The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Thinking in PolAR Pictures: Using Rotation-Friendly Mental Images to Solve Leiter-R Form Completion

Joshua H. Palmer, Maithilee Kunda
Department of Electrical Engineering and Computer Science, Vanderbilt University
PMB 351679, 2301 Vanderbilt Place, Nashville, TN 37235-1679, USA

Abstract

The Leiter International Performance Scale-Revised (Leiter-R) is a standardized cognitive test that seeks to “provide a nonverbal measure of general intelligence by sampling a wide variety of functions from memory to nonverbal reasoning.” Understanding the computational building blocks of nonverbal cognition, as measured by the Leiter-R, is an important step towards understanding human nonverbal cognition, especially with respect to typical and atypical trajectories of child development. One subtest of the Leiter-R, Form Completion, involves synthesizing and localizing a visual figure from its constituent slices. Form Completion poses an interesting nonverbal problem that seems to combine several aspects of visual memory, mental rotation, and visual search. We describe a new computational cognitive model that addresses Form Completion using a novel, mental-rotation-friendly image representation that we call the Polar Augmented Resolution (PolAR) Picture, which enables high-fidelity mental rotation operations. We present preliminary results using actual Leiter-R test items and discuss directions for future work.

Introduction

The Leiter International Performance Scale-Revised (Leiter-R) is a standardized test of human cognition that seeks to “provide a nonverbal measure of general intelligence by sampling a wide variety of functions from memory to nonverbal reasoning.” (Kaplan and Roid 2010) It consists of twenty subtests, all administered non-verbally, with subtest variants intended for individuals ranging from children as young as 2 years of age to adults. Examples of specific subtests include (Roid and Miller 1997): Figure Ground (FG), finding a figure/object embedded in an arbitrarily complex background; Form Completion (FC), synthesizing and localizing an object from its constituent slices; and Repeated Patterns (RP), filling in blanks with previously-seen elements.

Nonverbal, and especially visuospatial, cognition, is a critical factor in child development. Atypical patterns of nonverbal cognitive development characterize many conditions such as autism (Dawson et al. 2007), dyslexia (Smith-Spark et al. 2003), and others, and furthermore, visuospatial ability is increasingly recognized as a key contributor to STEM education and career success (Wai, Lubinski, and Benbow 2009). Thus, improving methods for measuring and understanding nonverbal human cognition is an important area of need for current research.

Neuropsychological tests like the Leiter-R, where a person’s performance is measured in behavioral terms, still represent the state-of-the-art in measuring human cognition. While advances in neuroimaging have certainly yielded many important insights into brain functioning, neuroimaging by itself does not directly tell us about a person’s *cognitive* functions. On tests like the Leiter-R, then, it is important to understand exactly what cognitive strategies are indicated by particular behavioral patterns of performance. AI systems that simulate problem solving on such tests can be used as a research tool to investigate the extent to which these cognitive tests are, in fact, testing the constructs they claim to be testing. While computational experiments with an AI system cannot directly tell us what strategies a person is using, they can be very informative about the space of cognitive strategies that are possible on a given test.

For example, the Leiter-R developers argue that the Figure-Ground (FG) subtest relies on a discriminative approach, whereas Form Completion (FC) is more generative/synthesizing. The FC literature indicates that examinees create a mental image of the fragments and rotate them to complete the full object. However, we hypothesize that FC can also be solved with a discriminative matching procedure a la FG, modulo rotation. If different individuals solve FC in different ways—some using a discriminative approach and some using a generative approach—then similar FC scores in two different individuals could in fact have very different cognitive interpretations.

In this paper, we present a new computational cognitive model that solves problems from the Leiter-R FC subtest. Our central claim is that the set of representations and operations proposed here are *sufficient* for solving at least some FC problems. Given that (to our knowledge) no previous computational models of the Leiter-R test exist, this computational account by itself provides some scientific value by helping us understand the information processing requirements of the FC subtest. Future work will include making comparisons to human performance data and also expanding the range of operations and strategies used by the model to try to capture additional dimensions of cognitive strategy variation that may be measurable by the FC test.

