Collaborative Online Teleoperation with Spatial Dynamic Voting and a Human “Tele-Actor” [*]

K. Goldberg, D. Song, Y. Khor, D. Pescovitz, A. Levandowski,
J. Himmelstein, J. Shih, A. Ho, E. Paulos, **UC Berkeley**
J. Donath, **MIT**

Internet-based “online robots” now provide public access to remote locations such as museums and laboratories. The Tele-Actor is a collaborative online teleoperation system for distance learning that allows many students to simultaneously share control of a single mobile resource. Our goal is to preserve the educational advantages of field trips without the drawbacks of group travel.

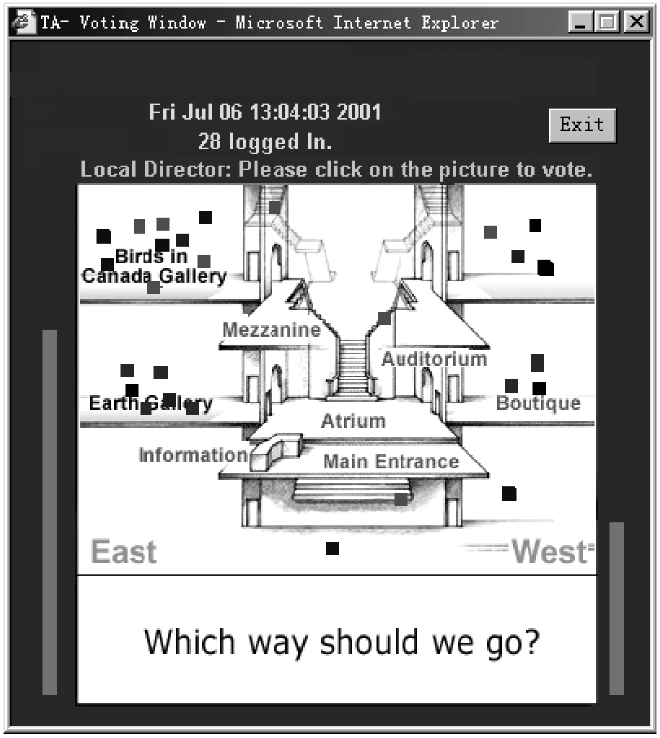
We propose the “Spatial Dynamic Voting” (SDV) interface for Multiple Operator Single Robot (MOSR) teleoperation. The SDV collects, displays, and analyzes a sequence of spatial votes from multiple online operators at their Internet browsers. The votes drive the motion of a single mobile robot or human “Tele-Actor”. This paper describes Version 3.0 of the system architecture, SDV interface, algorithms for automated goal selection, and metrics for collaboration and leadership. We report results from a July 2001 field test with 56 remote users. See: www.tele-actor.net.

1 Introduction

Consider the following scenario: an instructor wants to bring a class of high-school students to visit a working robotics lab or microelectronics fabrication facility. Due to safety, security, and liability issues, it may not be practical to arrange a class visit. Showing a prestored video does not offer the excitement and group dynamics of the field trip. In this paper we propose a system that can preserve the advantages of a field trip without the drawbacks of group travel.

The Internet provides a low-cost and widely-available interface that can make physical resources accessible to a broad range of participants. There are now dozens of “online robots”, a book from MIT Press [20], and an IEEE Technical Committee on Online Robots.

In most existing online robot systems, as in conventional telerobotics, one process (human or computer) controls a single robot. In the taxonomy proposed by Tanie et al. [9], these are Single Operator Single Robot (SOSR) teleoperation systems.



TA- Voting Window - Microsoft Internet Explorer

Fri Jul 06 13:04:03 2001

28 logged In.

Local Director: Please click on the picture to vote.

Which way should we go?

Figure 1: *Spatial Dynamic Voting (SDV) interface. Internet-based voters position colored square markers in the “election image” to indicate their motion preferences.*

Multiple Operator Single Robot (MOSR) systems are motivated by applications in education and journalism, where groups of users desire simultaneous access to a single robotic resource such as a camera. Inputs from all users must be combined to generate a single control stream for the robot. There are benefits to collaboration: teamwork is a key element in education at all levels [36, 11, 10]; also a group of users may be more reliable than a single (possibly malicious) user.

