Chapter 1

Networked Telerobots

1.1 Overview and Background

Telerobots, remotely controlled robots, are widely used to explore undersea terrains and outer space, to defuse bombs, and to clean up hazardous waste. Until 1994, telerobots were accessible only to trained and trusted experts through dedicated communication channels. This chapter describes Networked Telerobots, a new class of telerobots controllable over networks such as the Internet that are accessible to the general public. This chapter will describe relevant network technology, the history of networked telerobots within the broader field of teleoperation, properties of networked telerobots, how to build a networked robot, example systems, and topics for future research.

As illustrated below, the broader field of Teleoperation, where primary concerns are stability and time delay, is covered in Chapter 35. The field of Networked *robots*, where autonomous robots and sensors communicate over local networks, is covered in Chapter 44. Networked *telerobots*, the subject of the present chapter, focuses on teleoperated robot systems that are accessible by the public via web browsers.

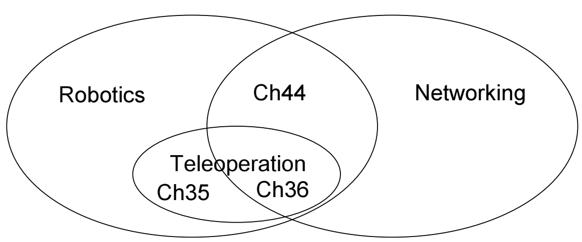

Figure 1.1: Relationship between the subjects of Networked Telerobots (Ch 36, the present chapter), Teleoperation (Ch 35), and Networked Robots (Ch 44).

As of 2006, several hundred networked telerobots have been developed and put online for public use. Many papers have been published describing these systems and a book on this subject by Goldberg and Siegwart is available from MIT Press [23]. Updated information about new research and an archive/survey of networked telerobots is available on the website of the IEEE Technical Committee on Networked Robots (Google “networked robots”), which fosters research in both networked telerobots and networked robots.

Networked telerobots have the following properties:

- The physical world is affected by a device that is locally controlled by a network "server" which communicates with remote human users through web browsers such as Netscape, which are generally referred to as "clients." As of 2006, the standard protocol for network browsers is HTTP, a stateless transmission protocol.
- Most networked telerobots are continuously accessible (online), 24 hours a day, 7 days a week.
- Since hundreds of millions of people now have access to the Internet, mechanisms are needed to handle client authentication and contention.
- Input and output for networked telerobots is usually achieved with the standard computer screen, mouse, and keyboard.
- Clients may be inexperienced or malicious, so online tutorials and safeguards are generally required.

1.2 A Brief History

Like many technologies, remotely controlled devices were first imagined in science fiction. In 1898, Nicola Tesla [55] demonstrated a radio-controlled boat in New York's Madison Square Garden. The first major experiments in teleoperation were motivated by the need to handle radioactive materials in the 1940s. Goertz demonstrated one of the first bilateral simulators in the 1950's at the Argonne National Laboratory [18]. Remotely operated mechanisms have been designed for use in inhospitable environments such as undersea [5] and space exploration [6]. At General Electric, Mosher [42] developed a two-arm teleoperator with video cameras. Prosthetic hands were also applied to teleoperation [56]. More recently, teleoperation is being considered for medical diagnosis [1], manufacturing [17] and micromanipulation [53]. See Chapter 35 and the book from Sheridan [49] for excellent reviews on teleoperation and telerobotics research.

The concept of hypertext (linked references) was proposed by Vannevar Bush in 1945 and was made possible by subsequent developments in computing and networking. In the early 1990's, Berners-Lee introduced the Hypertext Transmission Protocol (HTTP). A group of students led by Marc Andreessen developed an open source version of the first graphical user interface, the "Mosaic" browser, and put it online in 1993. The first networked camera, or "webcam", went online in November 1993 [27]

Approximately nine months later, the first networked telerobot went online. The "Mercury Project" combined an IBM industrial robot arm with a digital camera and used the robot's air nozzle to allow remote users to excavate for buried artifacts in a sandbox [22,34]. 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 [3,26]. These early projects pioneered a new field of networked telerobots. See [25,28,32,33,36,39,41,43,48] for other examples.

Networked telerobots are a special case of “supervisory control” telerobots, as proposed by Sheridan and his colleagues [49]. Under supervisory control, a local computer plays an active role in closing the feedback loop. Most networked robotics are type (c) supervisory control systems.

