Reflection Project 1: Yet Another Production System for solving 2x2 Ravens Progressive Matrices

Manikandan Ravikiran
mravikiran@gatech.edu

1 INTRODUCTION

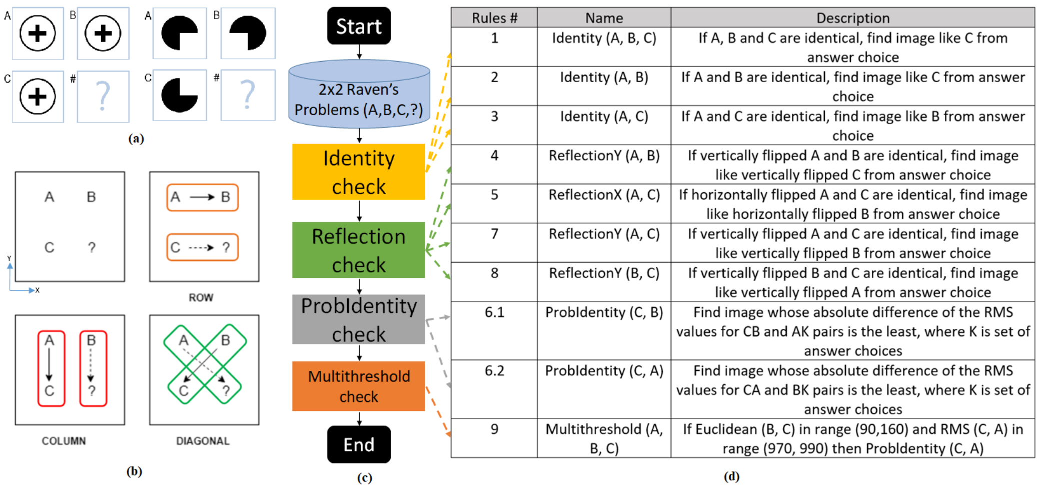
In this report, a production system designed and implemented by drawing perspectives from (Kunda, McGreggor, and Goel, 2009) is presented as a solution to 2x2 Raven's Progressive Matrices (RPM) (Raven, 1962). The developed agent solves 12/12 and 10/12 across Basic and Test/Challenge/Raven's sets, respectively. The report explains details on error analysis, efficiency, generality and cognitive connection and concludes with some implications on future work.

2 SOLUTION PROPOSAL & ALGORITHM

Knowledge Representation, Agent Reasoning and Design: In this work, pixel based visual representations for images (Figure 1a) with Affine Symbolic reasoning (Kunda, McGreggor, and Goel, 2009) was used. Overall, a production system with various rules is incrementally designed and developed, solving one or more RPM at a time (Figure 1c). These rules identifies the relationship **T** across rows/columns/diagonal of an 2x2 RPM problem (Figure 1b). Such a **T** is then applied to rows/columns/diagonal with the empty entry to generate a predicted value for the missing image which are then compared to answer choices via image similarity metric to find the solution.

Image Similarity Metrics: Two different image similarity metrics namely **Root Mean Square (RMS)** distance and **Euclidean Distance (ED)** are used with **thresholds** RPM-τ & ED-θ respectively. More details are explained across sections 3.1-3.6 and consolidated details are presented in appendix sections 6.2 and 6.3.

Algorithm: The final developed algorithm and process flow are as shown in Figure 1d & Figure 1c respectively, which consists of nine production rules a.k.a ***if-else*** cases which encompasses three broad categories relationships, namely ***Identity***, ***Reflection***, ***ProbIdentity*** & ***Multithreshold*** respectively. Each of the rules are described in Figure 1d and is developed incrementally by testing on the auto grader. During processing, each of rules are executed according to the numbered sequence **(Rules #)**. Whenever the input RPM violates a given rule, the agent



Rules #	Name	Description
1	Identity (A, B, C)	If A, B and C are identical, find image like C from answer choice
2	Identity (A, B)	If A and B are identical, find image like C from answer choice
3	Identity (A, C)	If A and C are identical, find image like B from answer choice
4	ReflectionY (A, B)	If vertically flipped A and B are identical, find image like vertically flipped C from answer choice
5	ReflectionX (A, C)	If horizontally flipped A and C are identical, find image like horizontally flipped B from answer choice
7	ReflectionY (A, C)	If vertically flipped A and C are identical, find image like vertically flipped B from answer choice
8	ReflectionY (B, C)	If vertically flipped B and C are identical, find image like vertically flipped A from answer choice
6.1	ProbIdentity (C, B)	Find image whose absolute difference of the RMS values for CB and AK pairs is the least, where K is set of answer choices
6.2	ProbIdentity (C, A)	Find image whose absolute difference of the RMS values for CA and BK pairs is the least, where K is set of answer choices
9	Multithreshold (A, B, C)	If Euclidean (B, C) in range (90,160) and RMS (C, A) in range (970, 990) then ProbIdentity (C, A)

Figure 1—Sample ravens matrices (a), Affine Relationships (b) Process flow (c) and production rules (d).

moves onto the next rule, otherwise computes the result and outputs the corresponding answer choice.