In this paper we describe a MOSR teleoperation system that allows a group of online users to view and vote on the movements of a single mobile camera. As illustrated in Figure 1, voters use their Internet browsers to collaboratively direct the remote camera using the “Spatial Dynamic Voting” (SDV) interface. The SDV provides a sequence of still

[*] IEEE International Conference on Robotics and Automation, May 2002, Washington, D.C. This work was supported in part by the National Science Foundation under IIS-0113147. For more information please contact goldberg@ieor.berkeley.edu.

"election images". User votes are represented by "votels": small colored squares that are positioned by users with a mouse click. In this example, the election image is a map. The majority of online voters want to move to the second floor gallery while others are split between moving to the boutique or gallery on the first floor.

The remote camera can be carried by a mobile robot or a human "Tele-Actor". We refer to our early prototypes of this system as version 1.0 and 2.0. In this paper we describe version the 3.0 system architecture, user interface, and SDV algorithms for automated goal selection. We report results from a live field test with a human Tele-Actor at the San Francisco Opera House conducted on July 18, 2001 with 56 remote online participants.

2 Related Work

Online robots, controllable over the Internet, are an active research area. In addition to the challenges associated with time delay, supervisory control, and stability, online robots must be designed to be operated by non-specialists through intuitive user interfaces and to be accessible 24 hours a day. See [21, 37, 23] for examples of recent projects.

Tanie, Matsuhira, Chong, et al. [9] proposed a very useful taxonomy for teleoperation systems: Single Operator Single Robot (SOSR), Single Operator Multiple Robot (SOMR), and Multiple Operator Multiple Robot (MOMR). The Tele-Actor system we consider here is Multiple Operator Single Robot (MOSR).

Most online robots are SOSR, where control is limited to one operator at a time. Tanie et al. analyzed an MOMR system where each operator controls one robot arm and the robot arms have overlapping workspaces. They show that predictive displays and scaled rate control are effective in reducing pick-and-place task completion times that require cooperation from multiple arms [9].

In an MOMR project by Fukuda, Liu, Xi, and colleagues [13], two remote human operators collaborate to achieve a shared goal such as maintaining a given force on an object held at one end by a mobile robot and by a multi-jointed robot at the other. The operators, distant from the robots and from each other, each control a different robot via force-feedback devices connected to the Internet. The authors show both theoretically and experimentally that event-based control allows the system to maintain stable synchronization between operators despite variable time-lag on the Internet.

MOMR models are also relevant to online collaborative games such as *Quake*, where players remotely control individual avatars in a shared environment.

In SOMR systems, one tele-operator or process controls multiple robots. This bears some relation to Cooperative (behavior-based) robots, where groups of autonomous robots interact to solve an objective [2]. Recent results are reported in [12, 35, 32, 7].

A number of SOSR systems have been designed to facilitate remote interaction. Paulos and Canny's Personal Roving Presence (ProP) telerobots, built on blimp or wheeled platforms, were designed to facilitate remote social interaction with a single remote operator [29, 30]. Fong, Thorpe and colleagues study SOSR systems where collaboration occurs between a single operator and a mobile robot that is treated as a peer to the human and modeled as a noisy information source [15]. Related examples of SOSR "cobots" are analyzed in [1, 26, 38, 4].

One precedent for an online MOSR system is described in McDonald, Cannon and colleagues [27]. For waste cleanup, several users to assist remotely using Point-and-Direct (PAD) commands [8]. Users point to cleanup locations in a shared image and a robot excavates each location in turn. In this Internet-based MOSR system, collaboration is serial but pipelined, with overlapping plan and execution phases. The authors demonstrate that such collaboration improves overall execution time but do not address conflict resolution between users.

Pirjanian studies how reliable robot behavior can be produced from an ensemble of independent processors [31]. Drawing on research in fault-tolerant software [25], Pirjanian considers systems with a number of homogenous processors sharing a common objective. He considers a variety of voting schemes and shows that fault-tolerant *behavior fusion* can be optimized using plurality voting [5] but does not consider spatial voting models such as ours.