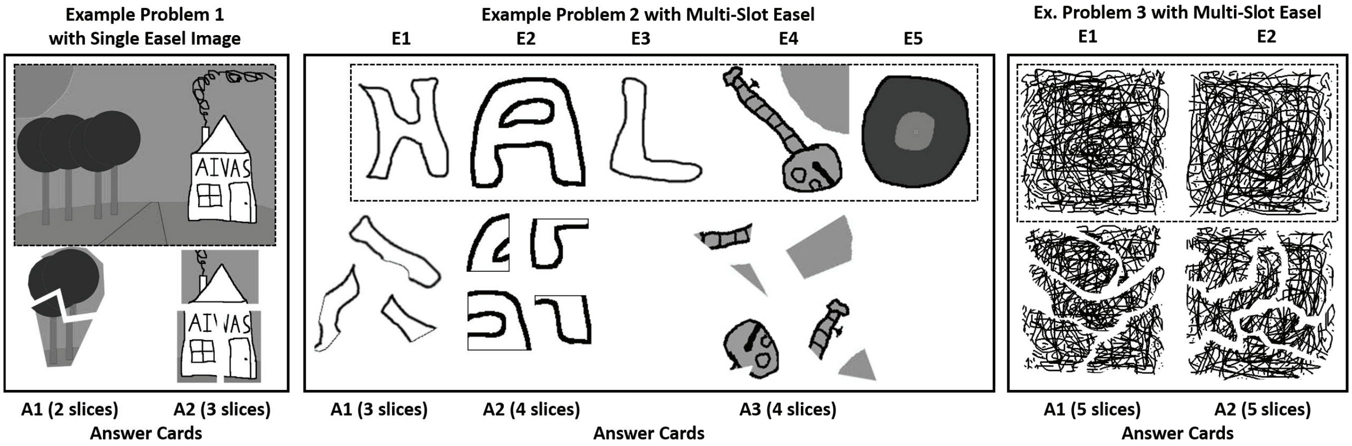

Figure 1: Examples of problems representative of those on the Leiter-R Form Completion (FC) subtest. To protect test security, actual problems are not shown. Terminology of easels, slots, and slices is our own. Left: Single-easel problem, in which each of two cards containing multiple image "slices" must be matched to their corresponding locations on the original (top) easel image. Center: Easy multi-slot easel problem, in which each of three cards containing multiple image "slices" must be matched to their corresponding "slots" on the original (top) divided easel image. Right: Difficult multi-slot problem.

Leiter-R Form Completion (FC)

In the Leiter-R Form Completion (FC) subtest, there are two main types of problems, labeled here as *single-easel* and *multi-slot*. In the former, examinees are asked to match the visual content of each one of a set of small *answer cards* to where it is found on a larger picture displayed on a tabletop easel. In the latter, the easel is split into multiple *slots* requiring examinees to pair each card choice with its corresponding slot. In either type of problem, on each card, the object of interest has been broken up into *slices* of arbitrary shape which have been arbitrarily rotated, scaled, and/or flipped.

See Figure 1 for illustrations of single-easel (left) and multi-slot (center and right) problems, with corresponding answer cards and image slices. The actual problems from the Leiter-R FC subtest range in difficulty levels similar to these example problems.

The Leiter-R FC subtest is a difficult task that seems to require a combination of visual memory, mental rotation, visual search, and more. While the task by itself would not be difficult for a specialized AI system to solve, we are interested in using a cognitive modeling approach to better understand task demands, and to identify combinations of representations and reasoning approaches that can yield successful performance on the test, or that lead to patterns of unsuccessful performance.

Related Work in AI

While there is much previous work in AI that models visual cognitive capabilities as using propositional (i.e., abstract) representations and operations (see (Glasgow, Narayanan, and Chandrasekaran 1995) for a review), we follow the alternate approach of modeling visual cognitive capabilities as being imagistic in nature. Representations are fundamentally image-like, and operations like "mental rotation" are continuous in time.

This theoretical stance is one espoused by Kosslyn in seminal work on human visual mental imagery (Kosslyn and Shwartz 1977) and continued in further AI work in domains that include reasoning about object stability (Funt 1980), molecular shape (Glasgow and Papadias 1992), and line graphs (Tabachneck-Schijf, Leonardo, and Simon 1997). Our model also shares similarities to template-based accounts of visual search, especially multi-stage search models (Rao et al. 2002; Kunda and Ting 2016). The question of learning operations like mental rotation is an important one for which the choice of representation is a critical factor; while we do not address the learning problem in this paper, we ultimately expect our work to continue in that direction, expanding on a few approaches that exist in the AI literature (Mel 1986; Memisevic and Hinton 2007; Seepanomwan et al. 2013).

With respect to studies of cognitive tests in particular, previous work in AI has taken both approaches. For example, our own work has explored imagery-based strategies for solving problems from tests such as Raven's Progressive Matrices (RPM) (Kunda, McGreggor, and Goel 2013), Block Design (Kunda, El Banani, and Rehg 2016), Embedded Figures (Kunda and Ting 2016), and Paper Folding (Ainooson and Kunda 2017). Other work in AI focusing on visuospatial cognitive tests, especially on the RPM test, has studied the use of propositional, non-imagery-based representations and strategies (Carpenter, Just, and Shell 1990; Lovett, Forbus, and Usher 2010; Rasmussen and Eliasmith 2011; Strannegård, Cirillo, and Ström 2013).

The Approach

In this paper, we will present three different computational models to solve FC. Before we get there we need to introduce a key representational concept: the PolAR Picture.

The PolAR Picture

We describe here a new image representation that we call the PolAR (**Pol**ar **A**ugmented **R**esolution) Picture. The inspiration for the PolAR Picture is the "WHISPER retina", which was a rotation-friendly image representation used to analyze the stability of "blocks world" structures in early AI research (Funt 1980).

The main idea behind the PolAR Picture representation is that, unlike rectangular pixel arrangements, circular pixel arrangements support high-fidelity circular rotation operations. While a PolAR Picture is stored using a rectangular matrix data structure, as standard images are, rows represent wedges and columns represent concentric rings (Funt 1980). This not only allows for efficient (albeit discrete) rotation, but also creates a situation in which inner rings have a higher resolution over that portion of the visual field than do outer rings, in loose similarity to the resolution differences in human vision. Hence, the term **A**ugmented **R**esolution refers to this fact that resolution increases (often into the subpixel regime) towards the center.

