Networked Robotic Cameras for Collaborative Observation of Natural Environments

Dezhen Song[1] and Ken Goldberg[2]

1: CS Department, Texas A&M University,
2: IEOR and EECS Departments, UC Berkeley

Abstract

Many viewers simultaneously observe events in a common, but remote, environment in applications ranging from scientific observation, journalism, distance education, security, and inspection to entertainment. Such "collaborative observation" can be achieved with an emerging class of networked robotic cameras. A primary challenge is to resolve contention for control of camera motion. In this paper we describe a series of prototype systems and algorithms we're developing.

I. Introduction

Scientific study of animals in situ requires vigilant observation of detailed animal behavior over weeks or months. When animals live in remote and/or inhospitable locations, observation can be an arduous, expensive, dangerous, and lonely experience for scientists. Emerging advances in robot cameras, long-range wireless networking, and distributed sensors make feasible a new class of portable robotic "observatories" that can allow groups of scientists, via the internet, to remotely observe, record, and index detailed animal activity. As a shorthand for such an instrument, we propose the acronym *CONE: Collaborative Observatory for Natural Environments.*

One challenge is to develop a mathematical framework for collaborative observation. Collaborative observation includes (1) collaboration between humans of different backgrounds, skill sets, and authority/permission levels and (2) collaboration between humans and automated agents whose behavior arises from sensor inputs and/or computation. As illustrated in Figure 4, our framework uses a *panoramic image* and set of *activity frames* to provide a unified representation for output and for input from both human observers and sensors.

II. Related Work

Since Nikola Tesla demonstrated the first radio-controlled boat in 1898 and Goertz demonstrated a bilateral manipulator in 1954 [7], remotely operated machines have been widely desired for use in inhospitable environments such as radiation sites, undersea [1] and space exploration [3], [24], [33]. Today, teleoperation is being developed for medical diagnosis [2], manufacturing [6] and micromanipulation [27]. See Sheridan [28] for an excellent review of the extensive literature on teleoperation and telerobotics. Most of these systems require fairly complex hardware at the human interface: exoskeleton master linkages are operated by highly trained specialists. In contrast, the Internet can provide public access by using only the interface available in standard browsers.

The hypertext transfer protocol developed at CERN in 1992 [4], provides a low-cost and publicly available network interface. In the Spring of 1994, we conjectured that we could use it to offer public access to a teleoperated robot via the Internet.

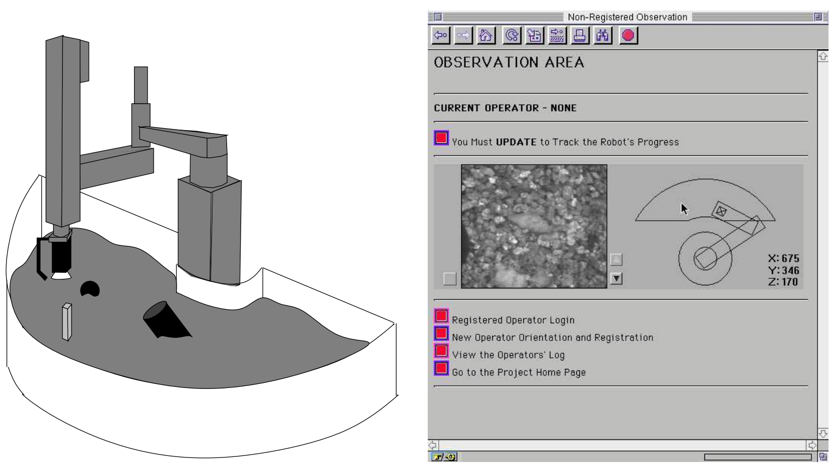


Fig. 1. Mercury Project (1994-1995). Above: Robot, camera and air nozzle above sandbox filled with buried artifacts. Below: Browser Interface using vanilla HTTP 1.0.