Performance Evaluation Metrics: Performance of the agent is accessed using accuracy, efficiency & generality metrics. Also, errors are categorized as **Wrong Principle (WP)** and **Incomplete Correlate (IC)** (Kunda et al., 2016) highlighted in blue and orange across Tables 1-5. More in-depth details are in sections 6.4 & 6.5.

3 EXPERIMENTAL RUNS AND DISCUSSION

This section presents and analyzes, various submissions tested on the auto grader. Each submission's description begins with selecting one or more RPM's, followed by its analysis and solution development, ending with cognitive connection, errors and improvement proposals. In this section, basic problems and challenge problems are represented by **BP** and **CP** respectively. Also the rules in Figure 1d is referred throughout.

3.1 Submission-1: Solving BP-1 to BP-5 (2019-08-21 17:54:52 UTC)

Intuition: To begin with, **manual analysis of BP-1 to BP-5 was carried out**. As seen in Figure 2, these problems satisfy identity and reflection relationships. Hence, **rules 1-5** (Figure 1d) involving **Identity** and **Reflection** was added to the agent, with an option to skip when none of the rules solve the problem. This submission used RMS-τ of 965. The **results** so obtained are in Table 1 with total execution time of 5.9 secs.

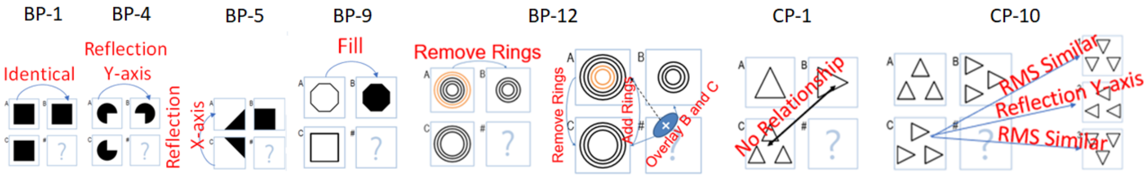


Figure 2—Examples of Patterns and relationship across BP's and CP's.

Cognitive Connection: Human's incrementally solve RPM's using variety of approaches (Carpenter, Just, and Shell, 1990). The agent designed, naturally simulates human thinking by incrementally analyzing various relationships such as Identity and Reflection, present in the images to find the solution. However, unlike humans the agent doesn't have any meta-cognition to gauge it's certainty about proposed solutions, and whether it should return an answer or skip the problem.

Table 1—Results from submission-1. ✓, ✗ & ✈ indicates correct, incorrect and skipped answers.

	B-01	**B-02**	**B-03**	**B-04**	**B-05**	**B-06**	**B-07**	**B-08**	**B-09**	**B-10**	**B-11**	**B-12**	**Accuracy**
Basic	✓	✓	✓	✓	✓	✈	✓	✓	✈	✈	✓	✈	8/12
Challenge	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✓	✈	1/12

	Accuracy
Ravens	6/12
Test	7/12

Error Analysis: While, the agent aces the 7/12 barrier on Basic and Test, due to lack of any rules to understand RPM's that follow fill (BP-9 in Figure 2) and add/removal patterns (BP-12 in Figure 2), multiple BPs are skipped. Further, the agent only solved CP-11, which adheres to reflection property, suggesting better rules are needed to handle CP's.

Improvement Proposal: The limitations of the agent could be improved considerably by adding new rule to tackle skipped problems like BP-12.

3.2 Submission-2: Solving BP-12 (2019-08-22 15:20:03 UTC)

Improvement Intuition: Based on error analysis from section 3.1, a rule was included to tackle BP-12. As seen in BP-12 (Figure 2), affine transformation is of limited use, as circle is symmetric. Multiple relationships are observed, where ***removal of two outer rings in A produces B*** or ***removal of two inner rings produces C*** or ***diagonally overlaying B and C produces A***.

Rule 6.1: Among multiple pattern's previous described, in this submission, diagonal relationship is considered and rule-6.1 (Figure 1d) i.e. ***ProbIdentity(B,C)*** is introduced, which selects the answer choice that satisfies *RMS(B,C)≈RMS(A,D)*[1]. We still skip problems, when none of the rules fail to come through. So the agent has **rules 1-5 & 6.1** with RMS-τ of 965 to obtain results as shown in Table 2 with total agent execution time of 6.04 secs.

Cognitive Connection: The agents solving process, after introducing rule 6.1 still seems to mimic the way a human would think to some extent. Especially ***"some extent"***, because while identity and reflection follows human thinking, mimicking removal and overlay of patterns through similarity based metric is different from human reasoning.

Error Analysis: Problems BP-12 was solved, without any other errors in BP's. Additionally CP-2 and CP-12 was also addressed. However in CP, the rule also produced 7 wrong answers. These errors (Figure 2) are i) due to lack of relationship between B and C (Ex: CP-1) and ii) Similarity in answer choices (Ex: CP-10).

Table 2—Results from submission-2. ✓, ✗ & ✈ indicates correct, incorrect and skipped answers. WP & IC are higlighted.

