In Proceedings of ICCM - 2007- Eighth International Conference on Cognitive Modeling. 25 - 30. Oxford, UK: Taylor & Francis/Psychology Press.

Towards Incorporating Visual Imagery into a Cognitive Architecture

Scott D. Lathrop (slathrop@umich.edu)
Computer Science and Engineering, 2260 Hayward Street
Ann Arbor, MI 48109-2121 USA

John E. Laird (laird@umich.edu)
Computer Science and Engineering, 2260 Hayward Street
Ann Arbor, MI 48109-2121 USA

Abstract

This paper presents a synthesis of cognitive architecture and visual imagery. Visual imagery is a mental process that relies both on cognitive and perceptual mechanisms and is useful for tasks requiring visual-feature and visual-spatial reasoning. Using visual imagery as motivation, we have extended the Soar cognitive architecture to support the construction, transformation, generation, and inspection of visual representations for general problem solving. This paper presents the high-level architectural design and discusses initial results from two domains.

Keywords: Cognitive architecture; visual imagery; multi-representational reasoning.

Introduction

Cognitive architecture research focuses primarily on abstract, symbolic representations and computations. Non-symbolic representations are used, but for control, and not for representing or manipulating task knowledge. There is, however, significant evidence that visual imagery plays an important role in many cognitive tasks (Kosslyn, et al., 2006; Barsalou, 1999). Our work seeks to investigate the synthesis of and interactions between cognition and mental imagery by extending the Soar cognitive architecture with visual imagery. In addition to Soar's native symbolic representation, visual imagery in our architecture uses a *depictive* representation as well as an intermediate, *quantitative* representation for images.

Our major result is a computational implementation of visual imagery and integration within a cognitive architecture. Functionally, this provides a computational advantage and additional capability for visual-feature and visual-spatial reasoning. Although our design is based on psychological and biological constraints, at this point, visual processing algorithms are ad hoc, and do not model the details of human performance. Our results illustrate the functional value of visual imagery and the challenges of creating complete models of such complex processes.

Related Work

Two of the most prominent cognitive architectures, EPIC (Kieras & Meyer, 1997) and ACT-R (Anderson et al., 2004), incorporate models of human perceptual and motor systems. However, rather than specifying and implementing the low-level details of perception and motor processing, (e.g. edge detection, joint coordinates), these systems focus on the timing and resource constraints between perception, cognition, and motor processing. Moreover, neither system has a long-term perceptual memory, which is necessary to gain access to a remembered object's visual features (i.e. shape representation). Neither system has any mechanism to support visual imagery.

Previous efforts to build computational models of imagery have not included the constraints that arise in integration with a general cognitive architecture. Kosslyn composed a detailed mental imagery model and created a computational implementation to simulate and test his ideas (1980). Glasgow and her colleagues built a computational model of imagery for a molecular scene analysis application (Glasgow & Papadias, 1992). While Glasgow incorporated psychological constraints in her model, such as the inclusion of three separate representations (descriptive, spatial, and visual), their implementation is application specific.

The CaMeRa model of Tabachneck-Schijf's et al. (1997) uses multiple representations and simulates the cognitive and visual perceptual processes of an economics expert teaching the laws of supply and demand. Their system includes both visual short-term and long-term memories that complement verbal memories, but the generality of the overall architecture is unclear. Visual STM includes a quantitative (node-link structure) and a depictive (bitmap) representation that is similar in design, although not in implementation, to our representations. Their shape representation is limited to algebraic (i.e. lines and curves) shapes and their spatial structure only models an object's location while ignoring orientation and size.

Barkowsky (in press) proposes that any model of mental imagery must include the following:

(1) Hybrid representational formats to include propositional and visual structures involving shape.
(2) Coupling between imagery and visual perception.
(3) Construction of images from pieces of knowledge.
(4) Processing with or without external stimuli.
(5) Multi-directional distributed processing and control.

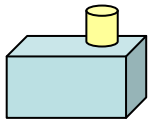
Our architecture addresses (1) – (3) and our future plans include incorporating visual imagery processing in the presence of perceptual stimulus (4). Our control structure initiates and controls imagery processes in a top-down manner while perceptual mechanisms process results in a bottom-up fashion. In Soar, the contents of working

memory determine which memories and processes are active without any centralized control (5). We also propose that the architecture must support transformation and generation of a depictive representation. The following sections discuss our initial implementation.