As illustrated in Figure 1, we set up an IBM SCARA robot arm over a semi-annular workspace containing sand and buried artifacts. We attached a CCD camera to the end of the arm along with a nozzle to direct air bursts into the sand. We then developed a HTTP 1.0 (Mosaic) browser interface to the hardware. The Mercury Project was operated by over 10,000 people and is widely regarded as the first Internet robot [11], [10].

Our subsequent project, the Telegarden, allowed users to view and interact with a remote garden filled with living plants. We incorporated a much faster Adept-1 industrial robot arm and allowed the robot to be multi-tasked to eliminate the user queue. The Telegarden was installed at a museum in Austria where it operated around the clock for nine years was operated by over 100,000 people online.

Fig. 2. The Tele-Garden (1995-2004). (with Joseph Santarromana, George Bekey, Steven Gentner, Rosemary Morris Carl Sutter, Jeff Wiegley, Erich Berger, and Thomas Steindl).

In 1994, working independently, a team led by K. Taylor and J. Trevelyan at the University of Western Australia demonstrated a remotely controlled six-axis telerobot in September 1994 [5], [17]. There are now dozens of Internet robots online, a book from MIT Press [12], and an IEEE Technical Committee on Networked Robots that has over 200 members. See [18], [26], [20], [19], [21], [23], [15], [25], [22] examples of recent projects.

III. The Tele-Actor and ShareCam

In 1999 we began exploring other models of access control, where user inputs are combined rather than sequenced. In [9], [8], we describe an Internet-based Multiple Operator Single Robot system that use vector averaging to combine multiple mouse inputs to simultaneously control a single industrial robot arm. In [13], [14], we describe a Java-based "Spatial Dynamic Voting" (SDV) interface that collects, displays, and analyzes a sequence of spatial votes from multiple online operators at their Internet browsers. The votes can drive the motion of a single mobile robot or, for increased mobility and agility, a human "Tele-Actor".

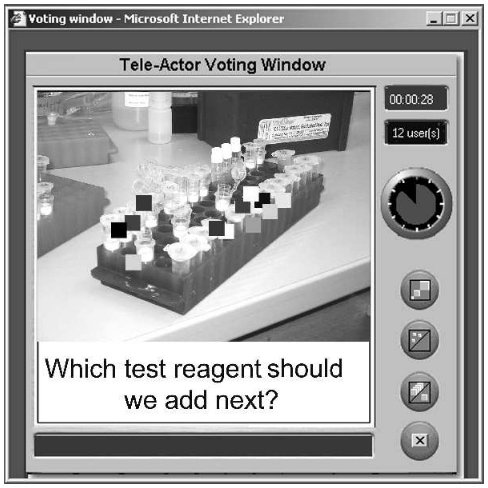


Fig. 3. Spatial Dynamic Voting Interface and the Tele-Actor (2001-2004).

IV. THE COLLABORATIVE FRAME SELECTION PROBLEM

We are now developing systems based on robotic pan, tilt, zoom cameras controllable by many simultaneous viewers over the Internet. Since there is one camera and many viewers, the challenge is to resolve contention about where to point the camera.

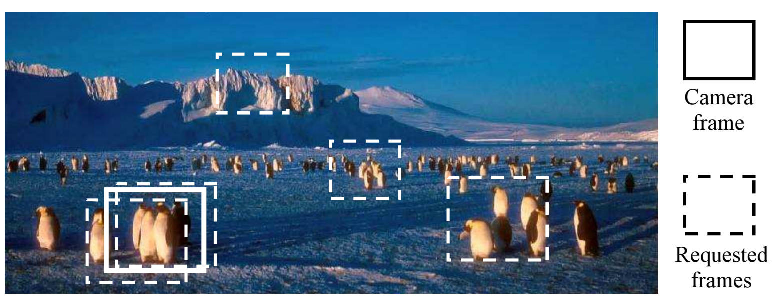


Fig. 4. Panoramic image and user or machine-requested “activity frames”.