	B-01	B-02	B-03	B-04	B-05	B-06	B-07	B-08	B-09	B-10	B-11	B-12	Accuracy
Basic	✓	✓	✓	✓	✓	✈	✓	✓	✗	✗	✓	✓	9/12
Challenge	✗	✓	✗	✗	✗	✈	✈	✗	✗	✗	✓	✓	3/12

	Accuracy
Ravens	6/12
Test	7/12

Improvement Proposal: Investigating skipped problem BP-6 shows that C is reflection of A along Y Axis (Figure 3). Also manual analysis of CP-9 (Figure 3) shows that, relationship is more prevalent between A & C, rather than the diagonals. For CP-2 (Figure 3), we can see that B and C are reflection of one another. Modifications to accommodate previously observations should improve the results across CP set. Each of these improvements are addressed in sections 3.3-3.5

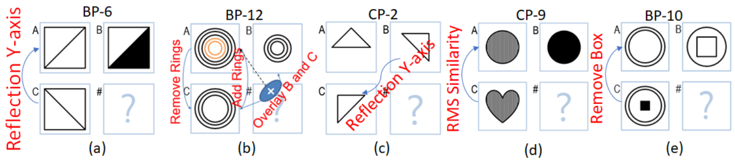


Figure 3—Patterns and relationship in BP-6 (a), BP-12 (b), CP-2 (c) & CP-8 (d).

[1] See section 6.6 for more detailed intuition and derivation of the formulae.

3.3 Submission 3: Solving BP-6 (2019-08-26 15:56:58 UTC)

Improvement Intuition: Based on error analysis from submission-2, for BP-6 we see C is reflection of A along Y axis (Figure 3), so rule-7 i.e. ***ReflectionX(A,C)*** was added to the agent with an RMS-τ of 967 and problem skipping was removed. Totally for submission-3, the agent had **rules 1-5, 6.1 and 7** respectively.

Performance: The agents completes all problems in is 5.95 secs to obtain results shown in Table 3.

Table 3—Results obtained from submission-3.✓ & ✗ indicates correct and incorrect answers. WP & IC are higlighted

	B-01	B-02	B-03	B-04	B-05	B-06	B-07	B-08	B-09	B-10	B-11	B-12	Accuracy
Basic	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓	10/12
Challenge	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	3/12

	Accuracy
Ravens	6/12
Test	7/12

Cognitive Connection and Error Analysis: No changes in cognitive connection since section 3.2 and no error analysis was done in this submission. Instead, solution to BP-10 was developed and submission-4 (Section 3.4) was executed.

3.4 Submission-4: Solving BP-10 (2019-08-28 16:07:33 UTC)

Improvement Intuition and Rule: Based on error analysis from submission-2, we can see that for BP-10 (Figure 3), that A and C are similar, where *C-(shaded box)=A or A+(shaded box)=C i.e A≈C*. By affine symbolic reasoning we have *(A-C)≈(B-D)*. Further, BP-12 also agrees with this observation, as such rather than rule-6.1 a.k.a ***ProbIdentity(B,C)*** relationship, rule-6.2 i.e. ***ProbIdentity(A,C)*** that validates if *RMS(A,C)≈RMS(B,D)*[2] was introduced in the agent with RMS-τ of 967. Overall the agent included **rules 1-5, 6.2 & 7**.

Table 4—Results obtained from submission-4.✓ & ✗ indicates correct and incorrect answers. WP & IC are higlighted.

	B-01	B-02	B-03	B-04	B-05	B-06	B-07	B-08	B-09	B-10	B-11	B-12	Accuracy
Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12/12
Challenge	✓	✗	✓	✗	✗	✗	✓	✓	✓	✗	✓	✓	7/12

	Accuracy
Ravens	7/12
Test	8/12

Performance: The agent solves BP-10 & 12. Also this rule solves BP-9 and generalizes to solve CP-1,3,7,8 & 9 leading to results in Table 4 with total execution time of 6 secs.

[2] *(A-C)≈(B-D)* → *RMS(A,C)≈RMS(B,D)*, is valid because RMS calculation includes pixel wise image subtraction. This is similar to submission 3 and 2.

Cognitive Connection: At this point, the agent is more of a production system for solving BP's, where the long-term memory consists of seven production rules that capture the relationship and solves the RPM's. There is a natural sync between production systems and human problem solving (Axten, 1973) and also the RPM's (Carpenter, Just, and Shell, 1990). However, there is no rule induction, that is designed as part of the agent which is typical in humans, where the adaptation of rule for newer unseen problem is inherent.

Error Analysis & Proposed Improvement: Inline with submission 3.4 no error analysis was done and instead a solution to solve CP-2 was designed and submission 3.5 was executed. Detailed errors are in Table 4.

3.5 Submission-5 Solving CP-2 (2019-09-04 09:35:11 UTC)

Improvement and Rules: Again from error analysis of submission 2, we see (Figure 3) that CP-2 adheres to reflection property where B and C are reflection of one another. To accommodate this, rule-8 i.e. ***ReflectionY(B,C)*** was introduced to the agent with RMS-τ of 965. Overall the agent included **rules 1-5, 6.2, 7 & 8**.