In [17] we described an Internet-based MOSR system that averaged multiple human inputs to simultaneously control a single industrial robot arm. We reported experiments with maze-following that suggested that groups of humans perform better than individuals in the presence of noise due to central limit effects.

In [18], we used finite automata to model collaborating users in a MOSR system such as Cinematrix, a commercial audience-interaction system [33]. The ensemble of inputs is averaged to compute a single stream of incremental steps to control the motion of a point robot moving in the plane. We analyze system performance with a uniform ensemble of well-behaved deterministic sources and then modeled malfunctioning sources that go silent or generate inverted control signals. We discovered that performance can *improve* in the presence of malfunctioning sources and continue to function even when a sizeable fraction of sources malfunction.

Outside of robotics, the notion of MOSR is related to a very broad range of group activities including social psychology, voting, economics, market pricing, traffic flows, etc. ACM organizes annual conferences on Computer Supported Collaborative Learning (CSCL) and Computer Supported Cooperative Work (CSCW). Surveys of research in this broader context can be found in [28, 24, 34, 16, 3, 14, 19].

3 System Architecture

The Tele-Actor system architecture is illustrated in Figure 2. Users share control of the Tele-Actor from their Internet browsers. Communication is coordinated with the Tele-Actor Server located on a high bandwidth (T3) connection. The Server connects via a socket to the Local Basestation at the remote site. The Local Basestation and Local (human) Director communicate with the Tele-Actor via a wireless interface. As the Tele-Actor moves through the environment, video images are captured at the Local Basestation and streamed back to the Tele-Actor server for distribution as .jpg images to the remote voters. Voter responses are collected at the Tele-Actor server, displayed for all active voters and for the Local Director, who relays election results to the Tele-Actor via a wireless audio channel.

4 SDV User Interface

The “Spatial Dynamic Voting” interface facilitates interaction and collaboration among remote users. Figure 1 illustrates the SDV interface displayed at the browser of all active voters. Users register online to participate by selecting a votel color and submitting their email address to the Tele-Actor server, which stores this information in our database and immediately sends back a password via email. The server also maintains a tutorial and an FAQ section to familiarize new users with how the systems works.

Using the SDV interface, voters participate in a series of short (eg: 1 minute) “elections”. Each election is based on a single image with a textual question. In the example from Figure 1, the Tele-Actor is visiting an architectural site. The election image shows a building with the question: ”Where should we go next?” Voters click on their screens to position their votels. Using the HTTP protocol, these positions are sent back to the Tele-Actor server and appear in an updated election image sent to all voters every 6-20 seconds. In this way voters can change their votes several times during an election. When the election is completed, SDV analysis algorithms based on clustering can analyze the voting pattern to determine a single command for the remote mobile robot.

The SDV interface differs from multiple choice polling because it allows spatially and temporally continuous inputs. To facilitate user training and asynchronous testing, the Tele-Actor system is available in two modes. In Offline mode, all election images are from a prestored library. In online mode, election images are sampled from live video captured by the Tele-Actor. Both offline and online SDV modes have potential for collaborative education, testing, and training. In this paper we focus on the online mode.

5 Hardware and Software

The Tele-Actor webserver is an AMD K7 950Mhz PC with 1.2GB memory connected to a 100Mbs T3 line. The Local Base Station is a Dell Pentium III 600Mhz laptop with 64MB memory connected to a 10Mbs T1 line at the remote site. It has a USB video card, which captures video at 320×240 resolution.

We used the Swann MicroCam wireless video camera, model ALM-2452 [1]. It is 18x34x20 mm and weighs 20 grams, with a 9 volt battery as its power supply. It has a 2.4 GHz analog RF output at 10 mW and transmits line-of-sight up to 300 feet with a resolution of 380 horizontal lines.

Custom software includes: (1) the client side SDV browser interface based on DHTML, (2) the Local Basestation image selection interface, and (3) the Tele-Actor server. During online mode, the Local Basestation, running Microsoft Windows 98, uses a custom C++ application to capture images with textual questions and transmit them to the Tele-Actor server for distribution.