As a result of this resolution change, PolAR Pictures have a central blindspot in the middle, because the resolution increase would otherwise result in an explosive increase in the number of pixels needed to represent centrally located information. (This blindsplot in PolAR Pictures is not so closely related to the blindsplot in human vision, but rather is a constraint of this representation that is a feature of interest, but not one that is intended to be a major theoretical commitment of our models.)

PolAR Pictures are computed from regular, rectangular-pixel images as follows. Let h_j be the height (in pixels) of any PolAR pixel in ring j. Let n be the number of rings and ring n be the outermost ring (one-based indexing). The most important PolAR parameter is the size of the square input (called "size", in units of pixels). Other parameters include h_n and "br", the ratio of the blindspot diameter to the PolAR Picture diameter. The former fixes the resolution of ring n and determines the number of wedges (i.e. rows), while the latter establishes the number of rings (i.e. columns). In all of our experiments, br = 0.2.

Figure 2 shows an example of Einstein as a PolAR Picture, varying these two auxiliary parameters. Notice that increasing h_n by a factor of two DECREASES both the number of wedges and rings also by a factor of two.

In actuality, each PolAR pixel is a "pseudo-square"; if h_n = 2, for example, then a PolAR pixel in the outermost ring will have a width of two and a center length of two pixels as well. In other words, the length of its inner edge is slightly less than two and the length of its outer edge is slightly more than two. Figure 3 provides a closeup of a PolAR picture, illustrating how the PolAR pixels decrease in size towards the center.

Equations Determining PoLAR Attributes: Let R_j be the distance (in pixels) from the PolAR center to the center of any PolAR pixel in ring j. Let R_{bs} be radius of the central blindspot. Let $ratio := h_{j-1}/h_j = R_{j-1}/R_j \; \forall j \in [2, n]$. Let m be the number of wedges. As before, n is the number of rings. Then, Algorithm 1 details how the inputs (size, h_n, br) establish the PoLAR Picture's attributes.

Algorithm 1 Establishing the PolAR Picture's attributes

```
procedure CALCPOLARATTRS
    $midpoint \leftarrow 0.5 * (size - 1)$
    $center \leftarrow [midpoint, midpoint]$
    $R_n^{ideal} \leftarrow 0.5 * (size - h_n)$
    $m \leftarrow \lfloor 2\pi R_n^{ideal} / h_n \rfloor$
    $R_n \leftarrow m h_n / (2\pi)$
    $ratio \leftarrow \frac{2R_n - h_n}{2R_n + h_n}$
    $ring\_split \leftarrow \log_{ratio}(1 - h_n/(2R_n))$
    $R_{bs}^{ideal} \leftarrow 0.5 * (br * size)$
    $log \leftarrow \log_{ratio}(R_{bs}^{ideal}/R_n)$
    $delta \leftarrow log - \lfloor log \rfloor$
    if $delta < ring\_split$ then
        $n \leftarrow \lfloor log \rfloor + 1$
    else
        $n \leftarrow \lfloor log \rfloor + 2$
    $R_{bs} \leftarrow ratio^{n + ring\_split - 1}$
```

PolAR Mappings: With the overall attributes established, it remains to calculate, for every PolAR pixel, which input Cartesian pixels map to it. Algorithm 2 details how to determine the relevant Cartesian "cPix" given a PolAR pixel with wedge and ring indices (i, j).

Algorithm 2 Mapping Cartesian pixels to PolAR pixels

```
# $(i, j) = (wedge\_index, ring\_index)$
procedure FINDCARTPIX(i,j)
    $\theta_i \leftarrow (2\pi/nWedges) * (i + 0.5)$
    $\hat{v} \leftarrow [\cos\theta_i, \sin\theta_i]$
    $R_j \leftarrow R_n * ratio^{n-j}$
    $closest\_cartPixel \leftarrow round(center + R_j \hat{v})$

    # $FindNeighbors(i, j, closest\_cartPixel)$
    # performs BFS that begins at $closest\_cartPixel$
    # and recursively explores neighbors, adding any
    # Cartesian pixels whose closest PolAR pixel is $(i, j)$
    $cPix \leftarrow FindNeighbors(i, j, closest\_cartPixel)$
    return $cPix$
```

The PolAR Picture can now be created by averaging the values of all Cartesian pixels mapping to a particular PolAR pixel (i, j) $\forall i \in [1, m], j \in [1, n]$. Ringwise "densities" (essentially averages which play an integral role in each of the models) can then be calculated by unioning the mappers of every PolAR pixel in a given ring for each ring. Furthermore, the variance of Cartesian pixels mapping to PolAR pixels can also be obtained.