Table 5—Results obtained from submission-5.✓& ✗ indicates correct and incorrect answers. WP & IC are highlighted.

	B-01	B-02	B-03	B-04	B-05	B-06	B-07	B-08	B-09	B-10	B-11	B-12	Accuracy
Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12/12
Challenge	✓	✓	✓	✗	✗	✗	✓	✓	✓	✗	✓	✓	8/12

	Accuracy
Ravens	7/12
Test	8/12

Performance: The agent solves CP-2, no changes is seen in results of Basic, Test and Raven's compared to submission-4. The agent achieves results as shown in Table 5, with time efficiency of 6.01 secs.

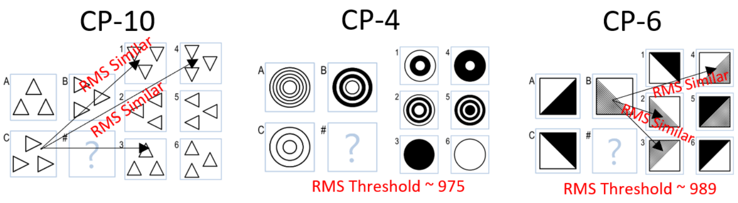


Figure 4—Relationships in erroneous challenge problems.

Error Analysis: Table 5 shows that CP's 4,5,6 and 10 are still erroneous. Manual analysis was carried out and following intuitions were obtained.

- **Common Similarity:** For Problem CP-10 (Figure 4), behavior of similar RMS values (1000-1010) across multiple answer choices can be seen.

- **Threshold:** Previously for submission 1-5, an RMS-τ of 965 was used. However, such a threshold is not valid for CP's 4-6, due to changes in patterns. In fact, analysis reveals that there is no single RMS-τ, that's usable across all problems. For example, in BP's the optimal value was 965. But for, CP's it couldn't be fixed at all, as it varied across the problems. For example, for CP-4 the RMS-τ for correct answer was 975 and for CP-5,6 the RMS-τ was around 989.
- **Alternative Similarity Metrics:** The above mentioned problem also persists across other similarity metric like Euclidean distances. However, experiments with Euclidean Distance as similarity metric & the rules 1-5,6.1,6.2,7,8 shows a unique property ***where it solved CP's (5-6) using 6.1 and gave wrong results to CP's (8-9) using 6.2***.

Improvement Proposal: Since Euclidean distance solves CP's (5-6) using rule 6.1 and RMS similarity solves CP's (8-9) using rule 6.2, combining them both would improve the overall results.

3.6 Submission-6: Solving CP-5 and CP-6 (2019-09-11 15:51:41 UTC)

Improvement Intuition: Based on observations and proposed improvement from submission-5, rule-9 a.k.a ***Multithreshold(A,B,C)*** was introduced which checks *ED(B,C)* in range of 90-160 and *RMS(C,A)* in range of 970-990. The agent checks the validity of the proposed rule on the RPM & if true then it executes rule 6.2 otherwise executes rules 6.1. As such, for this submission **rules 1-5,7,8,9 are included, where rule 9 in turn incorporates rules 6.1 and 6.2.**

Table 6—Results obtained from submission-6. ✓& ✗ indicates correct and incorrect answers. WP & IC are highlighted.

	B-01	B-02	B-03	B-04	B-05	B-06	B-07	B-08	B-09	B-10	B-11	B-12	Accuracy
Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12/12
Challenge	✓	✓	✓	✓	✓	✗	✓	✓	✓	✗	✓	✓	10/12

	Accuracy
Ravens	10/12
Test	10/12

Performance: This produces results of 12/12 in Basic and 10/12 in rest of the sets (Table 6), by solving CP's 4,5,8 and 9, with total execution time of 5.91 secs.

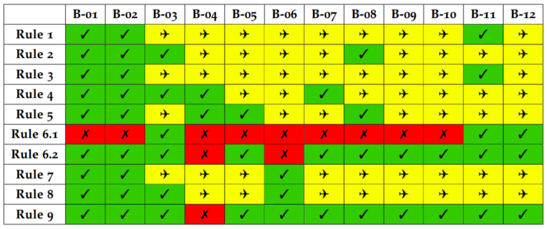
Cognitive Connection: The agent after introducing rule-9, is still a production system behaving like human. Additionally, it also shows behavior of analysis of given problem using multiple strategies. Human's while solving problems, try multiple hypothesis individually or in combination to solve a given problem, which is seen in the agent after addition of ***Multithreshold*** rule.

Errors and Possible Improvements: Solving CP-10 (Figure 4), requires analysis of orientation of pixels in addition to similarity, as all the answer choices produces same similarity values and CP-4 requires, analysis of alternative fill patterns.

4 CONCLUSION

Efficiency and Generality: Section 3.1-3.6 incrementally presents various submissions and modifications done to improve results across all the test. The developed agent achieve 12/12 on Basic and 10/12 on Test/Challenge/Raven's sets.