During both online and offline modes, the Tele-Actor server uses custom C and C++ applications to maintain the database and communicate with the LBS and with all active voters. The Tele-Actor server runs Redhat Linux 7.1 and the Apache web server 1.3.20. The Resin 2.0.1 Apache plug-in and Sun JDK 1.3.1 with Mysql database 3.23.36 provide java server pages to handle the user registration and data logging [2]. Custom software built on the graphic development toolkit GD 2.0.1 generates election images overlaid with current votel positions.

6 Online Field Test

We performed a one-hour field test at the 5th Annual Webby Awards on Wednesday, July 18, 2001 at San Francisco’s Opera House, from 10.30-11.30 pm Pacific Standard Time. Over 3000 people attended a reception after the ceremony and many more watched via live webcast.

From 2-16 July 2001, via email and newsgroup postings, we announced the Tele-Actor website was available for registration and testing. During this period users gained familiarity with the SDV interface using a library of 76 prestored election images. Two hundred voters registered during this period. The 18 July 2001 field test included 60 election cycles. Fifty-six voters participated, with a peak of 37 voters in any one election cycle.

[1] http://www.swann.com.au

[2] http://www.caucho.com

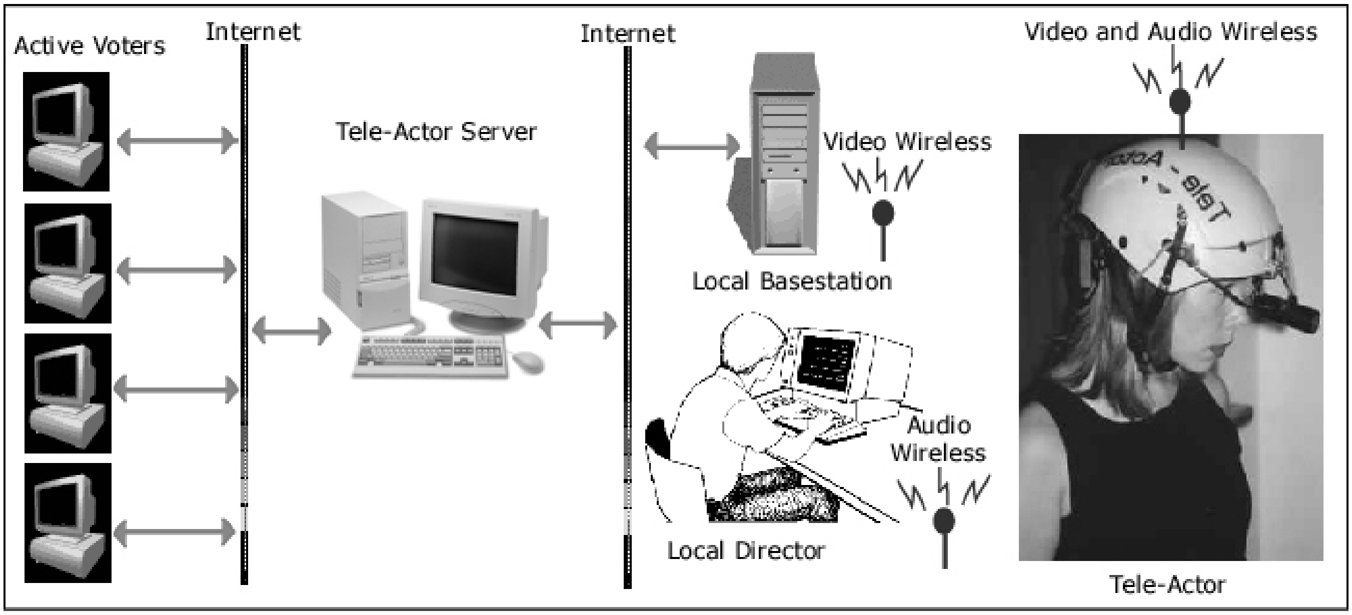


Figure 2: *Tele-Actor system architecture. Users on the Internet participate by voting on a series of election images. The human "Tele-Actor," with head-mounted wireless audio/video link, moves through the remote environment.*

During the field test, the Tele-Actor carried the wireless video camera through the lobby of the Opera House. The wireless camera transmitted live video images locally to a student crew at the Local Basestation, who selected frames and textual questions which were uploaded to the server at UC Berkeley and then sent out to all voters. All votes and election data were logged for subsequent analysis.