The Models

We are now ready to introduce our three main models:

- Slice Approach (PolAR-S): the entire slice is used in the search
- Subslice Approach (PolAR-SS): **TWO** (2) or **THREE** (3) sub-regions of a slice are used in a correlated search

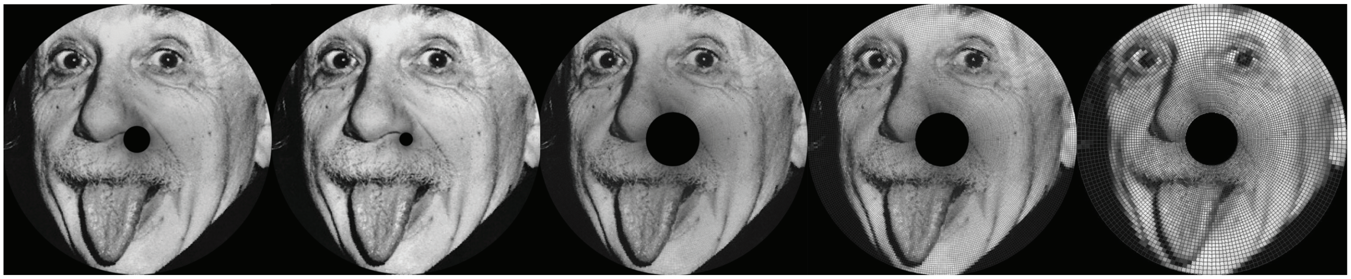

Figure 2: Example of PolAR Pictures of the famous photograph of Einstein, used here in size 250x250 pixels. "h_n" refers to size of the outermost PolAR pixel. "br" refers to ratio of the central blindspot and PolAR Picture diameters. For each PolAR Picture, (m x n) denotes that it has m wedges and n rings. **From left to right:** (a) PolAR Picture (h_n = 1, br = 0.1, 782 x 287). (b) PolAR Picture (h_n = 1, br = 0.05, 782 x 373). (c) PolAR Picture (h_n = 1, br = 0.2, 782 x 201). (d) PolAR Picture (h_n = 2, br = 0.2, 389 x 100). (e) PolAR Picture (h_n = 4, br = 0.2, 193 x 50).

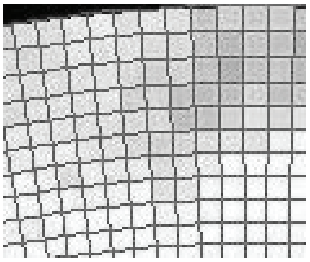

Figure 3: Closeup of PolAR Picture near fringes. Each "pseudo-square" is a PolAR pixel.

- Hybrid Approach (PolAR-H): **ONE** (1) sub-region is correlated with the slice

We will use the harder multi-slot sample problem (Figure 1, right) to illustrate these models.

Algorithms 3, 4, and 5 offer high-level overviews of the PolAR-S, PolAR-SS, and PolAR-H respectively.

Algorithm 3 PolAR-S

```
1: Find slice's transparent pixels
2: Pad slice to make it square
3: Make slice a PolAR Picture
4: Perform ring-based density search with slice
5: Perform rotational search with slice
```

Algorithm 4 PolAR-SS

```
1: Find slice's transparent pixels
2: Find subslices based on non-transparent variance
3: Make subslices PolAR Pictures
4: Perform ring-based density search with primary subslice
5: Perform rotational search with primary subslice
6: Perform correlative search with other subslice(s)
```

Algorithm 5 PolAR-H

```
1: Find slice's transparent pixels
2: Find subslice based on non-transparent variance
3: Make subslice and slice PolAR Pictures
4: Perform ring-based density search with subslice
5: Perform rotational search with subslice
6: Perform correlative search with slice
```

Transparent pixels. Each algorithm begins with finding the "slice's transparent pixels". While the slices on each card are surrounded by white space, the corresponding region on the easel is embedded within a background that is not necessarily white. The pixels comprising this white space that are mapped to non-white easel pixels are thus dubbed "transparent". To find these, a subroutine starts in the white corner of the slice and performs a BFS, which identifies all white pixels not enclosed by the slice as transparent. Subsequently, these pixels are ignored by our pixel-based similarity metric.

Variance-based subslice selection. Line 2 of Algorithms 4 and 5 state "Find subslice(s) based on non-transparent variance". By default, the square subslices are 35x35 pixels (unless the slice is too small in which case the subslice size is the minimum dimension of the slice). The algorithm iterates through all possible subslices and returns the subslice whose non-transparent pixel values yield a maximal variance. This is the "primary subslice" in the case of PolAR-SS and the sole subslice for PolAR-H. The motivation is that variance is often a heuristic for identifying "the most interesting" subslices; for instance, boundaries between lighter and darker regions will tend to be selected. **Note:** Transparent pixels are excluded from this analysis because there is no guarantee that their value is white.

PolAR-SS (Algorithm 4) creates two or more subslices. The primary subslice is chosen as explained above. To fully leverage the correlative search, we wanted to choose subsequent subslices that were sufficiently far apart from one another to maximize diversity of information. Thus, the list of all possible subslices (ordered by variance) is iterated through until a given subslice is at least a certain distance from all subslices chosen so far. This process succeeds at selecting distinct features for each of the chosen subslices.

Figure 4 showcases this process on one of the slices shown in Figure 1, right.