Table 7—Coverage and Generality of each developed rule on BP's



	B-01	B-02	B-03	B-04	B-05	B-06	B-07	B-08	B-09	B-10	B-11	B-12
Rule 1	✓	✓	✈	✈	✈	✈	✈	✈	✈	✈	✓	✈
Rule 2	✓	✓	✓	✈	✈	✈	✈	✓	✈	✈	✈	✈
Rule 3	✓	✓	✈	✈	✈	✈	✈	✈	✈	✈	✓	✈
Rule 4	✓	✓	✓	✓	✈	✈	✓	✈	✈	✈	✈	✈
Rule 5	✓	✓	✈	✓	✓	✈	✈	✓	✈	✈	✈	✈
Rule 6.1	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✓	✓
Rule 6.2	✓	✓	✓	✗	✓	✗	✓	✓	✓	✓	✓	✓
Rule 7	✓	✓	✈	✈	✈	✓	✈	✈	✈	✈	✈	✈
Rule 8	✓	✓	✓	✈	✈	✓	✈	✈	✈	✈	✈	✈
Rule 9	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓

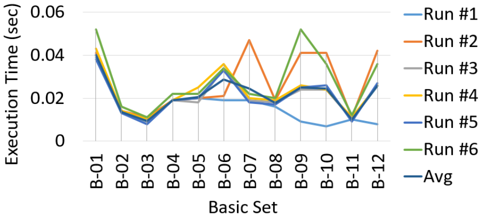


Figure 5—Efficiency of the developed production system on BP's.

Although, the agent used limited number of rules to solve the problem, these rules were quite general as the agent was able to address all of the problems in the basic set and most of the problem in the challenge set. Table 7 shows the generalization of each rule, across the basic set problems. As we can see, incrementally adding new rules, increases overall coverage of the system, more specifically the final ***Multithreshold*** alone covers 11 basic problems. Carefully examination shows that this rule is a general formulation of affine symbolic reasoning. Also as the problem complexity increases the agent consumes more to time for solving it (Figure 5), the final agent consumes 5.9 secs to solve all the problems.

Human Cognition and AI: The agent design simulates human thinking, especially the process of solving the RPM. Normally, humans start by **discerning various relationships between the images in each problem and then apply and adapt these to solve the problems incrementally** by observing each problem. Similarly the agent is architected like a production system, where it incrementally applies various relationships such as identity, reflection etc. to solve the problems.

Human's while solving RPM's typically address it in negligible time and **as problems become harder, the time consumption also increases**. Such a behavior is visible in the developed agent as well (Figure 10b), where agents spends lesser time on BP-(1,2,3) and more time on BP-(9,10,12). Even though agent is not as fast as human's solve RPM's, the pattern of time consumption is very similar.

Another aspects where agent is similar to human, involves **adapting to cases of problems**. Generally human's try to adapt their existing knowledge to newer problem via some form of reasoning, The agent through ***Multithresholding*** rule tries to work across multiple possible variation of same pattern. However, unlike human's there is no implicit rule induction.

The **design includes concepts learned during the class**, where the system designed is a **production system** with series of rules with **case-based reasoning** where multiple different thresholds are devised using heuristics to adapt for newer problems. However, the **errors made by agent are contradictory to human** testing, where most errors made by humans are repetition type (Kunda et al., 2016), while the **agent makes mostly wrong principle type errors**.

Further designing the agent based on visual representation, closely relates the agent to human's (Soulières et al., 2009), especially with **human relying on image relationship to solve the problems**.

Design Rectifications and Improvements: With availability of unlimited time and resources, following are the possible changes that can be done, to achieve more accurate results even more efficiently.

- **Threshold:** At the moment, multiple threshold's are selected manually to obtain the best result. Instead of this, an approach that maximizes the similarity or difference measure could be explored.
- **Avoid Reflection Check:** Reflection is a costly computational operation. Instead a similarity metric, that's produces same or very similar values for reflection and non-reflected images could be used.
- **Coreset analysis:** Currently, we can see that rule-9 alone covers all the BP's and half of CP's (Section 6.5), hence overall efficiency could be improved by selecting core sets of rules and optimizing the common components across them.
- **Analysis of Generalization:** In submissions 4 and 6, we can see that the rule generalizes well across multiple problems, reason for such a behaviour is still to be analyzed.

5 REFERENCES

[1] Axten, Nick (1973). “Human Problem Solving”. In: *Contemporary Sociology* 2.2, pp. 169–170. ISSN: 00943061, 19398638. URL: http://www.jstor.org/stable/2063712.

[2] Carpenter, Patricia, Just, Marcel Adam, and Shell, Peter (1990). “What one intelligence test measures: a theoretical account of the processing in the Raven Progressive Matrices Test.” In: *Psychological review* 97 3, pp. 404–31.

[3] Kunda, Maithilee, McGreggor, Keith, and Goel, Ashok K. (2009). “Addressing the Raven’s Progressive Matrices Test of "General" Intelligence”. In: *AAAI Fall Symposium: Multi-Representational Architectures for Human-Level Intelligence*.