7 SDV Metrics and Algorithms

In this section we propose algorithms and metrics for analyzing voting patterns in the Spatial Dynamic Voting interface that may allow a consensual command to be automatically extracted.

We define a votel as $v_i = [u, x, y, t]$, where u is a user id, x, y indicate a location in the election image, and t indicates the time when the votel was received at the server. During each election, the server collects V, a set of votels. Given V, we analyze voting patterns in terms of goals and collaboration. To illustrate, consider Figure 3, a typical election image from the field test.

7.1 Goal Identification

What constitutes a "goal" that users are voting on? For typical navigation questions such as "Where should we go next?", or "Who should we speak with next?" the goals are meaningful regions or segments of the image. We propose to use clustering algorithms [22] to identify groups of neighboring votels. After votels are classified into groups, one approach is to compute the convex hull of each group with 3 or more votels and treat each convex polygon as a distinct goal.

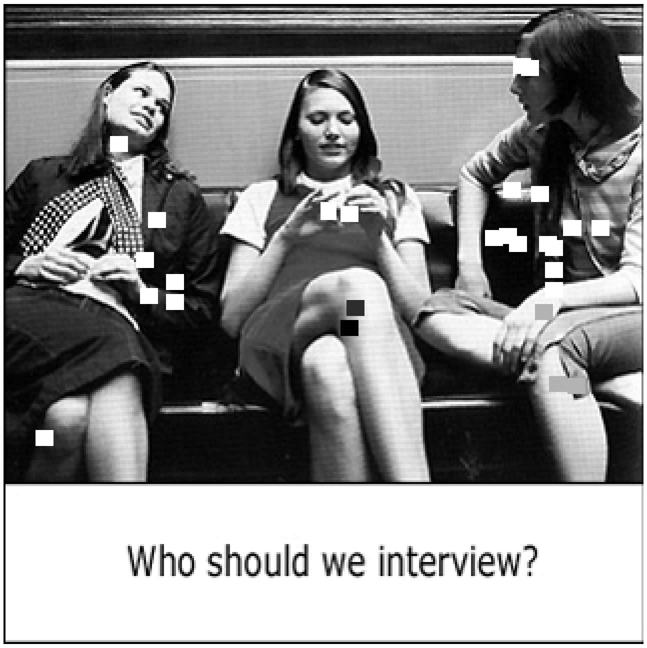


Figure 3: *Election Image with 27 votels from the field test.*

When the Tele-Actor is restricted to movements on a floor, the horizontal position of votels provides the primary navigation information. In such cases we can project all votels onto the horizontal axis and use a nearest neighbor algorithm [39, 6] to perform one-dimensional incremental interval clustering.

For the votels shown in Figure 3, the algorithm identified 3 goal intervals as summarized in Table 1.

Goal	Interval	Width	#Votes	c_i
1	[20, 84]	64	7	1.30
2	[161, 175]	14	4	3.38
3	[244, 296]	52	16	3.65
Overall	–	130	27	2.46

Table 1. SDV analysis of Election Image from Figure 3. Intervals and widths are in pixels.

After all votels are collected, the goal with maximum votes (Goal 3 in the example above) is selected for execution by the Tele-Actor.

7.2 Collaboration Metric

To what degree are voters collaborating? We can define a measure based on how votels are spatially correlated. For each goal i, we can compute a votel density ratio, c_i:

$$c_i = \frac{d_i}{d} = \frac{\frac{n_i}{a_i}}{\frac{N}{A}} = \frac{n_i}{N}\,(\frac{A}{a_i})$$

where d_i is the votel density (votes per unit area) for goal i, d is the overall average votel density, n_i is number of votel in goal i, a_i is the area or width of the goal i, N is the total number of votes and A is the area of the election image. This metric is proportional to the ratio n/N and inversely proportional to the area of the goal region. The metric is high when many votes are concentrated in a small goal region (high collaboration) and low when votes are uniformly spread among multiple goals (low collaboration). We can also compute an overall collaboration level for each election:

$$c = \frac{\sum n_i}{\sum a_i}\frac{A}{N}$$

When all votes fall into goal regions, $c = A/\sum a_i$, a measure of how focused the votels are. The collaboration metric for each goal and overall is given in the last column of Table 1, suggesting that users are collaborating in a focused manner to vote for goal 2 even though it has far fewer votes than goal 3.