Collaborative observation includes (1) collaboration between humans of different backgrounds, skill sets, and authority/permission levels and (2) collaboration between humans and automated agents whose behavior arises from sensor inputs and/or computation. We propose using a *panoramic image* and set of *activity frames* to provide a unified representation for output and for input from both human observers and sensors.

On the output (display) side, the wide-field panoramic image provides a relative spatial context for close-up camera views.

On the input side, each activity frame is a rectangular region with the aspect ratio of the camera. As illustrated in Figure 4, human users specify activity frames of interest by drawing them with standard mouse over the panoramic image; the boundaries of the frame intuitively match each desired camera view. Below we review algorithms we've developed that efficiently process a set of activity frames to compute optimal frames for the camera.

Each activity frame is a rectangular region with the aspect ratio of the camera. As illustrated in Figure 4, human users specify activity frames of interest by drawing them with standard mouse over the panoramic image; the boundaries of the frame intuitively match each desired camera view.

Let $c = [x, y, z]$ define a rectangular camera frame (the camera has a fixed aspect ratio of 4:3). User i requests a desired frame r_i. Given requests from n users, the system must compute a single global frame c^* that will best satisfy the set of requests. Clearly simple averaging will work poorly as it can produce centered camera frames that satisfy none of the users.

We define the "Coverage-Resolution Ratio (CRR)" as a reward, or "satisfaction" metric $s(c, r_i)$ based on how closely the requested frame r_i compares with a candidate camera frame c. One sample CRR metric is described below,

$$s_i(c) = \frac{Area(r_i \cap c)}{Area(r_i)} \min\bigl(\frac{z_i}{z}, 1\bigr). \tag{1}$$

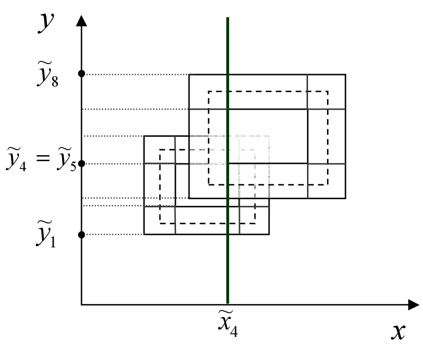
Equation 1 characterizes the intuition that satisfaction has to be an increasing function of coverage ratio $\frac{Area(r_i \cap c)}{Area(r_i)}$. In our definition, larger z or z_i means larger in frame size but lower in resolution. Therefore, an extremely large camera frame can increase coverage ratio but will decrease the resolution ratio $\frac{z_i}{z}$.

Each of n users submits a request. In the collaborative camera control, we want to find c^*, the value of c that maximizes overall satisfaction based only on the current set of requests:

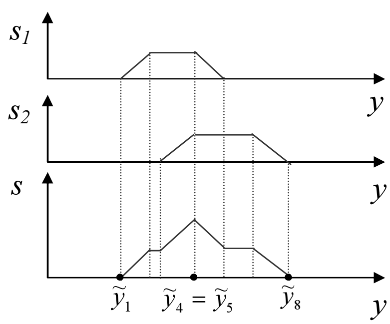
$$\max_c \sum_{i=1}^{n} s_i(r_i, c) = \sum_{i=1}^{n} \frac{Area(r_i \cap c)}{Area(r_i)} \min\bigl(\frac{z_i}{z}, 1\bigr). \tag{2}$$

In each motion cycle, we servo the camera to the computed position and zoom level.