Visual Representations

We assume visual imagery uses three distinct visual representations to include (1) an abstract symbolic representation, (2) a hybrid symbolic and quantitative representation, and (3) a depictive representation (Table 1). Each visual representation becomes more specific and committal as you move down the hierarchy.

Table 1: Visual Representations

Representation	Uses	Example
Abstract symbols	General, qualitative visual-feature and visual-spatial reasoning	object (can) object (box) color (can, yellow) color (box, blue) on (can, box)
Hybrid abstract and quantitative symbols	Quantitative visual-spatial reasoning	can height 5 radius 1 location <2,1,2> box length 10 width 6 height 4 location <0,0,0>
Depictive symbols	Visual-feature recognition Quantitative visual-spatial reasoning	

The abstract symbolic visual representation is the neutral, stable medium useful for general reasoning (Newell, 1990). Symbols denote an object, some visual properties of that object, and qualitative spatial relationships between objects. The meaning of the symbols is dependent on their context and interpretation rather than how the symbols are spatially arranged. The symbols are composable using universal and existential quantification, conjunction, disjunction, negation, and other predicate symbols.

The hybrid, intermediate representation labels objects with abstract symbols and denotes each object's location, orientation, and size with quantitative, vector-based values. The computational processes that infer information from this representation are sentential, algebraic equations.

The intermediate representation does not receive much attention in the imagery representational debate (Kosslyn, et al., 2006; Pylyshyn, 2002). However, it is important for the following reasons. First, neurological evidence shows that during visual-spatial imagery tasks, the visual cortex, or depictive representation, is not active (Mellet et al., 2000). However, the parietal cortex is active signifying a visual format distinct from the depictive representation.

Second, Marr stresses that bottom-up visual processing uses incremental, increasingly abstract levels of representations (Marr, 1982). This rational is also pertinent to visual imagery but in the "opposite" direction. Visual imagery cannot generate a depictive representation directly from qualitative, abstract symbols without first specifying metric properties, such as location, orientation, and size. Finally, from a computational perspective, there are some spatial reasoning tasks where reverting from qualitative symbolic representations to quantitative information is necessary for either efficiency or simply to infer new information (Forbus, Neilsen, & Faltings, 1991).

The depictive representation is useful for detecting object features (e.g. "does the letter 'A' have an enclosed space?") and spatial properties where the objects' topographical structure is relevant (e.g. "which is wider in the center, Michigan's lower peninsula or the state of Ohio?"). Space implies spatial extent within and between objects in a visual scene. Each point in the representation can have variable color and intensity, and the spatial arrangement of the points resembles the object(s) specific shape. Computationally, the depiction is a pixel-based data structure and the algorithmic processes are either algebraic or ordinal algorithms that take advantage of the topological structure.

Architecture

There are two software components in our architecture, (1) Soar and (2) Soar Visual Imagery (SVI). Soar provides the underlying control (via its procedural production memory and its decision procedure) and state representation (via its symbolic memories). SVI encompasses both visual perception and visual imagery mechanisms. Figure 1 shows the architecture with Soar (not to scale) across the top and the visual mechanisms inherent to SVI underneath. We will refer to this figure as we explain the architecture and elaborate on the specific visual imagery processes not shown in it. The architecture makes a distinction between memories (rectangles) and processes (rounded rectangles). The terminology is either Kosslyn's et al. (2006) or our own. We will start by explaining the memories and processes associated with visual perception working from the bottom to the top of Figure 1. Then we will discuss visual imagery from a top-down perspective.

Visual Perception

The *Visual Buffer* is the SVI short-term memory associated with the visual cortex. It maintains the depictive representation (Kosslyn, et al., 2006). A *Refresher* process activates the depiction based on information received from visual perception. Two sets of processes in SVI correspond to the ventral or "what" pathway and the dorsal or "where" pathway that extend from the visual cortex (Ungerleider & Mishkin, 1982). The *"What" Inspectors* are responsible for extracting object features, shape, and color from the Visual Buffer. They store each object's shape and color in a *Visual long-term memory* (LTM), neurologically believed to be in

the region of the inferior temporal lobe. SVI stores the shapes as a mesh topology in the Euclidean space, R^3.

