Project 3 : Journal

Swapna Radhakrishnan
sradhakr8@gatech.edu

1 INTRODUCTION

In this journal, the focus is predominantly on the thought process used to build an AI Agent for Project 3, that can solve a set of problems inspired by Raven's progressive matrices (3 x 3 problems).

In Project 1, AI Agent used different types of transformations like rotation, reflection, fill etc. to address 2 x 2 problems using visual approach, and did not use any specific algorithm to solve problems. Using that approach, AI agent was very similar to how human's think and was able to solve 11 Basic and 10 Test problems of type B correctly, but it could not solve any complex problems.

In Project 2, AI agent did not rely on Project 1's code and used an approach of visual heuristics to address 3 x 3 problems. Given a 3x3 problem, AI agent captures relationship measurements like **Dark Pixel Ratio (DPR),** between images horizontally and vertically. It did not generate a potential output image; instead, agent goes through answers provided and finds the answer image that captures this relationship. Joyner et al. (2016) indicates that there are several relationship measurements that can be used, and **DPR** is one of them. Using this approach, AI agent was able to solve 7 Basic and 8 Test problems of type C correctly, but it was not very similar to how humans think.

In Project 3, focus is on 3 x 3 problems of type D and E which are relatively more complex. Since DPR is a generic approach, I would like to reuse Project 2's code for Project 3 and evaluate how the AI agent handles problems D and E, specifically Test sets. This baseline code can then be enhanced further through multiple iterations in case some of the problems needs additional handling.

For the programming part of the problem, Python's *Pillow* library will be used to load images and *Numpy* will be used for DPR and all other calculations. *DprMatcher* will contain the logic for relationship and measurements. *FigureTransformation* has basic transformations like rotation, reflection etc. The method *find_best_match* in *DprMatcher* will be invoked from *Agent.solve()* to get the answer for a problem.

2 SUBMISSION LOG

The log of submissions to auto grader for Project 3 is shown in *Table 1*.

Table 1 — Submission Log

Submission #	Timestamp	Basic D+E Correct	Test D+E Correct	Challenge D+E Correct	Ravens D+E Correct
1	11/24/2020 8:26:00 AM	7	6	4	7
2	11/26/2020 3:28:00 PM	10	3	3	6
3	11/27/2020 11:48:00 AM	13	10	5	9
4	11/27/2020 9:04:00 PM	16	14	8	12
5	11/28/2020 1:19:00 PM	18	14	8	11
6	11/28/2020 7:04:00 PM	20	14	7	10
7	11/28/2020 11:10:00 PM	16	12	3	8
8	11/29/2020 12:29:00 PM	19	9	5	9
9	11/29/2020 2:00:00 PM	16	13	4	12
10	11/29/2020 6:41:00 PM	19	14	4	13
11	11/29/2020 7:55:00 PM	19	13	3	12

3 JOURNAL ENTRIES

3.1 Submission 1

- **Date of Submission:** Submission 1 was done on 11/24/2020 at 8:26 AM.
- **Changes included:** Since this is the first submission, the main goal is to evaluate how project 2 code performs for Set D and E problems, especially Test problems which cannot be tested without a submission. This agent will initially use basic transformations like rotation, reflection etc. to identify relationship between images. If it is unable to identify the transformation, then agent will use **DPR** ratio approach to capture relationship between images. **Dark Pixel Ratio** means the difference in percentage of the number of dark-

colored pixels with respect to the total number of pixels in the contiguous pixel sets of two matrix cells. Images are loaded using black & white mode ('1') to reduce noise. Same logic is used for both Set D and Set E problems. High level logic is given below

Agent.solve() - This is the main method, which takes each problem and invokes *FigureTransformation.find_best_match()* method. If no match is found, then it invokes *DprMatcher.find_best_match()* method, which will return the answer for each problem (number between 1 and 8)

FigureTransformation.find_best_match() - This method takes each 3 x 3 problem and runs transformations like identity, rotation, reflection, split and reflection and tries to find a matching answer using transformations.

