Perception, memory, and the field of view problem

William S. Gribble

Department of Electrical and Computer Engineering
University of Texas at Austin, Austin, Texas, 78712
grib@cs.utexas.edu

Robust control of a vision-based agent requires tight coupling between sensing and action. For mobile robots performing visually-guided navigation, this means closed-loop control of motion with respect to sensed features, landmarks, or other relevant parts of the visible environment. Real vision sensors have limited fields of view. This makes true closed-loop control with respect to an arbitrary set of landmarks impossible with practical vision systems, since only a fraction of the environment can be seen at any one time.

My dissertation describes a solution to the *field of view problem* for vision-based agents lacking omnidirectional sensors. I propose a unified object memory system which integrates short-term working memory of the local *visual space* with immediate object perceptions from the real-time image stream. Short-term memory includes a model of agent and object dynamics and a recursive position estimator. The significance of unification of short-term memory and direct perception is twofold. First, since the positions and properties of all relevant objects are either directly sensed or estimated from recent experience, navigation control laws can be written as closed-loop controllers operating without regard to the agent's current field of view. Second, given an estimator such as the Kalman filter which includes an estimate of state uncertainty, an independent *investigatory action* scheduler can dynamically optimize shifts of camera field of view.

In existing closed-loop visual control (or *visual servoing*) systems, the agent's plan contains explicit instructions for control of all actuators. The camera platform's degrees of freedom are used to maintain visual lock on a target, and locomotor degrees of freedom are used to cause the robot to follow a particular path or perform an action based on its sensory input. Envisioned as a closed-loop control system, this arrangement has a real-time stream of pixels as its input, a vector of motor controls as its output, and a set of vision and control algorithms – the plan, plus state and numerous transformation algorithms – in between. If camera field-of-view changes are required, they must be explicitly programmed as a part of the plan. This can be awkward or impossible when multiple goals must be pursued simultaneously.

The unified object memory allows us to split this loop into two parts:

- The camera controller and vision system, which has a stream of pixel arrays as its input, a pool of grounded symbolic descriptions of relevant objects (the unified working memory) as its output, and a *task-independent* controller and vision system in between. Field of view shifts are scheduled automatically to minimize uncertainty in estimates of object position and properties.
- The agent's planning and control system, which has grounded symbols representing relevant objects as its input and locomotor controls as its output. Feedback to the camera controller and vision system exists in the form of add and delete instructions to the working memory: when a new control algorithm is activated, it adds a list of relevant objects to the working memory. Likewise, objects are removed from working memory when they are no longer needed by any control laws.

There are significant benefits from this arrangement. Most importantly, the field of view problem is eliminated. Access to the properties of objects is not impeded when the physical field of view is directed elsewhere. Since the unified object memory has an estimate of the object's position and dynamics, the agent can reason sensibly about any set of objects regardless of their spatial separation. Direct observation is still of critical importance, since estimates are inherently uncertain and this uncertainty accumulates over time if no new observations are made. However, the agent is not required to directly observe an object in order to use it as a source of feedback for closed-loop locomotor control.

Also, plans can be written without explicit programming of sensing, tracking, and attention-shifting control. Rather than controlling sensors directly, the plan specifies a policy for identifying and tracking relevant objects and explicitly programs only those behaviors, such as locomotor control, that are directly related to task completion. This is a qualitative change in the nature of the agent's plan, a change which makes the plan appear more like an abstract algorithm for task completion and less like a series of open-loop procedural commands.