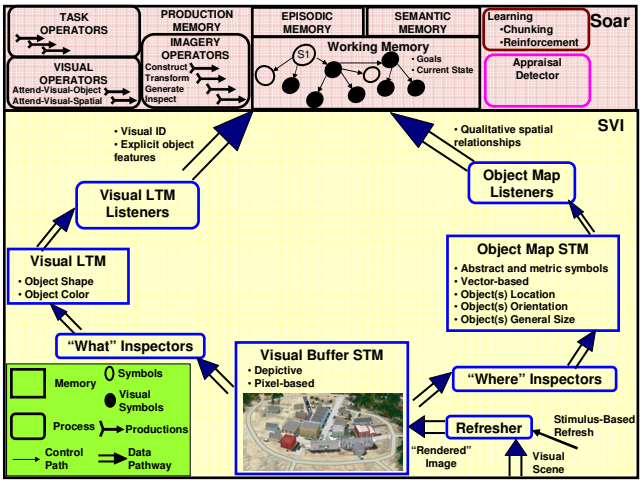


Figure 1: Architecture overview

The *"Where" inspectors* extract the location, orientation, and size of the objects in the Visual Buffer and store this information in the *Object Map* short-term memory. The Object Map roughly corresponds with the posterior parietal cortex and maintains the quantitative visual representation from Table 1. SVI implements this representation with a scene-graph data structure.

The *VisualLTMListeners* and the *ObjectMapListeners* consolidate the inspectors' results and create an abstract symbolic format for Soar's working memory. The Visual LTM Listeners provide an object's qualitative features along with a symbol *(visual-id)* denoting the object's shape and color in Visual LTM. Likewise, the Object Map Listeners create the qualitative spatial relationships between objects in the Object Map. Visual operators in Soar's production memory attend to the listeners input and associate it with existing knowledge.

Visual Imagery

For illustration, consider a Soar robot setting the table for dinner. Its current goal is to set one place setting, and in order to accomplish the goal it has to set each individual object (napkin, fork, plate, etc). It prefers to set the center object (i.e. plate) first so it can place the other objects relative to the center. The robot's working memory contains the symbolic representation of the place setting (Figure 2).

Each object's symbol structure is associated with the current state in Soar's working memory via a *visual-object* attribute. The place setting structure includes the primitive visual objects napkin, fork, plate, knife (not shown), and spoon (not shown) objects. Primitive visual objects have a visual-id attribute. Composite visual objects (i.e. place setting) denote an object containing other visual objects. Composites are augmented with *has-a* and *spatial-relationship* attributes defining how the object is composed. Spatial relationships indicate an object's location and topology in relation to other objects. For example, the fork is above (location) and connected (topology) to the napkin and left-of and disconnected from the plate. A viewpoint attribute specifies the spatial relationship perspective. Note that primitive objects may be associated with many composite objects and task knowledge may rearrange the spatial relationships or even synthesize composite objects to enable the creation of novel visual images.

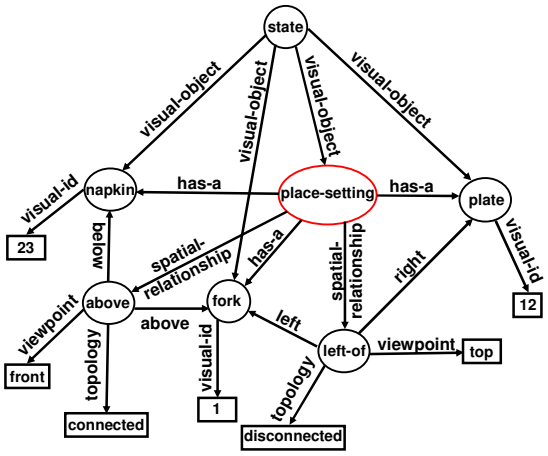


Figure 2: Soar working memory visual representation

Although the symbol structure in Figure 2 encapsulates a lot of information, it does not indicate the place setting's center object—either directly or through inference. When there is a lack of visual-feature or visual-spatial knowledge, an impasse occurs and Soar creates a special, visual imagery state. The state's initial knowledge consists of the symbolic representation of the object in question and the goal is to determine the desired information.