DprMatcher.find_best_match() - This method uses same approach as project 2, where in each 3 x 3 problem is divided into rows and columns of images. Each image is further divided to 16 blocks, and agent calculates ratio by ratio of DPRs for each block. The agent will then further calculate ratio by ratio of a block in one image to another image, for example, from image A to D. Once this is done for all blocks of all images, in a particular column, agent will use sum to get total deviation for a column. Same approach is repeated for other columns. For last column in image I position, agent will loop through answer images 1 to 8 to get the total deviation. Using these deviations, agent will then calculate mean deviation of columns by calculating difference between columns 3 and 2, and columns 3 and 1. This is the mean column deviation. The same approach is then repeated for rows as well to obtain mean row deviation. The answer option for a given problem will be the one that has lowest total deviation which is the sum of mean column deviation and mean row deviation.

- **Comparison with humans:** As this is a reference from project 2, this version of agent is not similar to how humans think. Humans would normally try to find patterns and then approach problem solving, whereas this version of AI agent predominantly relies on DPR approach to solve problems.
- **Agent's Performance:** Agent solved 7 Basic (3 Basic D, 4 Basic E), 6 Test (3 Test D, 2 Test E), 4 Challenge D+E and 7 Ravens D+E problems successfully. It was able to identify basic transformations like identity, and some problems where shape of the object in the center of image is same across rows/columns. Agent performed poorly for problems in Set E where there were image additions/subtractions etc. On reflection, I think DPR should be used as a fallback

option only and agent should try to identify relationship using image additions/ subtractions etc. Since problem set E and problem set D looks very different, I will focus on getting problem set E correct first as a next step. Since *numpy* is used for all logic, Agent's execution time was 2.54 seconds, which is very efficient.

3.2 Submission 2

- **Date of Submission:** Submission 2 was done on 11/26/2020 at 3:28 PM.
- **Changes included:** In this version, focus is on addressing specific pattern of problems in Set E, where in one image in a row is addition of other two images. *FigureTransformation* class is updated to include add transform using *numpy*, which will be evaluated for every problem and DPR is used as a fallback.
- **Comparison with humans:** This version of agent is similar to humans because human beings try to find patterns in a problem, and then find a solution using it. Agent also does the same, when it tries to address specific category of transformation related to image addition.
- **Agent's Performance:** Agent solved 10 Basic (3 Basic D, 7 Basic E), 6 Test (3 Test D, 2 Test E), 3 Challenge D+E and 6 Ravens D+E problems successfully. Agent successfully handled image addition problems but could not handle other transformations like image subtraction or XOR in set E. Also, it currently does not find any patterns in set D. On reflection, the next step is to improve the agent so that agent can handle other transformations in set E like subtraction and XOR. Also, the Agent's overall execution time increased to 3.73 seconds.

3.3 Submission 3

- **Date of Submission:** Submission 3 was done on 11/27/2020 at 11:47 AM
- **Changes included:** In this version, agent is updated to identify transformation for image subtractions and XOR. Image subtraction implies that one image in a row is the output of subtracting other two images in same row. Similarly, for XOR, transformation will find XOR of three images in a row. These transformations are added to specifically handle some patterns found in set E problems.
- **Comparison with Humans:** In my opinion, this version of the agent is quite similar to how humans think. Since pattern matching worked in the past,

agent tries to look for new patterns in set E and addresses them. This is also how humans approach when there are several problems in a problem set. They categorize problems into patterns and address them.

- **Agent's Performance**: Agent was able to reach good performance for set E problems, in this submission. Agent solved 13 Basic (2 Basic D, 11 Basic E), 10 Test (2 Test D, 8 Test E), 5 Challenge D+E and 9 Ravens D+E problems successfully. This indicates that agent is in the right direction of solving problems as it reached good performance for Set E problems. Agent still struggles to solve all set D problems except identity. As a next step, agent will try to identify patterns for Set D and address them specifically. Also, Overall execution speed increased to 6.04 seconds, which is still good.

