

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.

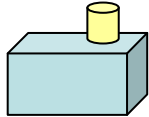
Our architecture addresses (1) – (3) and our future plans include incorporating visual imagery processing in the presence of perceptual stimulus (4). Our control structure initiates and controls imagery processes in a top-down manner while perceptual mechanisms process results in a bottom-up fashion.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