8 Future Work

This paper describes the Tele-Actor, a MOSR teleoperation system that allows a group of Internet users to share control of a remote resource using a new “Spatial Dynamic Voting” (SDV) interface. The mobile resource could be a robot or human. We described an implemented system, algorithms for automatic analysis of voting patterns, and preliminary results from a sixty-minute field test with 56 online voters.

Informal feedback from users indicated that the one-minute election cycle was too long, voters grew impatient. We are currently improving the system with a new Java-based user interface that will display moving votels and reduce election cycle times. We will also incorporate the 802.11b wireless ethernet standard to transmit low-frame-rate digital video. We are also studying fast algorithms for incremental clustering and automated goal selection that meet reasonable fairness criteria. To speed goal selection and provide incentives for collaboration, we are also developing a “leadership” metric that can be used to weight user votes. Our goal is to develop a practical collaborative teleoperation system that will facilitate distance education for a broad range of online experiences. To experiment with the latest version, please visit www.tele-actor.net.

Acknowledgments

Thanks to Rebecca Aust, K. ”Gopal” Gopalakrishnan, Cort Worthington, Marina Berliner, Billy Chen, Tiffany Shlain, Maya Draisin, and Kevin George for their contributions to this project, and John Canny, Vladimir Lumelsky, Norman Johnson, Ron Arkin, Larry Leifer, and Paolo Pirjanian for insightful discussions.

References

[1] H. Arai, T. Takubo, Y. Hayashibara, and K Tanie. Human-robot cooperative manipulation using a virtual nonholonomic constraint. In *IEEE International Conference on Robotics and Automation*, April 2000.

[2] R. Arkin. Cooperation without communication: Multiagent schema-based robot navigation. *Journal of Robotic Systems*, 9(3), 1992.

[3] R. Baecker. *Readings in Groupware and Computer Supported Cooperative Work: Assisting Human-Human Collaboration*. Morgan Kaufmann, 1992.

[4] C. Bernard, H. Kang, S. K. Sigh, and J.T. Wen. Robotic system for collaborative control in minally invaive surgery. *Industrial Robot: An international Journal*, 26(6), 1999.

[5] D. M. Blough and G. F.Sullivan. A comparison of voting strategies for fault-tolerant distributed systems. In *9th Symposium on Reliable Distributed Systems*, 1990.

[6] A.J. Broder. Strategies for efficient incremental nearest neighbor search. *PATREC: Pattern Recognition, Pergamon Press*, 23, 1990.

[7] Z. Butler, A. Rizzi, and R. Hollis. Cooperative coverage of rectilinear environments. In *IEEE International Conference on Robotics and Automation*, April 2000.

[8] D. J. Cannon. *Point-And-Direct Telerobotics: Object Level Strategic Supervisory Control in Unstructured Interactive Human-Machine System Environments*. PhD thesis, Stanford Mechanical Engineering, June 1992.

[9] N. Chong, T. Kotoku, K. Ohba, K. Komoriya, N. Matsuhira, and K. Tanie. Remote coordinated controls in multiple telerobot cooperation. In *IEEE International Conference on Robotics and Automation*, April 2000.

[10] P. Coppin and W. Whittaker et al. Eventscope: Amplifying human knowledge and experience via intelligent robotic systems and information interaction. In *9th IEEE International Workshop on Robot and Human Interactive Communication (Ro-man)*, 2000.

[11] C. H. Crouch and E. Mazur. Peer instruction: Ten years of experience and results. *Americian Journal of Physices*, 69(9), 2001.