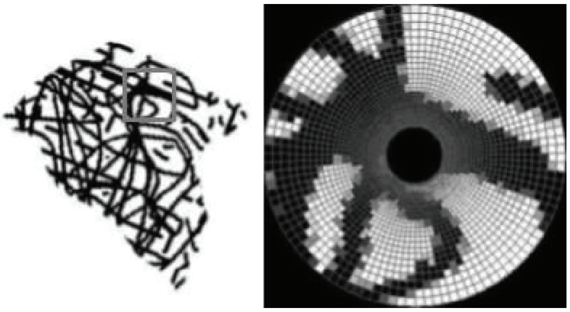

Figure 4: Example of variance-based subslice selection. Left: slice with gray box outlining region with highest non-transparent variance. Right: the PolAR Picture for this region.

Ring-based density delta search. This is shared by all three algorithms. The target region is the entire slice, primary subslice, or only subslice for PolAR-S, PolAR-SS, and PolAR-H respectively. For each easel slot, all subregions with size equal to the target region are iterated through and a priority queue maintains the "most similar" subregions. Without the existence of transparent pixels, the "most similar" heuristic is quite simple: for each ring (and the central blindspot as well), the search algorithm determines which Cartesian pixels map to it and calculates its average. This average can be thought of as a "density", hence the name of this subsection. These ring-based densities are calculated for both the target region and the given easel slot subregion and the deltas are calculated in between the two. Finally, the result that is returned is the average of these deltas such that all rings are uniformly weighted whereas the central blindspot has double the weight. The blindspot is weighted more heavily because its information cannot be used during the rotational search. As such, it is intuitive to make it a more significant factor.

The motivation behind this heuristic is that it is rotation-invariant; a PolAR picture returns the same set of densities regardless of its orientation. It also yields more information than just a single density representing the average of all pixels in the rings and blindspot. The rotation-invariance is key because the alternative is prohibitively expensive; the algorithm would have to iterate through all orientations for all possible subregions. With our approach, this initial ring-based density search chooses the most likely candidates for each easel slot to then explore more thoroughly in the subsequent rotational search.

Incorporating transparent pixels makes this process slightly more complex. For each ring (and the blindspot), the density is calculated assuming that all transparent pixels are equivalently white (value of 0) or black (value of 1). The former establishes a lower bound for what the density could be and the latter a higher bound. Then when calculating the density deltas, if the corresponding ring (or blindspot) density for the easel slot subregion is within these bounds, then the delta is zero; otherwise the delta is the distance between this density and the closest bound. This allows some flexibility but can also lead to too much ambiguity, as we will explore in the discussion section.

Rotational search. This is also common to all three algorithms and is performed using the same target region and the easel slot priority queues described in the preceding ring-based density search subsection. This part is straightforward. For each easel slot, the algorithm iterates through each likely candidate in the priority queue and performs a rotational analysis. This entails rotating the primary target's PolAR picture through all possible orientations (a discrete number established by the number of wedges) and comparing it to the candidate's PolAR picture using a given similarity metric. To reiterate, in this paper the MaxMin metric is used, as described in the earlier "Notion of Similarity" section. The analysis returns the orientation of the primary region that maximizes the similarity metric w.r.t. the candidate.

Correlative search. This occurs immediately after the rotational search fixes the orientation of the target region for a given candidate subregion. For each easel slot, the candidate yielding the best "composite" similarity is returned. This search is found in the PolAR-SS and the PolAR-H.

For PolAR-SS, recall that the target region is the primary subslice. Thus with the location and orientation of the primary subslice fixed in the easel slot, the algorithm can then locate and orient where the secondary subslice should be. (**Note:** It accounts for errors arising from using a computational model that discretizes rotation.) The primary and secondary subslices are now fixed. If only two subslices are considered, then the search is done. A composite similarity is calculated that is the average of the primary and secondary similarities. This average is weighted more heavily towards the secondary similarity (in this paper, the ratio was 1:5 primary : secondary). The rationale is that while the feature selected in the primary subslice may be found in many easel slot subregions, the chances that its corresponding secondary subslice would also exhibit relatively high similarity are low unless the match is a correct one. Thus, the secondary subslice is the main deciding factor.

If PolAR-SS considers three subregions, then now the tertiary subslice is located and oriented based upon the location of the secondary subslice. As per the same rationale as the two-region case, the composite similarity is computed with heavier weighting on the auxiliary subslice similarities (in this paper, the ratio was 1:3:5 primary : secondary : tertiary).

As for PolAR-H, the primary region is the one and only subslice. With this fixed, the placement of the entire slice within the easel slot can now be found. (**Note:** as before, the algorithm accounts for discrete rotation errors. Additionally, due to size differences, the PolAR Pictures for the slice and subslice use a different number of wedges. This means there is some extra work to locate and orient the slice in the easel.) The correlative search is similar to that of PolAR-SS. As before, for each slot, the candidate yielding the best composite similarity (weighted towards the slice similarity; in this paper, the ratio was 1:5 subslice : slice) is chosen.

The PolAR-S has no additional correlative search. Its primary region is the slice. With this fixed, the candidate with the highest similarity is returned for each slot.

Output. At this stage, for each easel slot the most likely candidate (i.e. most likely location and orientation of the primary region in the slot) has been found. A detail that has been left out is that this process happens twice in parallel: once for the original slice and another for a mirror image of the slice. This is because some problems (cannot say exactly which) feature reflections in addition to rotations. Therefore, for each slot, there are two most likely candidates: original and flipped. Whichever yields the best composite similarity remains and the other is discarded. The algorithm then outputs the slot with the best candidate, as well as all slot candidates, and visualizes this info.