[4] Kunda, Maithilee, Soulières, Isabelle, Rozga, Agata, and Goel, Ashok K. (2016). “Error patterns on the Raven’s Standard Progressive Matrices Test”. In: *Intelligence* 59, pp. 181–198. ISSN: 0160-2896. DOI: https://doi.org/10.1016/j.intell.2016.09.004. URL: http://www.sciencedirect.com/science/article/pii/S0160289616300149.

[5] Raven, John C. (1962). “Manual for Raven’s progressive matrices and vocabulary scales”. In: San Antonio, Texas: Pearson.

[6] Soulières, Isabelle, Dawson, Michelle, Samson, Fabienne, Barbeau, Elise Brochu, Sahyoun, Chérif P., Strangman, Gary, Zeffiro, Thomas A., and Mottron, Laurent (2009). “Enhanced visual processing contributes to matrix reasoning in autism.” In: *Human brain mapping* 30 12, pp. 4082–107.

6 APPENDICES

6.1 Representation and Reasoning

Knowledge Representation: Pixel based **visual representations** for images are used (see figure 1a), based on intuitions from (Carpenter, Just, and Shell, 1990) which suggest that pairwise spatial relationships between the problem are exploited during human problem solving process, which is represented as structural correspondence between the problem input images.

Reasoning: For submission 1-5 Affine Symbolic reasoning is used, where RPM problem is viewed as a sequence of images, where some affine transformation T can transform one image into a corresponding adjacent image (Kunda, McGreggor, and Goel, 2009).

6.2 Image Similarity Metrics

Root Mean Square (RMS): To get a measure of how similar two images are, root-mean-square (RMS) value of the difference between the images are calculated. If the images are exactly identical, this value is zero. The following function uses the difference function, and then calculates the RMS value from the histogram of the resulting image. Given two images X and Y of size WxH the RMS is calculated as

$$RMS = \sqrt{\frac{1}{WxH} \Sigma_{i=1}^{n} \left(X_i - Y_i \right)^2} \tag{1}$$

Euclidean Distance: Given two binarized images X and Y of size WxH, Euclidean Distance is calculated as

$$ED = \sqrt{\Sigma_{i=1}^{n} \left(X_i - Y_i \right)^2} \tag{2}$$

If the images are exactly identical, this value is zero, else the similarity is decided using the threshold θ. In this work, only rule 9 uses both Euclidean and RMS metrics, rest of the rules only uses RMS for image similarity computation.

6.3 Threshold Selection

Since the images are not exactly aligned, for comparing the images, this work uses images similarity metrics RMS and ED, where given a distance value of RMS or ED its compared with a threshold RMS-τ and ED-θ, if the distance is less than thresholds, the images are similar and vice versa.

Optimal threshold selection, is very important for the overall solution process, In this work, the threshold values are empirically setup to obtain the best performance.

For submission 1-4, 965 was the optimal threshold. However, from runs 5 on wards, using single threshold was difficult, hence multiple values in ranges was used. The details of various threshold used for each runs are as shown in Table 7. Please note that only rule 9 uses both euclidean and RMS metrics, rest of the rules only uses RMS.

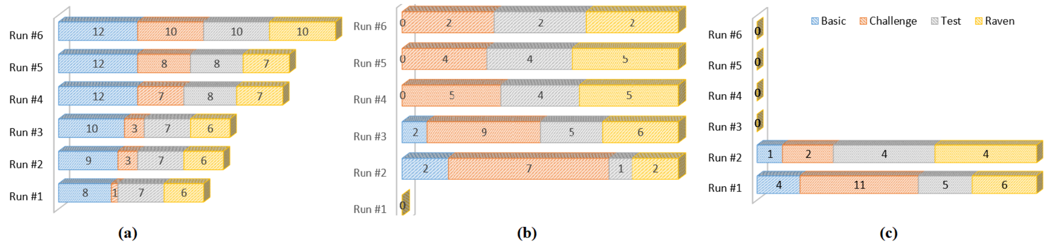
Table 7—Threshold values used for comparing image similarity across various rules.

	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6.1	Rule 6.2	Rule 7	Rule 8	Rule 9
Run 1	965	965	965	965	965					
Run 2	965	965	965	965	965	965				
Run 3	965	965	965	965	965	965		967		
Run 4	965	965	965	965	965		965	967		
Run 5	965	965	965	965	965		965	967	965	
Run 6	965	965	965	965	965			967	965	ED =(90,160) RMS=(970,990)

6.4 Error Metrics

The details of each of the error metrics is as explained below.

Accuracy: We use **precision** as the accuracy metric, which computes fraction of problems correctly answered by the agent. The consolidates results across all the runs are as shown in Figure 6 below.



(a) (b) (c)

Figure 5—Consolidated results of Correct (a), Incorrect (b) and Skipped problem across submissions

Efficiency: Efficiency is computed as time consumed in sec to solve a given problem i.e time consumed to execute the *Agent()* call for a given problem. Previously in Figure 5 efficiency was shown for Basic Problems, where we could see that efficiency changed with the complexity of the problem. Figure shows details of efficiency over challenge set. Again the trend is consistent with basic set, where the agent takes more time to solve complex problems and vice versa.