[12] B. Donald, L. Gariepy, and D. Rus. Distributed manipulation of multiple objects using ropes. In *IEEE International Conference on Robotics and Automation*, April 2000.

[13] I. Elhaji, J. Tan, N. Xi, W. Fung, Y. Liu, T. Kaga, Y. Hasegawa, and T. Fukuda. Multi-site internet-based cooperative control of robotic operations. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2000.

[14] H. Kuzuoka et al. Gestureman: A mobile robot that embodies a remote instructor's actions. In *ACM Conference on Computer Supported Collaborative Learning*, 2000.

[15] T. Fong, C. Thorpe, and C. Baur. Collaboration, dialogue, and human-robot interation. In *10th International Symposium of Robotics Research*, 2001.

[16] L. Gasser. Multi-agent systems infrastructure definitions, needs, and prospects. *Proceedings of the Workshop on Scalable MAS Infrastructure, Barcelona Spain*, 2000.

[17] 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)*, April 2000.

[18] K. Goldberg and Billy Chen. Collaborative control of robot motion: Robustness to error. In *International Conference on Intelligent Robots and Systems (IROS)*, 2001.

[19] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2), July 2001.

[20] K. Goldberg and Roland Siegwart, editors. *Beyond Webcams: An Introduction to Online Robots*. MIT Press, 2002.

[21] H. Hu, L. Yu, P. Tsui, and Q. Zhou. Internet-based robotic systems for teleoperation. *Assembly Automation Journal*, 21(2), 2001.

[22] A. Jain, P. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Transactions on PAMI*, 22(1), 2000.

[23] S. Jia and K. Takase. A corba-based internet robotic system. *Advanced Robotics*, 15(6), 2001.

[24] N. L. Johnson, S. Rasmussen, and M. Kantor. New frontiers in collective problem solving. Technical Report NM LA-UR-98-1150, May 1998.

[25] Y. W. Leung. Maximum likelhood voting for fault-tolerant software with finite output space. *IEEE Transactions on Reliability*, 44(3), 1995.

[26] K. Lynch and C. Liu. Designing motion guides for ergonomic collaborative manipulation. In *IEEE International Conference on Robotics and Automation*, April 2000.

[27] M. McDonald, D. Small, C. Graves, and D. Cannon. Virtual collaborative control to improve intelligent robotic system efficiency and quality. In *IEEE International Conference on Robotics and Automation*, April 1997.

[28] D. G. Meyers, editor. *Social Psychology*. McGraw-Hill, 1996.

[29] E. Paulos and J. Canny. Ubiquitous tele-embodiment: Applications and implications. *International Journal of Human-Computer Studies/Knowledge Acquisition*, 1997. Special Issue on Innovative Applications of the World Wide Web.

[30] E. Paulos and J. Canny. Designing personal tele-embodiment. In *IEEE International Conference on Robotics and Automation (ICRA)*, Detroit, MI, 1999.

[31] P. Pirjanian. *Multiple Objective Action Selection and Behavior Fusion Using Voting*. PhD thesis, Aalbord University, 1998.

[32] P. Pirjanian and M. Mataric. Multi-robot target acquisition using multiple objective behavior coordination. In *IEEE International Conference on Robotics and Automation*, April 2000.

[33] Rachel and Loren Carpenter. www.cinematrix.com.

[34] S. Rasmussen and N.L. Johnson. Self-organization in and around the internet. In *Proceedings of 6th Santa Fe Chaos in Manufacturing Conference*, April 1998.

[35] I. Rekleitis, G. Dudek, and E. Milios. Multi-robot collaboration for robust exploration. In *IEEE International Conference on Robotics and Automation*, April 2000.

[36] B. Rogoff, E. Matusov, and C. White. *Models of teaching and learning: Participation in a community of learners*. Oxford, England: Blackwell, 1996.

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

[38] R. Siegwart and P. Saucy. Interacting mobile robots on the web. In *IEEE International Conference on Robotics and Automation (ICRA)*, 1999.

[39] G. T. Toussaint. The relative neighbourhood graph of a finite planar set. *Pattern Recognition*, 12, 1980.