by Bush and Mosteller (1953), yielding an asymmetric similarity metric preferential to the problem matrix's relationships. From those values, we calculated the mean and standard deviation, and then calculated the deviation and confidence for each answer. We made note of which answers provided a confidence above our chosen level, and whether for each abstraction level the answer was unambiguous or ambiguous, and if ambiguous, in what manner.

As we were exploring the advent and disappearance of ambiguity and the effect of confidence, we chose to allow the algorithm to run fully at all available levels of abstraction, rather than halting when an unambiguous answer was determined.

Performance on the SPM test: 54 of 60

On the Raven's Standard Progressive Matrices (SPM) test, the Confident Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 54 of the 60 problems. The number of problems with detected correct answers per set were 12 for set A, 10 for set B, 12 for set C, 8 for set D, and 12 for set E. Of the 54 problems where the correct answers detected, 22 problems were answered ambiguously.

Performance on the APM test: 43 of 48

On the Raven's Advanced Progressive Matrices (APM) test, the Confident Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 43 of the 48 problems. The number of problems with detected correct answers per set were 11 for set A, and 32 for set B. Of the 43 problems where the correct answers detected, 27 problems were answered ambiguously.

Performance on the CPM test: 35 of 36

On the Raven's Coloured Progressive Matrices (CPM) test, the Confident Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 35 of the 36 problems. The number of problems with detected correct answers per set were 12 for set A, 12 for set AB, and 11 for set B. Of the 35 problems where the correct answers detected, 5 problems were answered ambiguously.

Performance on the SPM Plus test: 58 of 60

On the Raven's SPM Plus test, the Confident Ravens algorithm detected the correct answer at a 95% or higher level of confidence on 58 of the 60 problems. The number of problems with detected correct answers per set were 12 for set A, 11 for set B, 12 for set C, 12 for set D, and 11 for set E. Of the 58 problems where the correct answers detected, 23 problems were answered ambiguously.

Confidence and Ambiguity, Revisited

We explored a range of confidence values for each test suite of problems, and illustrate these findings in Table 2.

Note that as confidence increases from 95% to 99.99%, the test scores decrease, but so too does the ambiguity. Analogously, as the confidence is relaxed from 95% down to 60%, test scores increase, but so too does ambiguity. By inspection, we note that there is a marked shift in the rate at which test scores and ambiguity change between 99.9% and 95%, suggesting that 95% confidence may be a reasonable choice.

confidence threshold	SPM 60		APM 48		CPM 36		SPMPlus 60	
	correct	ambiguous	correct	ambiguous	correct	ambiguous	correct	ambiguous
99.99%	41	1	28	1	24	0	44	2
99.9%	49	4	38	8	30	0	53	5
99%	53	14	42	16	33	1	58	14
95%	54	22	43	27	35	5	58	23
90%	55	29	45	31	36	9	59	32
80%	57	36	45	38	36	9	59	37
60%	58	42	47	45	36	14	60	45

Table 2. The Effect of Confidence on Score and Ambiguity

Our findings indicate that at 95% confidence, those problems which are answered correctly but ambiguously are vacillating almost in every case between two choices (out of an original 6 or 8 possible answers for the problem). This narrowing of choices suggests to us that ambiguity resolution might entail a closer examination of just those specific selections, via re-representation as afforded by the fractal representation, a change of representational framework, or a change of algorithm altogether.

Comparison to other computational models

As we noted in the introduction, there are other computational models which have been used on some or all problems of certain tests. However, all other computational accounts report scores when choosing a single answer per problem, and do not report at all the confidence with which their algorithms chose those answers. As such, our reported totals must be considered as a potential high score for Confident Ravens if the issues of ambiguity were to be sufficiently addressed.

Also as we noted earlier, this paper presents the first computational account of a model running against all four variants of the RPM. Other accounts generally report scores on the SPM or the APM, and no other account exists for scores on the SPM Plus.

Carpenter et al. (1990) report results of running two versions of their algorithm (FairRaven and BetterRaven) against a subset of the APM problems (34 of the 48 total). The subset of problems chosen by Carpenter et al. reflect those whose rules and representations were deemed as inferable by their production rule based system. They report that FairRaven achieves a score of 23 out of the 34, while BetterRaven achieves a score of 32 out of the 34.

Lovett et al (2007, 2010) report results from their computational model's approach to the Raven's SPM test. In each account, only a portion of the test was attempted, but Lovett et al project an overall score based on the performance of the attempted sections. The latest published account by Lovett et al (2010) reports a score of 44 out of 48 attempted problems from sets B through E of the SPM test, but does not offer a breakdown of this score by problem set. Lovett et al. (2010) project a score of 56 for the entire test, based on human normative data indicating a probable score of 12 on set A given their model's performance on the attempted sets.

Cirillo and Ström (2010) report that their system was tested against Sets C through E of the SPM and solved 8, 10, and 10 problems, respectively, for a score of 28 out of the 36 problems attempted. Though unattempted, they predict that their system would score 19 on the APM (a prediction of 7 on set A, and 12 on set B).

Kunda et al. (2013) reports the results of running their ASTI algorithms against all of the problems on both the SPM and the APM tests, with a detailed breakdown of scoring per test. They report a score of 35 for the SPM test, and a score of 21 on the APM test. In her dissertation, Kunda (2013) reports a score of 50 for the SPM, 18 for the APM, and 35 on the CPM.