As a first step, visual imagery, processing has to re-encode Soar's symbolic representation into the intermediate, quantitative representation. To support this, general-purpose operators for *constructing* the metric representation (Figure 1) are encoded in Soar's production memory. Construction derives from a commonly demonstrated phenomenon in behavioral imagery experiments showing the time to generate a visual image is linearly dependent on the number of parts in the visual representation (Kosslyn, et al., 2006).

Within SVI, there are functional processes specific to imagery. The *Imager* receives the operator's command and symbolic information from Soar, interprets it, and passes the required information to a *Constructor* process (Figure 3). The Constructor builds the quantitative representation in the Object Map by combining each object's general shape information from Visual LTM with its qualitative spatial knowledge from Soar's working memory. For example, to build the place setting, visual imagery may first compose the fork and the plate by locating the fork to the left of the plate. In a similar fashion, processing adds the other objects to complete the quantitative representation.

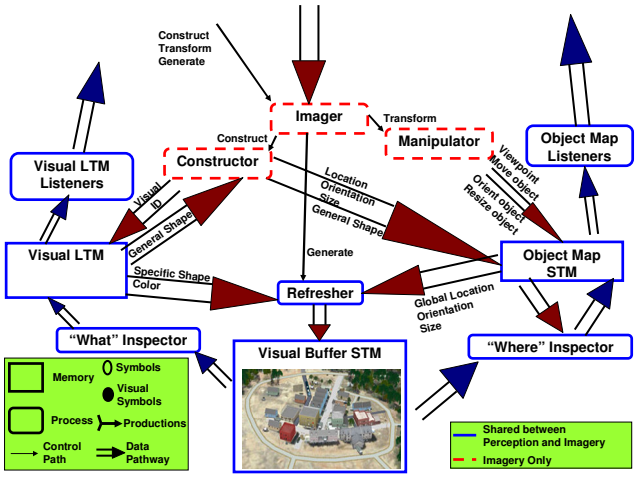


Figure 3: Construction, transformation, generation

The *transformation* operator (Figure 1) and the *Manipulator* process (Figure 3) emerge from another common behavioral phenomenon, made famous by Shepard's and Metzler's "mental rotation" experiment (1971). The operator changes the location, orientation, or size of a specific object or the perspective of the scene.

If the original query refers to an object's spatial orientation or relative size then the metric representation is sufficient. In the case of inferring the place setting's center object, this is the case. However, if the robot finishes setting the plate and is ready to pick up the napkin, it may want to know the relative difference in width between the plate and napkin. In this case, a depictive representation with each objects' specific shape is required. The *generation* operator initiates processing, and the Imager interprets the command and invokes the Refresher (Figure 3). The Refresher combines each object's specific shape and color from Visual LTM with the Object Map information and generates the depictive representation in the Visual Buffer.

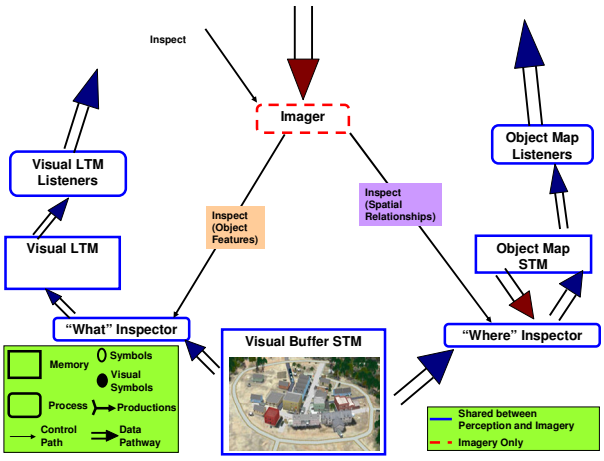


Figure 4: Inspection

After the system has constructed, transformed, and, if necessary, generated the depictive representation, the conditions are set for the inspection process (Figure 4). The *inspect* operator provides the Imager with the query. For example, "what is the center object of the scene?" or "which object is wider?" The Imager then activates the "What" and/or "Where" processes. These processes function as previously discussed with the exception that in visual imagery the Imager may direct the "where" inspectors to focus on the Object Map if the depiction is not required. The agent may iteratively add more detail to its visual representation and inspect it to refine its search.