3.4 Submission 4

- **Date of Submission:** Submission 4 was done on 11/27/2020 at 9:04 PM
- **Changes included:** In previous submission, agent predominantly checks for transformations horizontally. In this submission, agent addresses triangle symmetry. This pattern was noticed in Set D problems. Agent will check if transformations like image addition, subtraction, XOR etc. exist between images B,F,G as well as C, D and H. If it exists, agent will use that transformation on images A,E and identify correct answer from answer options 1 to 8.
- **Comparison with Humans:** Just like earlier, this version of agent is also similar to how humans think. As humans, we try to find more and more patterns to solve problems. Agent also does the same and tries to generalize it so that it can address both set D and E problems.
- **Agent's Performance**: Agent was able to improve performance on set D problems, in this submission. Agent solved 16 Basic (6 Basic D, 10 Basic E), 14 Test (6 Test D, 8 Test E), 8 Challenge D+E and 12 Ravens D+E problems successfully. Agent was able to solve triangle symmetry problems correctly in Set D but still struggles where there is a complex transformation between images, like problem D-05, D-07, D-09, D-08, D-12 etc. On reflection, this would need some specific handling to address specific type of transformations. Also, Overall execution speed reduced to 2.84 seconds.

3.5 Submission 5

- **Date of Submission:** Submission 5 was done on 11/28/2020 at 1:19 PM

- **Changes included:** The focus for this version is to address few more problems in Set D correctly, to achieve good performance for Set D problems. By analyzing Basic D problems, noticed a pattern of matching answers by elimination where you can eliminate images from answer options. If the transformation across rows/columns/diagonally is not *identity*, then assumption is answer image should not be one of the images in A to H. Hence, by elimination you can reduce the potential answer options, and if only one answer option remains, then that is the correct answer for the problem. Agent's logic is updated to check for match by elimination if agent is unable to find transformations prior to DPR Matching.
- **Comparison with Humans:** In my opinion, this version of the agent is exactly how humans/I think. Human beings quite often using these approaches, even in day to day lives. Given a complex problem and we remove answer choices which are not correct and whatever remains becomes the correct answer. This agent behaves in the same manner.
- **Agent's Performance**: Agent was able to improve performance on set D problems, in this submission. Agent solved 18 Basic (8 Basic D, 10 Basic E), 14 Test (6 Test D, 8 Test E), 8 Challenge D+E and 11 Ravens D+E problems successfully. Even though it did not use pattern matching, agent was able to solve some of the complex problems like D-05, D-07, D-09 using this approach. Even though this approach worked smoothly for Basic D, it struggled for same set of problems in Test D. On reflection, elimination is a nice to have approach but cannot be relied on. Agent needs to focus on finding some patterns to solve complex D problems. The overall execution speed is about 4.55 seconds.

3.6 Submission 6

- **Date of Submission:** Submission 6 was done on 11/28/2020 at 7:04 PM
- **Changes included:** At this stage, I was struggling to find patterns which were generic and not too specific for problems in D like D-05,07,08,09 etc. So, worked on adjusting threshold values for similarity scores used while comparing images. This was also a trial and error approach to check if existing logic was not solving test D only due to lower threshold values.

- **Comparison with Humans:** This version of AI agent is not similar to how humans think. This is more of a computational approach to address problems.
- **Agent's Performance**: Agent was able to improve performance on set D problems, in this submission. Agent solved 20 Basic (10 Basic D, 10 Basic E), 14 Test (6 Test D, 8 Test E), 7 Challenge D+E and 10 Ravens D+E problems successfully. With this approach, agent was able to address complex problems in Basic D correctly like D-05, D-07, D-09 but it still could not solve corresponding Test D problems. Moreover, with these incremental changes, agent's performance in Challenge and Ravens problems have started to decrease. On reflection, this is a nice to have approach but still does not address the root cause. As a next step, agent needs another approach to achieve good performance on Test D. The overall execution speed reduced to 4.10 seconds.

