Media Processing Ecologies

V. Michael Bove, Jr., *Member, IEEE*

Abstract—**Research in the MIT Media Laboratory's Object-Based Media Group is looking to solve multimedia signal processing problems through the creation of ecosystems of intelligent devices that organize themselves to distribute information and to divide large tasks. I examine the technological trends that make this approach possible and desirable, and briefly introduce three example systems.**

Index Terms—**Multimedia signal processing, parallel multimedia processing, self-organizing systems, peer-to-peer**

I. Introduction

Despite the continuing acceleration of clock speeds and the continuing architectural refinements in microprocessors and digital signal processors – as well as the emergence of interesting computational alternatives such as field-programmable gate arrays – the processing of video and audio information continues to find ways to consume the available resources. That this is so shouldn't be a surprise, since the required degree of media processing is constantly being raised: users now expect the streaming video images on their PC screens to be television quality (and even what constitutes "television quality" is a moving target). Wireless phones become still cameras and then videophones. Video cameras become wireless IP video cameras, and have to do real-time compression, and then become "smart cameras" and must perform object tracking and recognition.

In search of more computational power (with a large scalability range) as well as robustness, software developers have in recent years developed peer-to-peer [1] and grid computing [2] solutions to demanding problems (some of the peer-to-peer developers were looking for other advantages such as anonymity, too, but that's outside the scope of this discussion). Both terms "grid" and "peer-to-peer" have been used in so many ways that the definitions of each and the distinctions between them are subject to discussion; the important point to note here is that parallel processing has existed for many years, but self-organizing parallel processing by dynamic groups of possibly heterogeneous devices is relatively new.

Since at least Haeckel's writings in the 1860's and 1870's, [3] it has been understood that *ecology* refers to a system of relationships, both among organisms and between them and their environments. Research at the MIT Media Laboratory and elsewhere is beginning to consider computational ecologies, or models for computation that are based upon relationships and self-organization. An area of personal interest to the author is the use of such systems for solving multimedia processing problems, particularly those in which the elements are not simply processors but also sensors and perhaps sensory output devices.

While biologically-inspired computational models are currently of great interest, it's important not to carry the analogy too far. For example, it is becoming apparent that some resources whose scarcity would impose limits on such systems (*e.g.* wireless bandwidth) do not have to deplete with system scaling in the (analogous to natural resources) manner previously assumed.[4] Thus there remain promising areas of inquiry in topics such as scalability and resource allocation.

In the following section I introduce three examples of computational ecologies specifically related to the processing of multimedia information.

I. Three Example Systems

A. Eye Society

Eye Society (Fig. 1) is a system of small wireless autonomous mobile cameras that operate as a group to solve machine vision tasks. Given sufficient on-board computing to do real-time scene analysis without external processing resources, these cameras enable us to investigate how scene understanding can be improved when each camera is independently capable of analyzing its own sensor data, and sharing information on what it sees with its fellow devices.[5]

There are many reasons for wanting to integrate the observations of multiple video cameras. Perhaps the simplest set of tasks involves surveillance, where it may be desirable to coordinate observations such that a moving person or vehicle is tracked continuously while passing in and out of the range of a group of cameras. Communications between two cameras might simply indicate a direction for a pan-tilt head to move to capture a region of interest, or the first camera to spot an object could send a machine-vision description of the region. Note that in each case, the inter-camera traffic involves "metadata"; given limited communications capacity and limited processing we can't simply multicast every camera's video to every other camera.

More complex applications for multiple-camera capture involve a greater degree of image understanding. Examples include scene capture for combinations of real and virtual environments,[6] modeling people,[7] or understanding their activities for the field of Human-Computer Interaction.[8] To

V. Michael Bove, Jr. is with the Massachusetts Institute of Technology Media Laboratory, Cambridge MA 02139 USA, e-mail vmb@media.mit.edu.
This research has been supported by the Digital Life Consortium, the NASA-Langley/ODU Center for Advanced Engineering Environments, and the Information and Communications University.

date, most of this work has involved fixed cameras and external processing. In the Eye Society project, we seek to solve these and other problems with a "society" of small, cheap, autonomous wireless mobile cameras each controlled by an embedded processor board running Linux. Each camera can independently pan and tilt and move along an overhead lighting track, and is in constant communication with its fellow cameras as it does so.

Each camera in the project has wireless communication to other cameras and access to "offshore" resources such as databases and file space as required. The current version of the controller centers on a commercial processor board containing a StrongARM SA-1110 processor running at 206Mhz, 32MB of flash memory, an SA-1111 companion chip (which allows interfacing to USB devices), and additional analog and digital inputs and outputs used to control locomotion and camera servos and to accept input from sensors such as microphones. An add-on board of our design contains motor drivers. Communication is through IEEE 802.11b.