Results

The results illustrate the functional and computational value of visual imagery in two distinct domains. The first domain derives from Larkin & Simon's work demonstrating the computational advantage of diagrams (1987). In the problem they investigate (Figure 5), the model must locate object features, (e.g. vertices, line segments, triangles) and infer relationships (e.g. angles, congruency) that initial task knowledge does not specify.

Although we doubt a human could solve the problem without an external diagram, we chose this task because it stresses the construction and inspection of a quantitative representation. The task does not require a depiction as initial knowledge specifies the main feature (lines) from which other features can be inferred. As either symbolic or metric representations are sufficient, we can compare them and determine computational and functional differences.

The second domain derives from Kosslyn and Thompson (2007). In this experiment, the subjects hear a letter from the English alphabet and the experimenters ask them to visualize it in its uppercase format. Next, the subjects hear a cue, such as "curve", "enclosed-space", or "symmetry" and indicate whether the letter has the particular feature. For example, the letter 'A' has an enclosed space and vertical symmetry while 'U' has a curve. The Soar model also "hears" a question, visualizes the letter, searches for the desired feature, and then "verbally" responds

We chose this visual-feature task because it involves all imagery processes and representations. Unlike the geometry domain, symbolic or quantitative representations cannot solve this task without explicitly encoding every feature. The task also includes an external environment that emphasizes the interaction of visual imagery and cognition.

Although our initial goals are functionality driven, we also make comparisons with human data and discuss the shortcomings. Two reasons for these shortcomings include our uncertainty of the types of algorithms humans use to recognize features, and our architecture's lack of "image maintenance" that occurs when the image's vividness decays and must be refreshed (Kosslyn et al. 2006).

Geometry Problem

The problem states that there are four lines (A, B, C, D). Line A is parallel to line B and line C intersects line A. Line D bisects the line segment formed by the intersection of line C with lines A and B (Figure 5). The goal is to show that the

two triangles formed are congruent. To prove congruency, the model must employ a basic geometry rule, such as the angle-side-angle (ASA) rule. The ASA rule states if two angles and the included side of a triangle are congruent to two angles and the included side of another triangle, then the two triangles are congruent. In Figure 5, the model must show E1=E2, e1=e2, and c=b.

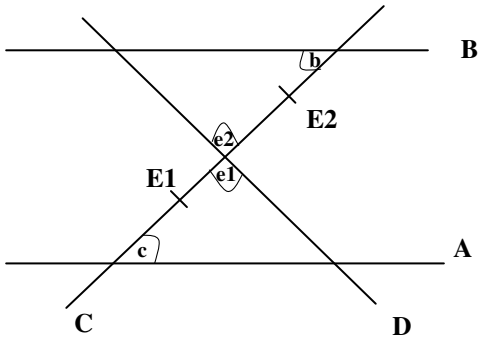


Figure 5: Geometry Problem

We compared two models. The first uses only symbolic representations (Soar Only) and has operators to create and process geometric objects and relationships. For example, "if two lines intersect, then create a vertex". The model creates these features until it can show the triangles are congruent. The second model (SVI) constructs a metric representation from the original description. It then inspects it for the desired features and relationships and uses the information along with the ASA rule to prove congruency.

The SVI model requires less real and simulated time (Figure 6). "Soar Only" spent much of its time considering objects and relationships that were not required to solve the problem. The SVI model also requires less task knowledge (Figure 7). The "Soar Only" model requires knowledge about geometric structures inherent to SVI's imagery operations. Functionally, this suggests that SVI decreases the amount of knowledge required to learn such a task.