3.7 Submission 7

- **Date of Submission:** Submission 7 was done on 11/28/2020 at 11:10 PM
- **Changes included:** As an alternate option to address problems in D like D-05,07,08,09 etc., worked on improving fallback option of DPR to improve performance of Test D. Adjusted the weights used during mean deviation calculation to give more accurate relationship between images. This was also a trial and error approach to check if fallback option of DPR could help solve test D problems
- **Comparison with Humans:** This version of AI agent is not similar to how humans think. Like earlier, this is more of computational approach to address problems.
- **Agent's Performance**: Agent's performance reduced in this submission. Agent solved 16 Basic (8 Basic D, 8 Basic E), 12 Test (5 Test D, 7 Test E), 3 Challenge D+E and 8 Ravens D+E problems successfully. With this approach, agent started failing for both Test D and Test E problems, which indicates this was not the right approach to proceed. On reflection, agent needs to find specific patterns to address Basic D problems D-05,D-07,D-08,D-09. The hope is by targeting specific basic D problems, corresponding Test D problems will get addressed as well. The overall execution speed reduced to 3.83 seconds.

3.8 Submission 8

- **Date of Submission:** Submission 8 was done on 11/29/2020 at 12:29 PM
- **Changes included:** In this submission, added a new logic for addressing patterns for Set D problems. For problems like D-05,D-07,D-08,D-09 etc., noticed a pattern of intersection that can be similar between pairs in a triplet. For example, if the intersection of images B:F and F:G are same and C:D and D:H are also same, then that is a pattern. Agent uses this to find an answer image which matches the intersection of A:E, E: answer option and A:answer option.
- **Comparison with Humans:** This version of AI agent is very similar to how humans think. In this version, AI agent found a pattern which is generic and can handle multiple problems and uses it. Humans also behave in the same way, where in we find common patterns that can be used across problems to get solutions.
- **Agent's Performance**: Agent's performance on Test D reduced drastically in this submission. Agent solved 19 Basic (9 Basic D, 10 Basic E), 9 Test (1 Test D, 8 Test E), 5 Challenge D+E and 9 Ravens D+E problems successfully. Agent struggled to address all Test D problems except identity, which clearly indicates this may not be the right approach for agent. On reflection, agent needs to find alternate patterns to address Basic D problems D-05,D-07,D-08,D-09. The overall execution speed reduced to 3.29 seconds.

3.9 Submission 9

- **Date of Submission:** Submission 9 was done on 11/29/2020 at 2:00 PM
- **Changes included:** In the previous submission, pattern matching of image intersection resulted in failure for all Test D problems. In this submission, updated the logic of fallback option of DPR on submission 7's code, to also perform calculations across diagonal / triangle symmetry. By getting deviation of images across triangle symmetry, the hope is to address D problems which are complex like D-05,07,08,09 etc., which are not currently handled by DPR. Also, in this version, matching by elimination logic is removed to get more accurate results using DPR.
- **Comparison with Humans:** This version of AI agent is not very similar to how humans think. In this version, AI agent relies on improving

computational logic to address complex problems, since pattern matching did not work earlier. Human beings would not behave in this manner.

- **Agent's Performance**: Agent's performance improved in this submission. Agent solved 16 Basic (7 Basic D, 9 Basic E), 13 Test (5 Test D, 8 Test E), 4 Challenge D+E and 12 Ravens D+E problems successfully. With this approach, agent seems to be in the right path of problem solving, and it has recovered from failures in prior submission. Also, without elimination approach, this submission gives more accurate results. Agent still struggles to address Basic and Test D problems related to D-05,D-07,D-08,D-09,D-12 etc., which are more complex. On reflection, agent needs to re-visit option of finding specific patterns to address Basic D problems D-05,D-07,D-08,D-09, so that Test set can be addressed as well. The overall execution speed increased to 11.78 seconds.

