RPC Milestone C Journal

Author Name
username@gatech.edu

Abstract—Welcome to Joyner

1 CURRENT FUNCTIONALITY

The agent's evolution from solving 2x2 problems to handling 3x3 problems has involved several key developments in its approach. Initially, when dealing with 2x2 problems, the agent focused on straightforward tasks like internal share detection and direct pixel comparison using Difference Pixel Ratio (DPR) and Intersection Pixel Ratio (IPR). These metrics allowed the agent to compare images directly and find the best matching candidate from a set of options. For 2x2 problems, the agent also applied basic transformations like flipping, rotating, and mirroring to detect similarities between figures.

However, as the agent transitioned to solving 3x3 problems, it required a more complex strategy. In these problems, the agent expanded its operations to consider relationships between multiple rows and columns simultaneously. For each problem, the agent now compares images across entire rows (A to B to C) and columns (A to D to G), calculating DPR and IPR values for these relationships. Doing this enables the agent to identify the overall pattern or transformation applied across both the rows and columns. For example, it compares pairs like A and B, D and E, and G and H to identify transformations such as scaling, shape changes, or rotational symmetry.

Moreover, the agent now generates candidate solutions for each option, assessing their likelihood based on adaptive DPR/IPR thresholds that are dynamically calculated to fit the patterns observed in the problem. This adaptive thresholding allows the agent to account for variations in the complexity of shapes or transformations, refining its approach to find the closest match.

2 AGENT PERFORMANCE

2.1 How well does your agent currently perform? How many problems does it get right on the Set C problems?

In local testing, the agent correctly answers 10 out of 12 Basic C problems and 5 out of 12 Challenge C problems. The agent shows consistent performance across simpler Basic C problems but struggles more with the Challenge C set. The Gradescope testing reflects a similar pattern, with the agent passing 9 out of 12 Basic C problems and 8 out of 12 Test C problems. However, like the local testing, the agent encounters more difficulty with the Challenge C problems, passing 5 out of 12.

2.2 What problems does your agent perform well on? What problems (if any) does it struggle on? Why do you think it performs well on some but struggles on others (if any)?

The agent performs well on problems that involve direct image comparison, pattern recognition, and simpler transformations such as flipping, rotating, or mirroring. For instance, Basic C problems like C-02, C-03, and C-04, which have clear visual patterns, are handled accurately by the agent. The pixel comparison techniques it uses for similarity detection, combined with transformations like image flipping, help it solve problems where the relationships between the figures are visually straightforward.

However, the agent struggles with more complex problems, particularly in the Challenge C set. These problems clearly require more abstract reasoning, such as identifying shape addition or removal, counting the number of internal shapes, or detecting subtle image transformations that are not purely pixel based. As an example, Challenge C-09, and C-10 include relationships between the figures, which the agent's current pixel-based approach fails to capture. To solve these types of problem, the agent would need to account for the geometric relationship between the internal shapes rather than relying solely on pixel-level comparisons. Implementing methods like contour detection, vertex counting, or geometric transformations might yield better results for this type of problem.

2.3 How efficient is your agent? Does it take a long time to run? (Give specific metrics) Does it slow down significantly on certain kinds of problems? If so, by how much? Why do you think your agent runs slower on certain problems?