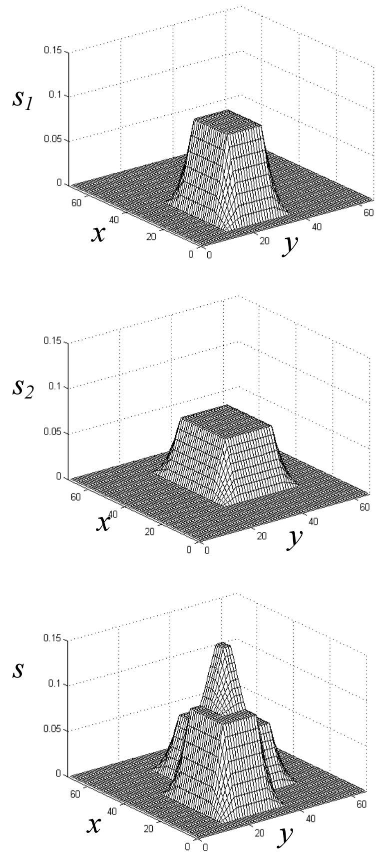
Since the reward metric is non-concave and non-differentiable, efficiently computing the optimal solution for Equation 2 is non-trivial as



(a) Top view of s_1 and s_2



(b) Piecewise linear objective function at $x = \tilde{x}_4$



(c) 3D view of objective functions

Fig. 5. *Shape of reward metric for a fixed camera zoom level. For each user, their specified activity frame gives rise to an objective function that is plateau-like as illustrated in (c). The function consists of 5 planar and 4 quadratic surfaces at the corners. The overall objective function is the summation of plateaus generated by activity frames from all users.*

illustrated in Figure 5. In [31], we show that the shape of the objective function for a single user has a plateau-like shape. To efficiently compute the summation of a set of plateaus, we developed an $O(mn^2)$ exact algorithm based on idea of sweeping and incremental computation. Since the camera may have a continuously variable zoom and user requests are not necessarily rectangular, we have developed a series of algorithms as summarized in Table I.

Activity frames can also provide a natural representation for input from

No.	Type	Zoom	Request	Solution	Complexity	Pub.
1	Centralized	m levels	Rectangle	Exact	$O(mn^2)$	[31]
2	Centralized	m levels	Rectangle	Approximation	$O(mn^{\frac{3}{2}} \log n)$	[16]
3	Distributed	m levels	Rectangle	Exact	Server: $O(mn)$ Client: $O(n)$	[32]
4	Centralized	Continuous	Rectangle	Exact	$O(n^3)$	[30]
5	Centralized	Continuous	Polygon	Approximation	$O((n + 1/\epsilon^3) \log^2 n)$	[30]
6	Distributed	Continuous	Polygon	Approximation	Sever: $O(n)$ Client $O(1/\epsilon^3)$	[29]

TABLE I

Algorithms developed for Collaborative Frame Selection, where n is number of activity frames specified, and m is the number of camera zoom levels.

sensors. For example, pyroelectric motion sensors respond to activity within a convex spatial region that can be projected onto the image plane and conservatively bounded by a rectangular activity frame. The same is true for optical beam sensors, pressure pads, and directional microphones.

For example, consider a set of commercial pyroelectric motion sensors configured to detect animal motion (eg. motion of warm bodies > 50 lbs). Each sensor has an associated field of view and responds with different quantitative levels based on mass and velocity of movement. When several sensors go off simultaneously, a series of camera positions may be selected as proposed above. It is also important not to "starve" any sensor that may indicate a crucial observation. Similar "starve" effect can also happen to a minority user, whose observing interests may be different from the majority.

We can augment the frame selection model in Equation 2 by introducing time variable t and, for each sensor, a linear gain function ω_i. The gain is a function of camera motion history, sensor reliability, and scientific observation priorities.

$$\max_{c(t)} \sum_{i=1}^{n} \omega_i(t) s_i(r_i(t), c(t)) = \sum_{i=1}^{n} \omega_i(t) \frac{Area(r_i(t) \cap c(t))}{Area(r_i(t))} \min\left(\frac{z_i(t)}{z(t)}, 1\right). \tag{3}$$