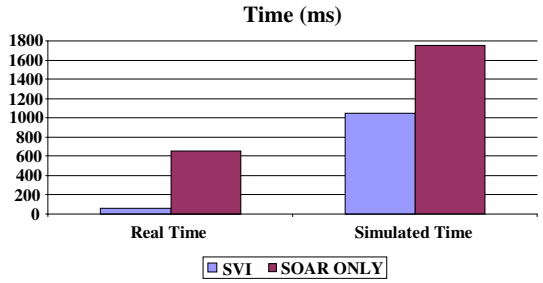


Figure 6: Time for each agent.
Simulated time is decision cycles x 50 ms.

The model is not psychologically plausible because of its unrealistic ability to maintain arbitrary amounts of information in its visual buffer. We expect humans would require an external diagram and thus require more time to solve the problem. However, the task demonstrates imagery's computational advantages and added capability.

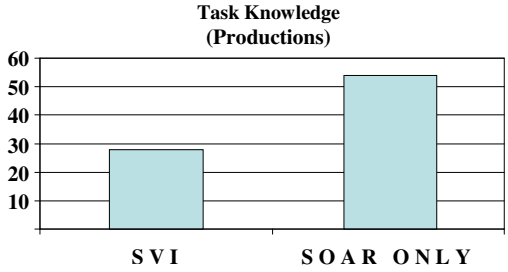


Figure 7: Initial task knowledge for each agent

Alphabet Experiment

Our evaluation for this experiment focuses on three areas. First, the requirement for generating and transforming depictive representations forced us to reconsider the design. Our previous discussion reflects this evaluation. Second, we make a subjective comparison between the feature detection algorithms and note that even though the representation is depictive, the processing may not. For example, to detect curves we use a variation of the Hough transform (Mat Jafri & Deravi 1994). The algorithm maps edge pixels onto a parameter space and uses a "voting" algorithm to determine the parameters that indicate a curve. Although the algorithm has interesting perceptual characteristics in that it is parallelizable, it uses sentential, algebraic computations. For detecting enclosed spaces, we employ an algorithm using pixel rewrites to take advantage of the topological space and locality of neighboring pixels that is clearly more "depictive" (Furnas, et al., 2000).

Finally, we compare the model's response time[1] (RT) with human data from Kosslyn's experiment (2007). Figures 8–9 show the comparison with the letters along the x-axis sorted from left to right according to human response time. Both humans and Soar show variability in the time to detect enclosed spaces[2], but the average time is almost identical (Figure 8). In the case of symmetry, however, Soar shows little variability while humans show a lot (Figure 9).

Again, we make no claim that the algorithms are similar to how humans recognize these features. Since the architecture does not incorporate image maintenance, the time required to recognize symmetry dominates the results. Our algorithm determines symmetry by transforming the original depiction around the axis of symmetry and comparing it with the original orientation. Rather than performing this operation in a single step, we hypothesize that humans must continuously rotate and regenerate the letter. This demonstrates that even if the overall architecture

[1] Based on average CPU time over 30 trials and scaled for comparison with human data.

[2] Curves and enclosed spaces show a similar graph with the exception that the range of response times were spread out more for both the human (~600ms) and Soar agent (~500ms) data.

is correct (our hypothesis), the devil of modeling human behavior is in the details of low-level visual processing.

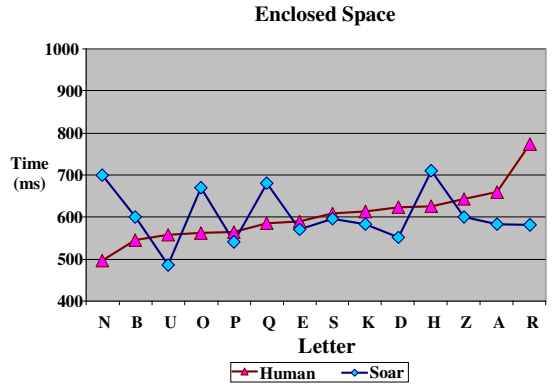


Figure 8: Enclosed space response time comparison
Human $\mu : 604, \sigma : 65$, Soar $\mu : 595, \sigma : 14$