3.10 Submission 10

- **Date of Submission:** Submission 10 was done on 11/29/2020 at 6:41 PM
- **Changes included:** In this version, agent includes a new pattern to address D-05, D-07, D-09 problems. One of the patterns found is adding images across a row, will result in same image for each row. This pattern can then be used to determine the answer image, by comparing if the resulting image with answer option is similar to images created by adding rows A:C or D:F. This image addition approach is further extended to add images column wise, diagonally, reverse diagonally as well as triangle symmetry cases. Even though this is a specific pattern for D-05,D-07, D-09 problem, it is generalized sufficiently so that it can be applied to all problems as well. Also, in this version, agent has separate solve method/logic to handle Set D and Set E problems. This is to avoid any implications of newly introduced pattern check on Set E problems.
- **Comparison with Humans:** This version of AI agent is very similar to how humans think. In this version, AI agent relies on pattern matching to solve complex problems. Moreover, having separate solutions for Set D and Set E is also very human like.
- **Agent's Performance**: Agent's performance improved in this submission. Agent solved 19 Basic (10 Basic D, 9 Basic E), 14 Test (6 Test D, 8 Test E), 4 Challenge D+E and 13 Ravens D+E problems successfully. With this

approach, agent can handle Basic D problems correctly using patterns, even though it still could not address corresponding Test D problems like D-05,06,07,08 etc. On reflection, agent may need some adjustments to threshold values to handle Test D problems. The overall execution speed reduced to 9.38 seconds

3.11 Submission 11

- **Date of Submission:** Submission 11 was done on 11/29/2020 at 7:55 PM
- **Changes included:** This was a trial and error approach to get just one more answer correct in Test D problems, to reach good performance in Test D. The threshold values while comparing images are more relaxed now, to check if that can address the Test D problem as Basic D works as expected.
- **Comparison with Humans:** This version of AI agent is exactly how humans think. As humans, we are always optimizing to find the ideal threshold or ratio to get better results. AI agent also does the same.
- **Agent's Performance**: Agent's performance reduced in this submission. Agent solved 19 Basic (10 Basic D, 9 Basic E), 13 Test (6 Test D, 7 Test E), 3 Challenge D+E and 12 Ravens D+E problems successfully. Adjusting the threshold values results in failures for earlier working problems, which is not a good approach to use. Agent continues to struggle in handling Test D problems specifically D-05, D-06, D07, D-08, D-11, D-12. The overall execution speed is 10.39 seconds

4 CONCLUSION

To summarize, the overall process of designing AI agent for project 3 was predominantly based on finding patterns to solve problems, thereby targeting one type of problem at a time. When starting with project 3, the initial approach was to rely on Project 2 and use **DPR** to solve problems in Set D and E. Since agent did not give expected results in first submission, starting submission 2 the focus is to find patterns to address specific categories of problems. Patterns like addition, subtraction and XOR worked for Set E problems, while Set D problems needed more specialized handling. DPR is still used as a fallback option in case transformations cannot identify the pattern. With this approach, the final version of agent became a hybrid form of transformations and DPR approach.

The final version of the agent is *gradescope* is Submission 10. This agent's behavior is quite similar to the way I would approach the test. When I take a test, I will start with a list of generic transformations that will help solve the problem. If generic transformations are not sufficient, then I will check for transformations that can handle specific problems. As a backout option, I would then rely on established methods to address the problem. This AI agent also behaves the same way. It uses generic transformations initially, then specific transformations and in the absence of a solution, relies on established approaches like DPR to find the most possible answer.

The final version of AI agent can solve 19 Basic, 14 Test, 4 Challenge and 13 Raven's tests. It could not reach expected performance for Test D problems and was able to handle 6/12 of Test D problems. If I had more time, I would make multiple improvements to the agent. I would add more transformations and pattern matching to address Test D problems. I would also check the sequence of transformations or add more weightage to some transformations, so that most optimal transformation can be chosen. Moreover, I would add features where the agent is able to detect shapes using connected component labelling approach. Finally, I would like to add some amount of learning to the agent using neural networks, where the agent can learn from its own mistakes and improve, to generate more accurate results.

5 REFERENCES

1. Joyner, D. A., Bedwell, D., Graham, C., Lemmon, W., Martinez, O., & Goel, A. K. (2015, June). Using Human Computation to Acquire Novel Methods for Addressing Visual Analogy Problems on Intelligence Tests. In ICCC (pp. 23-30).