We propose a gain function based on camera history as follows. We define a "dissatisfaction" value for each user (in this case each sensor) based on how poorly the last camera frame was aligned with the sensor's last activity frame request: $u_i(t) = 1 - s_i(r_i(t), c(t))$. This "dissatisfaction" gain can accumulate over time: $\omega_i(t) = \sum_{k=0}^{t-1} \frac{u_i(k)}{2^{t-1-k}}$, so that when other sensors are satisfied with consistent camera motion, the neglected sensor gradually gains in influence. This can be defined in a recursive format,

$$\omega_i(t) = u_i(t-1) + \omega_i(t-1)/2.$$

Effectively, the weight of the un-observed region will increase until it is observed. Preliminary experiments suggest that this approach is robust, insuring that all sensors contribute and preventing the system from having observation driven by only a small number of dominating sensors (or users!).

V. Conclusion and Future Work

This paper reviews a series of prototype networked robotic systems and associated algorithms for collaborative observation.

We are currently extending our framework to consider resource limited observation, heterogenous user groups, optimizing camera trajectory, temporal modeling, sensor modeling, sensor monitoring and fault detection, and robotic actuation. We will develop automated agents based on sensors, robotic calibration for rapid deployment, and a video database for archiving, indexing, and query of observed scientific data.

Acknowledgments

This work was funded in part by National Science Foundation IIS-0113147, by Intel, Adept, Panasonic, Qualcomm, and Microsoft Corporations, and UC Berkeley's Center for Information Technology Research in the Interest of Society (CITRIS). Thanks to Dana Plautz, Frank van der Stappen, Steven Gentner, Carl Sutter, Jeff Wiegley, Michael Mascha, and Nick Rothenberg. Joseph Santarromana, George Bekey, Steven Gentner, Rosemary Morris Carl Sutter, Jeff Wiegley, Erich Berger, Thomas Steindl, Howard Moraff, John Canny, Eric Paulos, Judith Donath, Bob Farzin, Eric Mankin, Peter Lunenfeld, Roland Siegwart, Ruzena Bajcsy, Paul Wright, and Anthony Levandowski.

References

[1] R. D. Ballard. A last long look at titanic. *National Geographic*, 170(6), December 1986.

[2] A. Bejczy, G. Bekey, R. Taylor, and S. Rovetta. A research methodology for telesurgery with time delays. In *First International Symposium on Medical Robotics and Computer Assisted Surgery*, Sept 1994.

[3] A. K. Bejczy. Sensors, controls, and man-machine interface for advanced teleoperation. *Science*, 208(4450), 1980.

[4] Tim Berners-Lee, Robert Cailliau, Jean-Francios Groff, and Bernd Pollerman. World-wide web: The information universe. *Electronic Networking:Research, Applications and Policy*, 1(2), Westport CT, Spring 1992.

[5] Barney Dalton and Ken Taylor. A framework for internet robotics. In *IEEE International Conference On Intelligent Robots and Systems (IROS): Workshop on Web Robots*, Victoria, Canada, 1998.

[6] Matthew Gertz, David Stewart, and Pradeep Khosla. A human-machine interface for distributed virtual laboratories. *IEEE Robotics and Automation Magazine*, December 1994.