Note regarding similarity: We have used the term "similarity" throughout this section. This concept is ambiguous and the "correct" metric to use is a context-dependent design decision. We settled on a pixel-based similarity metric, dubbed "Max-Min":

Let A,B be matrices representing grayscale images, each pixel value normalized to the range 0.0 (white) to 1.0 (black). Here are the three metrics/pseudo-metrics studied in this paper that yield 0 and 1 for minimum and maximum similarity respectively:

$$MaxMin(A, B) := \frac{sum(min\{A, B\})}{sum(max\{A, B\})}$$

$$= \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} min\{a_{ij}, b_{ij}\}}{\sum_{i=1}^{m} \sum_{j=1}^{n} max\{a_{ij}, b_{ij}\}}$$

We intend to explain our motivations, as well as other approaches, in a full-length journal paper.

As an example, Figure 5 shows the visual output of PolAR-H for the slice shown in Figure 4. The easel slot shown is in fact the correct slot.

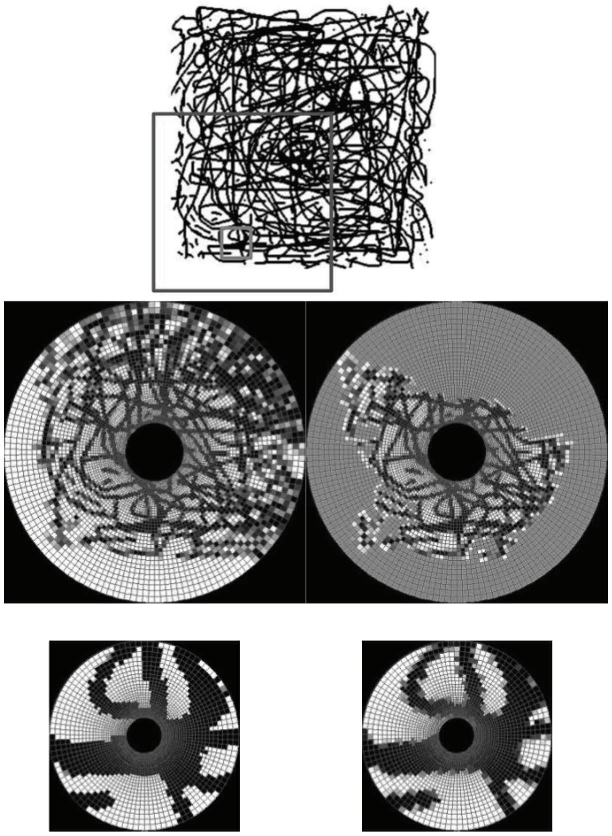

Figure 5: Output of PolAR-H for the slice in Figure 4. Top row: easel slot overlaid with dark gray box and lighter gray box representing the locations of the slice and subslice respectively. Middle row: PolAR Picture for easel slot subregion where slice is located (left); rotated PolAR Picture of original slice surrounded by "sea" of transparent pixels (right). Bottom row: PolAR Picture for easel slot subregion where subslice is located (left); rotated PolAR Picture of original subslice (right).

Results

Now we present the results of applying our three main models (PolAR-S, PolAR-SS, PolAR-H) to the Leiter Form Completion subtest (Problems 03 through 14). (Problems 1 and 2 are intended as simple examples for human test-takers, and Problem 15 was of a somewhat different type and so was omitted here.) In actuality, there are four sets of results because PolAR-SS is split into PolAR-SS2 (two subslices) and PolAR-SS3 (three subslices). Problem inputs for all the models were scanned directly from an official copy of the Leiter-R test. Slices were segmented manually.

Figure 6 (left) displays the problem-wise accuracies of the models in localizing the slices associated with that problem. Recall that Problems 03 through 08 are single-easel which means that the model just has to locate where the slice is in the easel and how it is oriented to be correct. For Problems 09 through 14, however, there are multiple easel slots to choose from. To be correct, the model has to identify the proper easel slot and then find the proper location/orientation of the slice within that slot. We would give a more a detailed breakdown if not for the necessity of maintaining the Leiter's test privacy.

Figure 6 (right) offers a different metric of accuracy that is more lenient. Each problem has a certain number of choices and each choice has a certain number of slices. Given a choice, if at least one of its slices is correctly located, then that choice is also considered to be correctly located.

Lastly, it should be noted that with regard to the hard sample multi-slot problem (Figure 1, right), all four models localize the slices and choices with 100% accuracy.

Discussion

At first glance, the results admittedly do not look terribly promising. The multi-slot Problems 09 through 14 have four to five easel slots. Thus a model that randomly tries to pair slices with the correct slot would be 20% to 25% accurate. However, within each slot, there are on the order of a million combinations of location and orientation. Thus, the slice accuracies shown in Figure 6, although seemingly low, are astronomically better than random. The single-easel Problems 03 through 08 have similarly many combinations.

What these numbers do not convey are all the near-misses.