Generality: Generality is evaluated as the fraction of problems, that a given rule can cover in the absence of other rules. The generality of developed production system on basic problems are previously shown in Table 7. The generality over challenge set is as shown in Table 8.

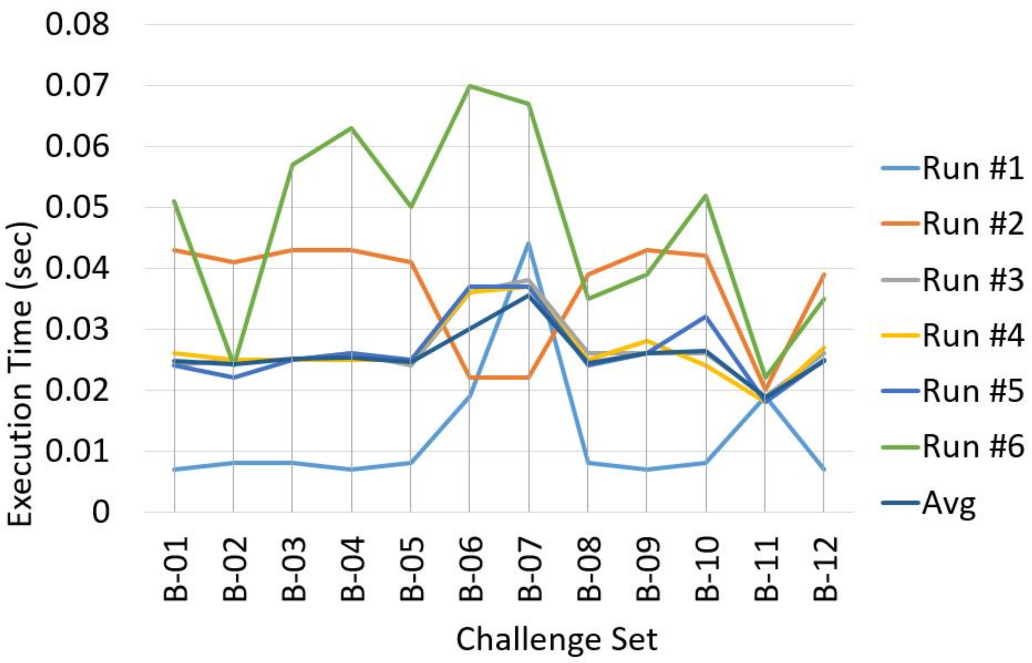


Figure 6—Efficiency of the Production system on the challenge set.

Table 8—Coverage and Generality of each developed rule on CP's

	B-01	**B-02**	**B-03**	**B-04**	**B-05**	**B-06**	**B-07**	**B-08**	**B-09**	**B-10**	**B-11**	**B-12**
Rule 1	✓	✗	✓	✗	✓	✓	✓	✗	✗	✗	✗	✓
Rule 2	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈
Rule 3	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈
Rule 4	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈
Rule 5	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈
Rule 6	✓	✗	✓	✗	✓	✓	✓	✗	✗	✗	✗	✓
Rule 7	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈
Rule 8	✈	✓	✈	✈	✈	✈	✈	✈	✈	✈	✈	✈
Rule 9	✈	✈	✈	✈	✈	✈	✈	✓	✓	✈	✓	✓

6.5 Error Categories

Typical problem solving error's by humans could be categorized into four types namely 1) Repetition, 2) Difference, 3) Wrong Principle, and 4) Incomplete Correlate. In this work, to establish a connection between the humans and the agent with respect to errors, these metrics were analyzed. However all of the error fall under following two categories. Previously in sections 3.1-3.3, the categories of errors were highlighted. In this section, the definition and examples are presented in line with (Kunda et al., 2016).

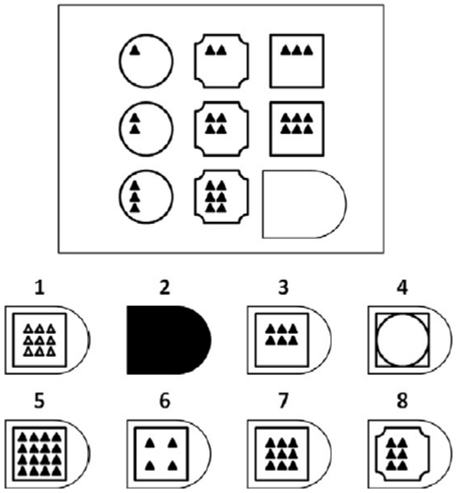


Figure 7—Example of an RPM-like problem. The correct answer is 7. (Kunda et al., 2016)

Repetition: Repetition (R) errors occur when the chosen agent copies a matrix entry adjacent to the blank space. Choosing an R answer choice may represent some degree of perseveration or fixation on the problem matrix, such that an answer is selected using perceptual matching between the matrix entries closest to the blank space and the available answers. Answer choices 3 and 8 in Figure 7 are examples of Repetition errors.

Difference: Difference (D) errors occur when the chosen distracter is qualitatively different in appearance from the other choices. D kind of answer choices include those that are completely blank, as well as those that have extraneous shapes that are not found anywhere else in the problem. Answer choices 2 and 5 in Figure 7 are its examples.