Initial research seeks to solve solutions to such problems as calibration and confirmation of observations. The ability of the cameras to move allows them to seek locations that minimize undesirable effects such as occlusion and specular reflections, maximize visibility of useful features such as edges and three-point perspective, and verify hypotheses by such means as egomotion or stereopsis.

Longer-term work includes increasing the processor power of the cameras to allow more sophisticated real-time processing, recasting traditional machine-vision algorithms for a distributed environment, incorporating additional sensors such as microphones, and enabling other sorts of devices (both stationary sensing devices of the sort described in the following discussion and unconstrained-motion floor robots) to become part of the acquisition-and-understanding ecosystem.

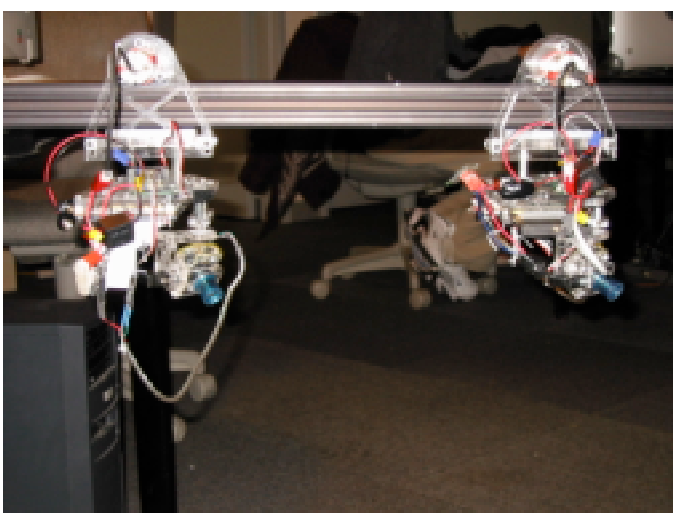

Fig. 1. Two Eye Society devices on an overhead track.

A. Smart Architectural Surfaces

The Smart Architectural Surfaces project, a collaboration between the Object-Based Media Group of the MIT Media Laboratory and the Information and Communications University in Seoul, Korea, examines the inclusion of sensory input/output and computation into building materials such that "smart rooms" and other intelligent input/output spaces can be built from modular elements. The communications and computational approaches are related to those of Eye Society, though with nonmobile devices that include more sensing modalities, and information output as well as input.

A single unit of a first-generation prototype system is illustrated in Fig. 2. This version can be thought of as a foot-square "brick" (though later versions will be more like tiles) from which walls can be built. Each unit contains similar processing and communication to the Eye Society cameras, along with input devices such as a camera, microphone, ultrasonic proximity sensor, temperature sensor, and humidity sensor. Output devices include a speaker, color-changing illumination, and either a display panel or a transparency-changing plastic sheet that can be used as a screen for an external projector. Other wireless devices such as Personal Digital Assistants can become members of the peer-to-peer application environment as needed.

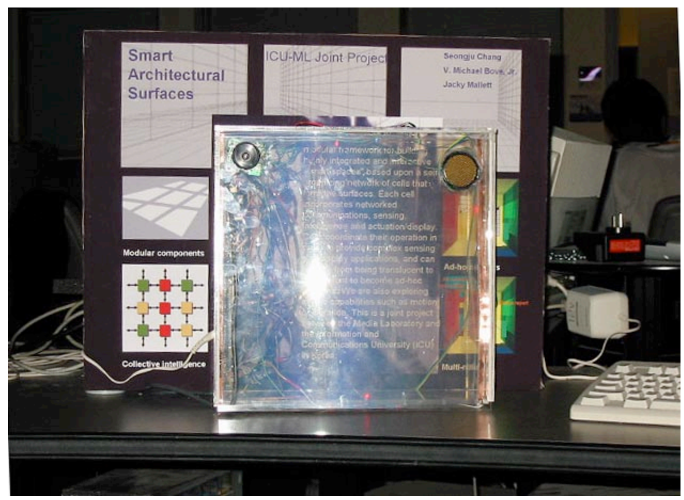

Fig. 2. One module from the Smart Architectural Surfaces project. Walls built from these units form the basis for smart rooms, linked meeting rooms, and other sensor- and output-equipped spaces.

A. Extremely-Distributed Media Processing

Conceptually related to the preceding project but operating at a much smaller scale and a much finer grain is work that was done by Butera under the name of "paintable" computing.[9],[10] The idea here was to deploy ultra-miniature processing nodes by the thousands in bulk fashion, laminating them into building materials or even mixing them with a coating and painting them onto surfaces. While on the whole such a system offers seemingly unlimited computational power for multimedia processing purposes, the individual devices contain modest amounts of memory, have wireless communication ranges on the order of millimeters to centimeters (meaning that each can likely communicate directly with only ten to twenty other devices), lack explicit addressing, and might not be very reliable.