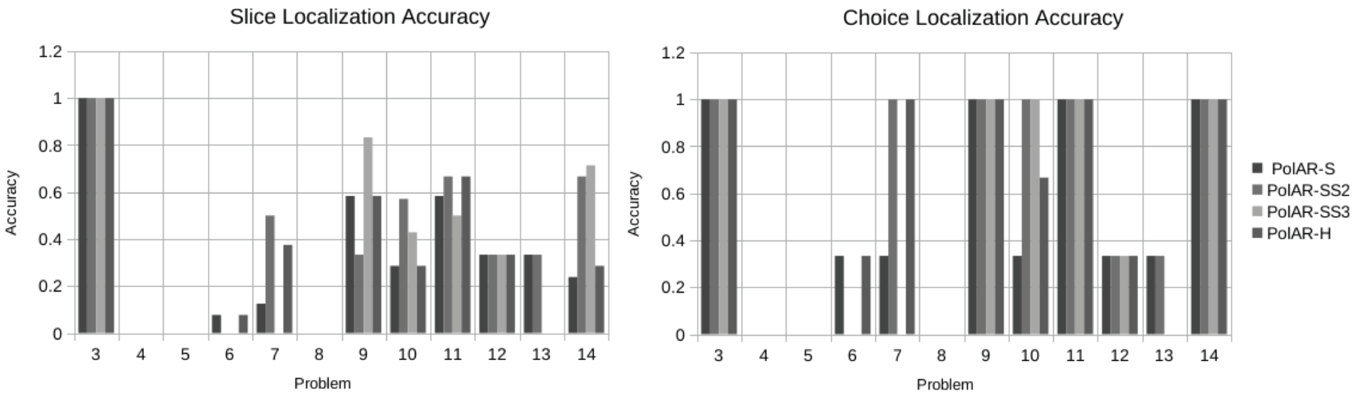

Figure 6: Left: Displays the problem-wise accuracies of the models in localizing slices among all the easel slots. Right: Displays the problem-wise percentages of choices where the models successfully locate AT LEAST ONE of the choice‘s slices.

For example, often the models will find the slices in their correct slots for the multi-slot problems. However, another incorrect slot might have a candidate with a slightly better composite similarity and thus this incorrect slot gets chosen. This was especially prominent in Problem 09 with PolAR-SS2. Problem 09 is characterized by simple geometric shapes (mostly rectangles). Thus there are many identical right angles that confused PolAR-SS2 (as shown by its relatively poor performance in Figure 6). It would locate a slice correctly within its slot but due to random aberrations in printing, right angles in another slot are considered to have a slightly greater similarity. With this in mind, it makes sense then that PolAR-SS3 (i.e. the inclusion of a third subslice) performed significantly better than PolAR-SS2; three subslices entails three corners (because corners have the greatest variance) which defines an entire rectangle, whereas two subslices can only define a more generic line segment with corners at either end.

The most striking performance hit is seen with many of the earlier single-easel Problems 03 through 08. In particular, no model succeeded in locating ANY slices for Problems 04, 05, and 08. Even though the same scanner and settings were used for both the slices and the easels, the colors came out differently; there are differences in printing on the publisher‘s end. The color differences meant that the correct locations were not even ending up in the priority queues. Ideally, the models would be robust enough to identify similarity between two regions with different hues (i.e. a red rectangle is similar to a blue rectangle). However, the similarity metric we used was pixel-based and thus quite sensitive to these differences.

Another performance hit is related to the transparency analysis. While it increases the flexibility of searching, it can also eliminate information and make the density delta search too ambiguous. For example, especially in Problem 14, slices can contain many white interior pixels that are not completely enclosed by a border. As such, these pixels erroneously get labeled as transparent and too little information remains. This suggests we need to revisit how we determine transparency. PolAR-SS outperforms in this scenario because it is purely subslice-based and subslices are chosen specifically to avoid transparent pixels.

As for good news, the hard sample multi-slot problem (Figure 1, right) showcases the power of the models. Recall that all models had 100% success rate for this problem. On one hand, the conditions are perfect for the PolAR models; there are no differences in either scaling or shading — the slices are copies of their corresponding slot subregions. However, the slots and the slices have so much noise that a human would have a very difficult time solving it. Imagine just being given the slice in Figure 4 and arriving at the solution shown in Figure 5. These successes indicate that the underlying approaches have merit; however, it remains to make them more robust to compete with human level performance. The other good news is that the choice localization results seem promising; as Figure 6 shows, many of the problems have 100% accuracy across all four models. The performance drop on some of the earlier problems as well as Problems 12/13 can be explained by differences in shading and scaling respectively.

Contributions and Future Work

The PolAR representation is geared towards mental rotation specifically, but mental rotation is only one capability that is needed within a broader computational framework of imagery-based reasoning (Kosslyn and Shwartz 1977). Other imagery-based operations, such as translation, zooming, and image composition, are not currently implemented but could easily be incorporated into the PolAR framework, and in fact have been explored in previous models by our team (Kunda, McGreggor, and Goel 2013). While these additional operations do not appear to be strictly necessary for the FC test, they may be required for other Leiter-R tests, and also may contribute to improved/different levels of performance on the FC test as well.