Wrong Principle: Wrong principle (WP) errors occur when the chosen answer choice is a copy or composition of elements from the problem matrix. A WP answer might be chosen if the agent fails to identify the relationship from the matrix and instead combines the entries according to some other rule or relationship. Answer choices 4 and 6 in Figure 7 are examples of wrong principle category of errors.

Incomplete Correlate: Incomplete correlate (IC) errors occur when the chosen answer is almost, but not quite, correct. For example, some IC answer choices represent a rotation or reflection of the correct answer. Answer choice 1 in Figure 7 is an example of an IC. Consolidated error statistics of various errors are as shown in Figure 8.

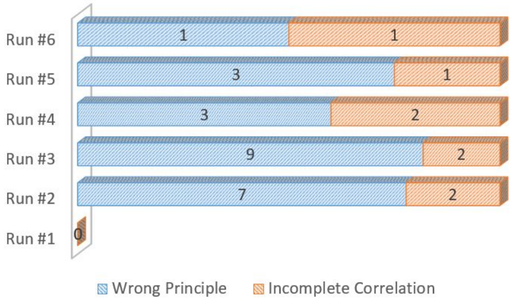


Figure 8—Consolidates error statistics across Basic and Challenge Test sets similar to (Kunda et al., 2016)

6.6 Curious Case of Basic Problem 12

In this section, various rules are derived for solving BP-12, through analysis of image and its relationship with distance metrics. We have four images A,B,C and answer choice D={1,2,3,4,5,6}.

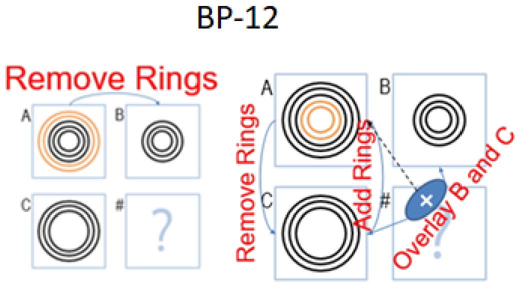


Figure 9—Basic Problem 12, with multiple relationships between the rings.

6.6.1 *Properties of BP-12:*

The basic problem 12 is very unique as it shows multiple possible relationships. As seen in BP-12 (Figure 6.6), affine transformation is of limited use, due to circle's symmetric nature. Four possible relationships exists namely

1. **Removal of two outer rings in A produces B**: From image subtraction opera-

tion, this can be formulated as

$$\mathrm{R1} = \boldsymbol{(A \alpha C) \approx B} \tag{3}$$

2. **Removal of two inner rings produces C**: This can be written as

$$\mathrm{R2} = \boldsymbol{(A \alpha B) \approx C} \tag{4}$$

3. **Diagonally overlaying/adding B and C should lead to A:** Similarly to previous cases

$$\mathrm{R3} = \boldsymbol{(B \beta C) \approx A}$$

, where α is element wise image subtraction and β in image addition, which in turn can be executed using image subtraction.

6.6.2 *Deriving rules 6.2 from BP-12:*

Consider R1, from this rule 6.2 is derived as follows.

$$\mathrm{R1} = \boldsymbol{(A \alpha\ C) \approx\ B}$$

For ease of understanding, α is replaced by "-", without any change in meaning.

$$\mathrm{R1} = \boldsymbol{(A\text{-}C) \approx\ B}$$

Assuming, B>> δ, we have

$$\mathrm{R1} = \boldsymbol{(A\text{-}C) \approx\ (B\text{-}\delta)}$$

By Affine Symbolic Reasoning, we can write

$$\mathrm{R1} = \boldsymbol{(B\text{-}D) \approx (A\text{-}C)}$$

Originally, by definition RMS subtracts images pixels wise, i.e. it implicitly does image difference operation, hence (A-C), can be replaced as *RMS(A,C)* where this returns average of square root of all the non-overlapping pixels from A.

$$\boldsymbol{RMS(A,C) \approx RMS(B,D)}$$

6.6.3 *Deriving rules 6.1 from BP-12:*

Alternatively consider, R3, from previous section.

$$\mathrm{R3} = \boldsymbol{(B \beta C) \approx A} \tag{5}$$

For ease of understanding, β is replaced by "+", without any change in meaning.

$$\mathrm{R3} = \boldsymbol{(B+C) \approx A} \tag{6}$$

Here again, overlay (adding B on top of C) can be obtained by image subtraction. Hence we can have

$$\mathrm{R3} = \boldsymbol{(B-C) \approx A} \tag{7}$$

Assuming, γ is very small i.e. $\gamma <<$A, we can write

$$\mathrm{R3} = \boldsymbol{(B-C) \approx A-\gamma} \tag{8}$$

By affine symbolic reasoning and assuming D$\approx$ γ we can write above equation as

$$\mathrm{R3} = \boldsymbol{(B-C) \approx (A-D)} \tag{9}$$

Again like R1, using RMS in place of subtraction we get rule 6.1

$$\boldsymbol{RMS(B,C) \approx RMS(A,D)}$$