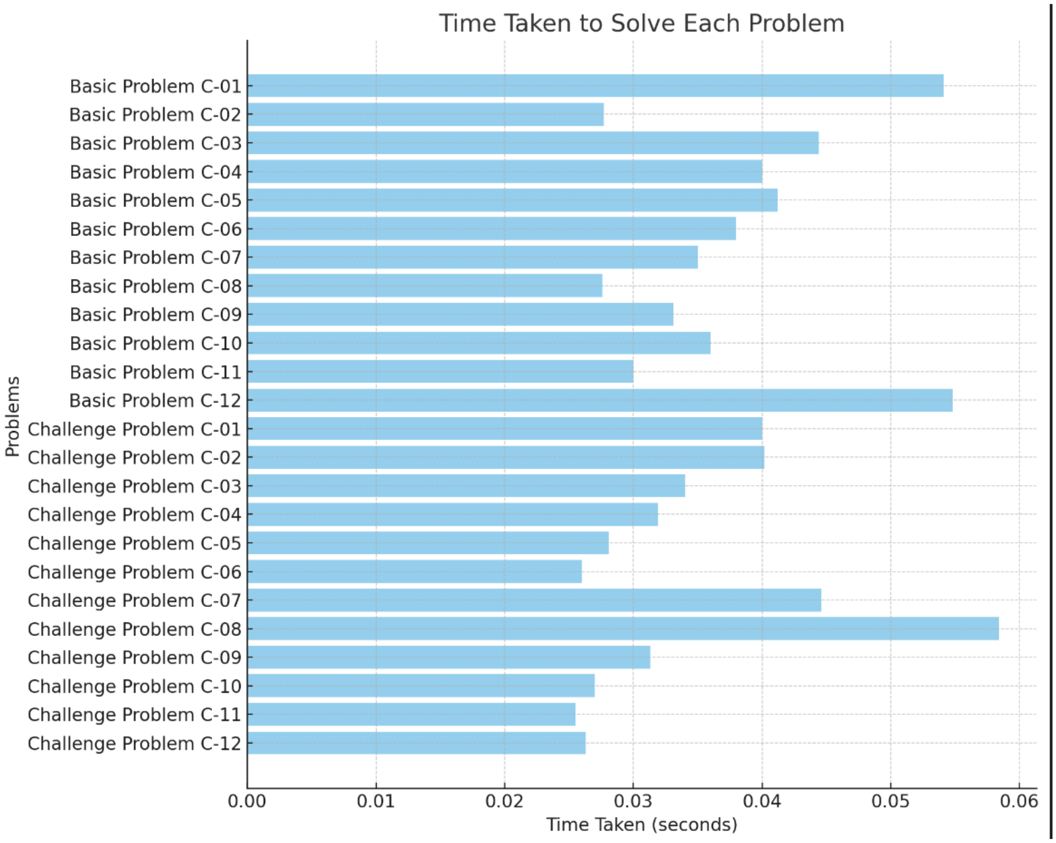

Figure 1—Problem Set C Times to Solve Problems.

The agent demonstrates good efficiency overall, consistently solving both Basic and Challenge problems in under 0.06 seconds during local testing. On average, it takes 0.0384 seconds to solve Basic problems and 0.0365 seconds for Challenge problems, showing similar performance across both categories. While there are no major slowdowns, some problems, like Challenge Problem C-08, took slightly longer at 0.0584 seconds. This is most likely due to the increased complexity of this problem and the fact that there are visual patterns present that require more pixel comparisons and image processing, thus more time to solve.

3 AGENT IMPROVEMENTS

3.1 How do you plan to improve your agent's performance on these problems before the final project submission

To improve the agent's performance before the final project submission, the focus will be on refining the image comparison methods, particularly the DPR and IPR calculations. This is primarily because the agent, in its current state, strictly relies on DPR and IPR for solving 3x3 problems. While these methods work effectively in many cases, they struggle on more complex problems where pixel-level analysis alone may not be sufficient. One improvement to address this limitation is to introduce adaptive thresholds for pixel similarity, which could help better handle intricate patterns or shape transformations that DPR and IPR aren't capturing effectively. Additionally, integrating a more advanced shape recognition or pattern-matching algorithm would be beneficial, especially for problems where visual complexity goes beyond simple pixel comparison. This enhancement could allow the agent to better understand the spatial relationships between objects and perform more accurate comparisons. To do this, the agent could use a transformation-based analysis, such as flips, rotations, and mirrors, to be more flexible, allowing the agent to detect subtle or unconventional transformations in 3x3 problems. This will be particularly useful for problems where geometric transformations are not the primary visual relationship, but instead involve composite patterns or more abstract arrangements like Challenge Problem C-08.

3.2 Looking ahead to Sets D and E, which problems do you think your agent will be able to solve at its present stage? Which problems will it struggle on?

Looking ahead to Sets D and E, the agent, in its current state, is likely to perform well on problems where direct pixel comparisons are effective, especially those with simple visual transformations or uniform geometric patterns. This would include all the problems in Basic Problems D and E.

However, the agent will likely struggle with more abstract or complex problems in Sets D and E that involve intricate relationships or transformations not easily captured through pixel-level analysis. For instance, problems involving changes in object count, size, or position, as well as problems requiring understanding of more sophisticated spatial arrangements, may pose a significant challenge. An example of this can be seen in Challenge Problem D-11.

4 PEER FEEDBACK

In terms of feedback from classmates, I would be particularly interested in learning from those who, like me, are using DPR and IPR as the primary checks in either 2x2 problems, 3x3 problems, or both. Specifically, I want to know if they are incorporating other checks when DPR and IPR are not sufficient, and if so, how they are doing so. While DPR and IPR work well for problems involving direct pixel comparison, it is not capturing more abstract relationships or differentiating between patterns with similar pixel distributions.

As such, I'm curious to learn if other students are layering additional techniques, such as shape recognition, edge detection, or symmetry checks, after DPR and IPR fail to provide a clear answer. It would also be helpful to know how they are structuring their agents to decide when to apply these additional checks, whether they’re based on a threshold, pattern recognition, or some other heuristic. Ultimately, this type of feedback would offer valuable insights into enhancing my own agent's problem-solving capabilities in more complex scenarios, which will be necessary for the upcoming assignments.