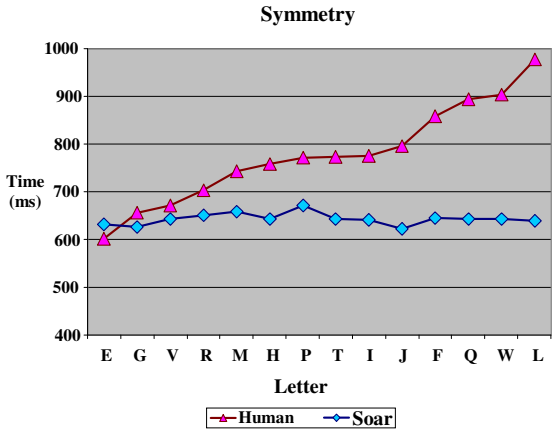


Figure 9: Symmetry response time comparison
Human $\mu : 778, \sigma : 104$ Soar $\mu : 643, \sigma : 12$

Conclusion

We have demonstrated that it is possible to extend a general cognitive architecture with a comprehensive model of imagery that includes using multiple visual representations; sharing mechanisms with vision; and incorporating construction, transformation, generation, and inspection. It also expands architectures by linking perceptual-based thought and cognition. This union provides new capabilities and computational efficiency for visual-feature and visual-spatial reasoning. As we move forward, we desire to expand the inspection processes and evaluate the architecture in an environment where perception and imagery interact, spatial and depictive forms of imagery are necessary, and the overall task is not to answer a question but involves making decisions and executing them in a rich environment.

Acknowledgments

The authors would like to thank Stephen Kosslyn and William Thompson for the human data and collaboration.

References

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An Integrated Theory of the Mind. *Psychological Review, 111*(4), 1036-1060.

Barkowsky, T. (in press). Modeling mental spatial knowledge processing: An AI perspective. In: F. Mast and L. Jaenke (Eds.), Spatial processing in navigation, imagery, and perception. Berlin: Springer.

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences, 22*, 577-660.

Forbus, K. D., Neilsen, P., & Faltings, B. (1991). Qualitative spatial reasoning: the clock project. *Artificial Intelligence, 51*(1-3), 417–471.

Furnas, G., Qu, Y., Shrivastava, S., & Peters, G. (2000). *The Use of Intermediate Graphical Constructions in Problem Solving with Dynamic, Pixel-Level Diagrams* (Vol. 1889).

Glasgow, J., & Papadias, D. (1992). Computational Imagery. *Cognitive Science, 16*, 355-394.

Kieras, D. E., & Meyer, D. E. (1997). An Overview of the EPIC Architecture for Cognition and Performance with Application to Human-Computer Interaction. *Human-Computer Interaction, 12*, 391-483.

Kosslyn, S. M. (1980). *Image and Mind*. Cambridge: Harvard University Press.

Kosslyn, S. M., & Thompson, W. L. (2007). *Can people "see" implicit properties as easily in imagery and perception? (In preparation)*.Unpublished manuscript.

Kosslyn, S. M., Thompson, W. L., & Ganis, G. (2006). *The Case for Mental Imagery*. New York, New York: Oxford University Press.

Larkin, J. H., & Simon, H. A. (1987). Why a Diagram is (Sometimes) Worth Ten Thousand Words. *Cognitive Science, 11*, 65-99.

Marr, D. (1982). *Vision*. San Francisco: Freeman.

Mat Jafri, M. Z., & Deravi, F. (1994, 2 November 1994). *Efficient algorithm for the detection of parabolic curves.* Paper presented at the Proceedings of SPIE - Vision Geometry III, Boston, MA, USA

Mellet, E., Bricogne, S., Tzourio-Mazoyer, N., Ghaem, O., Petit, L., Zago, L., et al. (2000). Neural Correlates of Topographic Mental Exploration: The Impact of Route versus Survey Perspective Learning. *NeuroImage, 12*, 588-600.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, Massachusetts: Harvard University Press.

Pylyshyn, Z. (2002). Mental Imagery: In search of a theory. *Behavioral and Brain Sciences, 25*, 157-238.

Shepard, R. N., & Metzler, J. (1971). Mental rotation of three-dimensional objects. *Science, 171*, 701-703.

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

Ungerleider, L. G., & Mishkin, M. (1982). Two cortical visual systems. In D. J. Ingle, G. M.A. & R. J. W. Mansfield (Eds.), *Analysis of visual behavior* (pp. 549-586). Cambridge, MA: MIT Press.