Software on a such a system is organized into autonomous, migratory self-contained executables called as "process fragments." All data is communicated either as the payload to a process fragment or by means of a mirroring operation that makes a portion of each processing particle's memory visible to each other particle within its communication radius. When a process fragment arrives at a particle with free memory space enough to hold it, it executes its code portion, looks for relevant data in the mirrored memory area, and perhaps posts some other data there. A process fragment can request a transfer out of a particle after executing.

Even though any device has knowledge of only the other devices in its immediate neighborhood, it is possible to communicate with particular devices at a distance by means of a process fragment that generates a gradient field tagged with a label identifying the source (Fig. 3). Essentially, it does this by sending copies of itself to "uninfected" particles, incrementing an integer number of hops back to the source; next each copy estimates a floating-point distance based upon the integer hop counts of all its neighbors (which are visible in the mirrored memory area). It is then possible for process fragments in remote locations to "ride the gradient back" to communicate with the source.

Multimedia-related applications that have been investigated in this environment include distributed audio storage and playback, image segmentation, and searchable image libraries. These are discussed in the above-cited references.

The original work on this project was done in a process simulator, but recent hardware designs have been inspired by the results.[11]

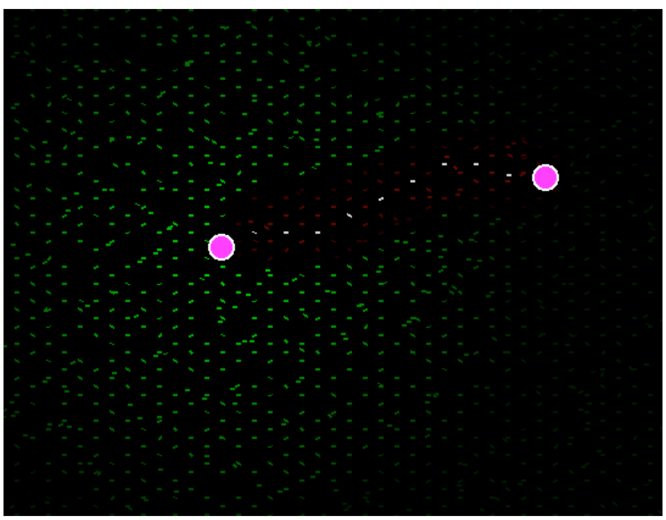

Fig. 3. Snapshot from the Paintable Computing simulator, in which one processing particle (left) radiates code that computes its distance from the source. A second particle (right) can construct a pathway to the first by routing messages through particles that follow the gradient back to the source.

Acknowledgment

The author wishes to thank Jacky Mallett for her work on the Eye Society and Smart Architectural Surfaces projects; on the latter project thanks also to Seongju Chang, Jae-Do Chung, and Brian Mullins. The extremely-distributed media processing work appears thanks to Bill Butera and Jim McBride.

References

[1] J. Kubiatowicz, "Extracting guarantees from chaos," *Communications of the ACM,* vol. 46, no. 2, pp. 33-38, Feb. 2003.

[2] H. Casanova, "Distributed computing research issues in grid computing," *ACM SIGACT News,* vol. 33, no. 3, pp. 50-70, Sept. 2002.

[3] E. Haeckel, *Generelle Morphologie der Organismen,* G. Reimer, Berlin, 1866.

[4] M. Gastpar and M. Vetterli, "On the capacity of wireless networks: the relay case," *Proc. INFOCOM 2002*, vol. 3, pp. 1577-1586, 2002.

[5] J. Mallett and V. M. Bove, Jr., "Eye Society," *Proc. IEEE ICME 2003,* in press.

[6] S. Moezzi, L.-C. Tai, and P. Gerard, "Virtual View Generation for 3D Digital Video" *IEEE MultiMedia*, vol. 4, no. 1, Jan.-Mar. 1997.

[7] D. Beresford, *et al.,* "Whole-body modeling of people from multiview images to populate virtual worlds," *The Visual Computer,* vol. 16, no. 7, pp. 411-436, 2000.

[8] M. H. Cone and K. W. Wilson, "Learning spatial event models from multiple-camera perspectives," *IEEE IECON '99 Proceedings*, vol. 1, pp. 149 –156, 1999.

[9] W. Butera, "Programming a paintable computer," PhD Thesis, Massachusetts Institute of Technology, Cambridge MA, 2001.

[10] W. Butera, V. M. Bove, Jr. and J. McBride, "Extremely distributed media processing," *Proc. SPIE Media Processors* vol. 4674, pp. 37-41, 2002.

[11] J. Lifton *et al.,* "Pushpin Computing System overview: a platform for distributed, embedded, ubiquitous sensor networks," in F. Mattern and M. Naghshineh (eds): *Pervasive 2002, Proceedings of the Pervasive Computing Conference*, Springer-Verlag, Berlin, pp. 139-151, 2002.