LABORATORY NOTES

The iLab Neuromorphic Vision C++ Toolkit: Free tools for the next generation of vision algorithms

Because of its truly interdisciplinary nature—benefiting from the latest advances in experimental and computational neuroscience, electrical engineering, control theory, and signal and image processing—neuromorphic engineering is a very complex field. This has been one of the leading motivations for the development of a Neuromorphic Vision Toolkit at the iLab of the University of Southern California to provide a set of basic tools that can assist newcomers in the field with the development of new models and systems. More generally, the iLab Neuromorphic Vision C++ Toolkit project aims at developing the next generation of vision algorithms, the architecture of which will closely mimic the neurobiology of the primate brain rather than being specifically developed for a given set of environmental conditions or tasks. To this end, it provides a software foundation that is specifically geared towards the development of neuromorphic models and systems.

At the core of the toolkit are a number of neuroscience models, initially developed to provide greater understanding of biological vision processing, but here made ready to be applied to engineering challenges such as visually-guided robotics in outdoor environments. Taken together, these models provide general-purpose vision modules that can be easily reconfigured for, and tuned to, specific tasks. The gross driving architecture for a general vision system, the basis of many of the modules available in the toolkit, is shown in Figure 1.

Input video, whether captured by camera or from other sources, is first processed by a bank of low-level visual feature detectors. These are sensitive to image properties such as local contrast, orientation, or motion energy and mimic the known response properties of early visual neurons in the retina, lateral geniculate nucleus of the thalamus, and primary visual cortex. Subsequent visual processing is then split into two cooperating streams. The first is concerned with the rapid computation of the 'gist' and layout of the scene and provides coarse clues by which the system obtains a sense of the environmental conditions (e.g., indoors vs. outdoors, on a track vs. off-road) and of its position within the environment (e.g., path is turning left, the scene is highly cluttered). The second stream is concerned with directing both attention and the eyes towards the few most visually-conspicuous objects in the scene. This stage relies on a neural saliency map, which gives a graded measure of 'attractiveness' to every location in the scene, and is modeled on the neural architecture of the posterior parietal cortex in the monkey brain. At any given point in time, the system uses the gist for basic orientation in the scene and sequentially attends to interesting objects (which could be obstacles, landmarks to aid navigation, or target objects being looked for).

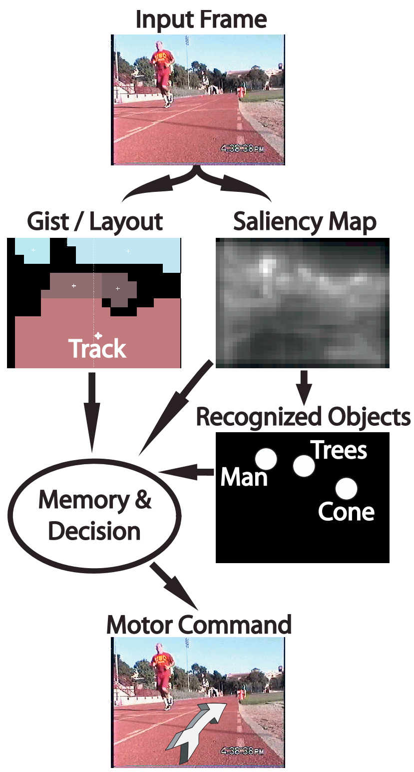

Figure 1. The iLab Neuromorphic Vision C++ Toolkit provides software modules for the implementation of some of the major components of primate vision, as depicted here. These include low-level visual processing algorithms, bottom-up visual attention and a saliency map, object recognition, as well as algorithms for the rapid computation of scene-level gist and rough layout. When used on a robot, the toolkit additionally provides modules for image and video digitization, short- and long-term symbolic and spatial memories, decision processes, and control of actuators. The source code for the toolkit is freely available.[2]

Several neural models are available in the toolkit for the implementation of the next processing stage—concerned with identifying the object that has drawn the attention and the eyes—and most of these models are inspired by the visual-response properties of neurons in the infero-temporal cortex. Finally, additional modules are available for short-term and long-term memory, cognitive knowledge representation, and modulatory feedback from a high-level task definition (e.g., look for the stop sign) to the low-level visual processing (e.g., emphasize the contribution of red to the saliency map, prime the object recognition module for the 'traffic sign' object class).

Not all of the components shown in the figure have been fully implemented, and many are at a very preliminary stage of development: including some simply not yet existing. The interesting point to note already, however, is how the biologically-inspired visual system architecture proposed here is very different from typical robotic-vision and computer-vision systems, which are usually defined to solve a specific problem (e.g., find a stop sign by looking for its specific shape using an algorithm matched to its exact geometrical properties). This promises to make the systems developed around this architecture particularly capable when dealing with novel complex outdoors scenes and unexpected situations, as has been widely demonstrated by, for example, our model of bottom-up attention.[1]

The entire source code for the iLab Neuromorphic Vision C++ Toolkit is distributed freely upon request, in an effort to encourage more researchers to explore the potential of neuromorphic vision for real-world applications.

For more information about the iLab Neuromorphic Vision C++ Toolkit, see our web page.[2]

Laurent Itti
University of Southern California

References
1. http://iLab.usc.edu/bu/
2. http://iLab.usc.edu/toolkit/

If have a toolkit or other goodies to share, contact the Editor at ***sunny@sunnybains.com***