We describe PolAR and associated algorithms here as a “computational cognitive model.” While our model is not necessarily cognitively plausible in a strict sense, we do know that humans perform many similar imagery-based operations to those performed by our model, though the exact format of representations and operations will differ in the human brain. Whether people solve FC problems in a way that is, at a high level, similar to our model is an open question, and will require more detailed comparisons to human data (e.g. eye-tracking, error analysis, etc.). We see the model in this paper as one of many possible strategies that people may use to solve FC problems. Eventually, we hope that models such as this one will help illuminate and explain individual differences in cognitive strategies on tests like the FC test, especially in cognitive conditions like autism. We also plan to tackle the other Leiter-R subtests, using the current model as a starting point.

One specific direction of interest is looking into "generative" approaches to FC. Our current FC approach is "discriminative" because the choices are decomposed into slices, and then each is individually searched for in the slots; the algorithm tries to discriminate where constituent pieces are located. A generative approach, on the other hand, would attempt to *first* combine all the choice‘s slices into a complete object, and *then* search for that object in the easel. This would serve as a useful counterpoint to the current discriminative model; our hypothesis is that humans employ both of these strategies, perhaps switching based on a person's developmental stage or the complexity of a given problem. We expect to explore this hypothesis through a combination of continued computational experiments combined with human studies.

Acknowledgments

This research was supported in part by the National Science Foundation (Grant #1730044) and by the Vanderbilt Discovery Grant program. Many thanks to Calum Hartley, who brought our attention to the Leiter-R test through discussions of his research on cognitive development in autism.

References

Ainooson, J., and Kunda, M. 2017. A computational model for reasoning about the paper folding task using visual mental images. In *Proceedings of the 39th Annual Conference of the Cognitive Science Society, London, UK*.

Carpenter, P. A.; Just, M. A.; 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.

Dawson, M.; Soulières, I.; Ann Gernsbacher, M.; and Mottron, L. 2007. The level and nature of autistic intelligence. *Psychological science* 18(8):657–662.

Funt, B. V. 1980. Problem-solving with diagrammatic representations. *Artificial Intelligence* 13(3):201–230.

Glasgow, J., and Papadias, D. 1992. Computational imagery. *Cognitive science* 16(3):355–394.

Glasgow, J.; Narayanan, N. H.; and Chandrasekaran, B., eds. 1995. *Diagrammatic Reasoning: Cognitive and Computational Perspectives*. Cambridge, MA, USA: MIT Press.

Kaplan, R. M., and Roid, D. P. S. 2010. *Psychological Testing: Principles, Applications, & Issues, Eighth Edition.* Belmont, California: Wadsworth.

Kosslyn, S. M., and Shwartz, S. P. 1977. A simulation of visual imagery. *Cognitive Science* 1(3):265–295.

Kunda, M., and Ting, J. 2016. Looking around the minds eye: Attention-based access to visual search templates in working memory. *Advances in cognitive systems* 4:113–129.

Kunda, M.; El Banani, M.; and Rehg, J. M. 2016. A computational exploration of problem-solving strategies and gaze behaviors on the block design task. In *38th Annual Conference of the Cognitive Science Society, Philadelphia, USA*.

Kunda, M.; McGreggor, K.; and Goel, A. K. 2013. A computational model for solving problems from the ravens progressive matrices intelligence test using iconic visual representations. *Cognitive Systems Research* 22:47–66.

Lovett, A.; Forbus, K.; and Usher, J. 2010. A structure-mapping model of raven's progressive matrices. In *Proceedings of the Cognitive Science Society*, volume 32.

Mel, B. W. 1986. A connectionist learning model for 3-d mental rotation, zoom, and pan. In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, 562–71.

Memisevic, R., and Hinton, G. 2007. Unsupervised learning of image transformations. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, 1–8.

Rao, R. P.; Zelinsky, G. J.; Hayhoe, M. M.; and Ballard, D. H. 2002. Eye movements in iconic visual search. *Vision research* 42(11):1447–1463.

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

Roid, G. H., and Miller, L. J. 1997. *Leiter International Performance Scale-Revised: Examiner's Manual*. Wood Dale, Illinois: Stoelting Co.

Seepanomwan, K.; Caligiore, D.; Baldassarre, G.; and Cangelosi, A. 2013. Modelling mental rotation in cognitive robots. *Adaptive Behavior* 21(4):299–312.

Smith-Spark, J.; Fisk, J.; Fawcett, A.; and Nicolson, R. 2003. Investigating the central executive in adult dyslexics: Evidence from phonological and visuospatial working memory performance. *European Journal of Cognitive Psychology* 15(4):567–587.

Strannegård, C.; Cirillo, S.; and Ström, V. 2013. An anthropomorphic method for progressive matrix problems. *Cognitive Systems Research* 22:35–46.

Tabachneck-Schijf, H. J. M.; Leonardo, A. M.; and Simon, H. A. 1997. CaMeRa: A computational model of multiple representations. *Cognitive Science* 21(3):305–350.

Wai, J.; Lubinski, D.; and Benbow, C. P. 2009. Spatial ability for stem domains: Aligning over 50 years of cumulative psychological knowledge solidifies its importance. *Journal of Educational Psychology* 101(4):817.