[7] Raymond Goertz and R. Thompson. Electronically controlled manipulator. *Nucleonics*, 1954.
[8] K. Goldberg and B. Chen. Collaborative control of robot motion: Robustness to error. In *International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 655–660, October 2001.
[9] K. Goldberg, B. Chen, R. Solomon, S. Bui, B. Farzin, J. Heitler, D. Poon, and G. Smith. Collaborative teleoperation via the internet. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 2019–2024, April 2000.
[10] K. Goldberg, M. Mascha, S. Gentner, N. Rothenberg, C. Sutter, and Jeff Wiegley. Beyond the web: Manipulating the physical world via the www. *Computer Networks and ISDN Systems Journal*, 28(1), December 1995. Archives can be viewed at http://www.usc.edu/dept/raiders/.
[11] K. Goldberg, M. Mascha, S. Gentner, N. Rothenberg, C. Sutter, and Jeff Wiegley. Robot teleoperation via www. In *International Conference on Robotics and Automation*. IEEE, May 1995.
[12] K. Goldberg and R. Siegwart, editors. *Beyond Webcams: An Introduction to Online Robots*. MIT Press, 2002.
[13] K. Goldberg, D. Song, Y. Khor, D. Pescovitz, A. Levandowski, J. Himmelstein, J. Shih, A. Ho, E. Paulos, and J. Donath. Collaborative online teleoperation with spatial dynamic voting and a human "tele-actor". In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1179–84, May 2002.
[14] K. Goldberg, D. Song, and A. Levandowski. Collaborative teleoperation using networked spatial dynamic voting. *The Proceedings of The IEEE*, 91(3):430–439, March 2003.
[15] K. Han, Y. Kim, J. Kim, and S.Hsia. Internet control of personal robot between kaist and uc davis. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2002.
[16] Sariel Har-Peled, Vladlen Koltun, Dezhen Song, and Ken Goldberg. Efficient algorithms for shared camera control. In *19th ACM Symposium on Computational Geometry, San Diego, CA*, June 2003.
[17] **http://telerobot.mech.uwa.edu.au/**.
[18] H. Hu, L. Yu, P. W. Tsui, and Q. Zhou. Internet-based robotic systems for teleoperation. *Assemby Automation*, 21(2):143–151, May 2001.
[19] S. Jia, Y. Hada, G. Ye, and K. Takase. Distributed telecare robotic systems using corba as a communication architecture. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2002.
[20] S. Jia and K. Takase. A corba-based internet robotic system. *Advanced Robotics*, 15(6):663–673, Oct 2001.
[21] J. Kim, B. Choi, S. Park, K.Kim, and S. Ko. Remote control system using real-time mpeg-4 streaming technology for mobile robot. In *IEEE International Conference on Consumer Electronics*, 2002.
[22] R.C. Luo and T. M. Chen. Development of a multibehavior-based mobile robot for remote supervisory control through the internet. *IEEE/ASME Transactions on Mechatronics*, 5(4):376–385, 2000.
[23] T. Mirfakhrai and S. Payandeh. A delay prediction approach for teleoperation over the internet. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2002.
[24] R. S. Mosher. Industrial manipulators. *Scientific American*, 211(4), 1964.
[25] L. Ngai, W.S. Newman, and V. Liberatore. An experiment in internet-based, human-assisted robotics. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2002.

[26] R. Safaric, M. Debevc, R. Parkin, and S. Uran. Telerobotics experiments via internet. *IEEE Transactions on Industrial Electronics*, 48(2):424–31, April 2001.

[27] T. Sato, J. Ichikawa, M. Mitsuishi, and Y. Hatamura. A new micro-teleoperation system employing a hand-held force feedback pencil. In *International Conference on Robotics and Automation*. IEEE, May 1994.

[28] Thomas B. Sheridan. *Telerobotics, Automation, and Human Supervisory Control*. MIT Press, 1992.

[29] D. Song. *Systems and Algorithms for Collaborative Teleoperation*. PhD thesis, Department of Industrial Engineering and Operations Research, University of California, Berkeley, 2004.

[30] D. Song, A. Pashkevich, and K. Goldberg. Sharecam part II: Approximate and distributed algorithms for a collaboratively controlled robotic webcam. In *IEEE/RSJ International Conference on Intelligent Robots (IROS)*, Nov. 2003.

[31] D. Song, A. F. van der Stappen, and K. Goldberg. Exact and distributed algorithms for collaborative camera control. In *The Workshop on Algorithmic Foundations of Robotics*, Dec. 2002.

[32] D. Song, A.F. van der Stappen, and K. Goldberg. An exact algorithm optimizing coverage-resolution for automated satellite frame selection. In *IEEE International International Conference on Robotics and Automation (ICRA 2004), New Orleans, US*, 5 2004.

[33] R. Tomovic. On man-machine control. *Automatica*, 5, 1969.