McGreggor et al. (2011) contains an account of running a preliminary version of their algorithm using fractal representations against all problems on the SPM. They report a score of 32 on the SPM, 11 on set A, 7 on set B, 5 on set C, 7 on set D, and 2 on set E. They report that these results were consistent with human test taker norms. Kunda et al. (2012) offers a summation of the fractal algorithm as applied to the APM, with a score of 38, 12 on set A, and 26 on set B.

The work we present here represents a substantial theoretical extension as well as a significant performance improvement upon these earlier fractal results.

Conclusion

In this paper, we have presented a comprehensive account of our efforts to address the entire Raven's Progressive Matrices tests using purely visual representations, the first such account in the literature. We developed the Confident Ravens algorithm, a computational model which uses features derived from fractal representations to calculate Tversky similarities between relationships in the test problem matrices and candidate answers, and which uses levels of abstraction, through re-representing the visual representation at differing resolutions, to determine overall confidence in the selection of an answer. Finally, we presented a comparison of the results of running the Confident Ravens algorithm to all available published accounts, and showed that the Confident Ravens algorithm's performance at detecting the correct answer is on par with those accounts.

The claim that we present throughout these results, however, is that a computational model may provide both an answer as well as a characterization of the confidence with which the answer is given. Moreover, we have shown that insufficient confidence in a selected answer may be used by that computational model to force a reconsideration of a problem, through re-representation, representational shift, or algorithm change. Thus, we suggest that confidence is hereby well-established as a motivating factor for reasoning, and as a potential drive for an intelligent agent.

Acknowledgments

This work has benefited from many discussions with our colleague Maithilee Kunda and the members of the Design and Intelligence Lab at Georgia Institute of Technology. We thank the US National Science Foundation for its support of this work through IIS Grant #1116541, entitled "Addressing visual analogy problems on the Raven's intelligence test."

References

Barnsley, M., and Hurd, L. 1992. *Fractal Image Compression*. Boston, MA: A.K. Peters.

Bringsjord, S., and Schimanski, B. 2003. What is artificial intelligence? Psychometric AI as an answer. *International Joint Conference on Artificial Intelligence,* 18: 887–893.

Bush, R.R., and Mosteller, F. 1953. A Stochastic Model with Applications to Learning. *The Annals of Mathematical Statistics*, 24(4): 559-585.

Carpenter, P., Just, M., and 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.

Cirillo, S., and Ström, V. 2010. An anthropomorphic solver for Raven's Progressive Matrices (No. 2010:096). Goteborg, Sweden: Chalmers University of Technology.

Haugeland, J. ed. 1981. *Mind Design: Philosophy, Psychology and Artificial Intelligence*. MIT Press.

Hofstadter, D., and Fluid Analogies Research Group. eds. 1995. *Fluid concepts & creative analogies: Computer models of the fundamental mechanisms of thought.* New York: Basic Books.

Hunt, E. 1974. Quote the raven? Nevermore! In L. W. Gregg ed., *Knowledge and Cognition* (pp. 129–158). Hillsdale, NJ: Erlbaum.

Kunda, M. 2013. Visual Problem Solving in Autism, Psychometrics, and AI: The Case of the Raven's Progressive Matrices Intelligence Test. Doctoral dissertation, Georgia Institute of Technology.

Kunda, M., McGreggor, K. and Goel, A. 2011. Two Visual Strategies for Solving the Raven's Progressive Matrices Intelligence Test. *Proceedings of the 25th AAAI Conference on Artificial Intelligence*.

Kunda, M., McGreggor, K. and Goel, A. 2012. Reasoning on the Raven's Advanced Progressive Matrices Test with Iconic Visual Representations. *Proceedings of the 34th Annual Meeting of the Cognitive Science Society*, Sapporo, Japan.

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, pp. 47-66.

Levesque, H. J., Davis, E., & Morgenstern, L. 2011. The Winograd Schema Challenge. *In AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning.*

Lovett, A. Forbus, K., and Usher, J. 2007. Analogy with qualitative spatial representations can simulate solving Raven's Progressive Matrices. *Proceedings of the 29th Annual Conference of the Cognitive Science Society.*

Lovett, A., Forbus, K., and Usher, J. 2010. A structure-mapping model of Raven's Progressive Matrices. *Proceedings of the 32nd Annual Conference of the Cognitive Science Society.*

Mandelbrot, B. 1982. *The fractal geometry of nature.* San Francisco: W.H. Freeman.

McGreggor, K. (2013). Fractal Reasoning. Doctoral dissertation, Georgia Institute of Technology.

McGreggor, K., Kunda, M., & Goel, A. K. (2011). Fractal Analogies: Preliminary Results from the Raven's Test of Intelligence. In Proceedings of the Second International Conference on Computational Creativity (ICCC), Mexico City. pp. 69-71.

McGreggor, K., & Goel, A. 2011. Finding the odd one out: a fractal analogical approach. *Proceedings of the 8th ACM conference on Creativity and cognition* (pp. 289-298). ACM.

McGreggor, K., and Goel, A. 2012. Fractal analogies for general intelligence. *Artificial General Intelligence.* Springer Berlin Heidelberg, 177-188.

Raven, J., Raven, J. C., and Court, J. H. 2003. *Manual for Raven's Progressive Matrices and Vocabulary Scales.* San Antonio, TX: Harcourt Assessment.

Rasmussen, D., and Eliasmith, C. 2011. A neural model of rule generation in inductive reasoning. *Topics in Cognitive Science,* 3(1), 140-153.

Tversky, A. 1977. Features of similarity. *Psychological Review,* 84(4), 327-352.