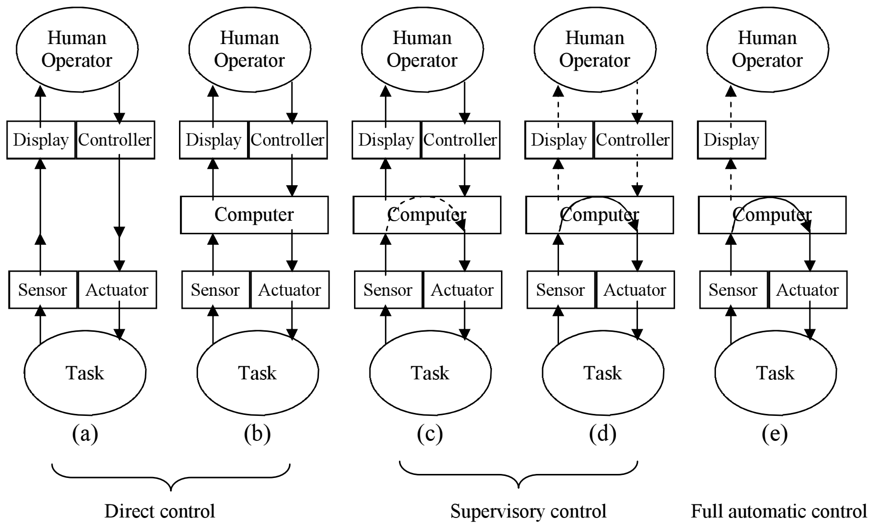

Figure 1.2: A spectrum of teleoperation control modes adapted from Sheridan’s text [49]. We label them a-e, in order of increasing robot autonomy. At the far left would be a mechanical linkage where the human directly operates the robot from another room through sliding mechanical bars, and on far right would be a system where the human role is limited to observation/monitoring. In c-e, the dashed lines indicated that communication may be intermittent.

Although a majority of networked telerobotic systems consist of a single human operator and a single robot [4, 11, 12, 29–31, 44, 58], Tanie, Matsuhira, Chong, et al. [8] propose a useful taxonomy : Single Operator Single Robot (SOSR), Single Operator Multiple Robot (SOMR), Multiple Operator Single Robot (MOSR), and Multiple Operator Multiple Robot (MOMR).

The decade from 1995-2005 witnessed the extensive development in networked telerobots. New systems, new experiments and new applications go well beyond traditional fields such as defense, space, and nuclear material handing [49] that motivated teleoperation in early 1950s. As the Internet introduces universal access to every corner of life, the impact of networked robots becomes broader and deeper in modern society. Recent applications range from education, industry, commercial, health care, geology, environmental monitoring, to entertainment and arts.

Networked telerobots provide a new medium for people to interact with remote environment. A networked robot can provide more interactivity beyond what a normal videoconferencing system. The physical robot not only represents the remote person but also transmits multi-modal feedback to the person, which is often referred as “telepresence” in literature [44]. Paulos and Canny’s Personal ROving Presence (PRoP) robot [45] and Jouppi and Thomas’ Surrogate robot [44] are recent representative work.

Networked telerobots have great potential for education and training. In fact, one of the earliest networked telerobot systems [54] originates from the idea of a remote laboratory. Networked telerobots provide universal access to the general public, who may have little to no knowledge of robots, with opportunities to understand, learn, and operate robots, which were expensive scientific equipment limited to universities and large corporate laboratories before. Built on networked telerobots, online remote laboratories [10, 35] greatly improves distance learning by providing an interactive experience. For example, teleoperated telescopes help students to understand astronomy [13]. Teleoperated microscope [47] helps student to observe micro-organisms. The Tele-Actor project [24] allows a group of students to remotely control a human tele-actor to visit environments that are normally not accessible to them such as clean-room environments for semi-conductor manufactory facility and DNA analysis laboratories.

1.3 Communications and Networking

Below is a short review of relevant terminologies and technologies on networking. For details, see texts by [57].

A communication network includes three elements: *links*, *routers/switchers*, and *hosts*. Links refer to the physical medium that carry bits from one place to another. Examples of links include copper or fiber optic cables and wireless (radio frequency or infrared) channels. Switches and routers are hubs that direct digital information between links. Hosts are communication end points such as browsers, computers, and robots.

Networks can be based in one physical area (Local Area Network, or LAN), or distributed over wide distances (Wide Area Network, or WAN). Access control is a fundamental problem in networking. Among a variety of methods, the "Ethernet" protocol is the most popular. Ethernet provides a broadcast-capable, multi-access LAN. It adopts a Carrier Sense Multiple Access (CSMA) strategy to address the multi-access problem. Defined in the IEEE 802.x standard, CSMA allows each host to send information over the link at any time. Therefore, collisions may happen between two or more simultaneous transmission requests. Collisions can be detected either by directly sensing the voltage in case of wired networks, which is referred to as Collision Detection (CSMA/CD) or by checking the time-out of an anticipated acknowledgement in wireless networks, which is referred to as Collision Avoidance (CSMA/CA). If a collision is detected, both/all senders randomly back off a short period of time before the re-transmission. CSMA has a number of important properties: 1) it is a completely decentralized approach, 2) it does not need clock synchronization over the entire network, and 3) it is very easy to implement. However, the disadvantages of CSMA are: 1) the efficiency of the network is not very high and 2) the transmission delay could change drastically.

As mentioned previously, LANs are inter-connected with each other via routers/switchers. The information transmitted is in packet format. A packet is a string of bits and usually contains source address, destination address, content bits, and a checksum. Routers/switchers distribute packets according to their routing table. Routers/switchers have no memory of packets, which ensures the scalability. Packets are usually routed according to a First-In-First-Out (FIFO) rule, which is independent of the application. The packet formats and addresses are independent of host technology, which ensures extensibility. This routing mechanism is referred to as packet switching in networking literature. It is quite different from a traditional telephone network, which is referred to as circuit switching. A telephone network is designed to guarantee a dedicated circuit between a sender and a receiver once a phone call is established. The dedicate circuitry ensures communication quality. However, it requires a large number of circuits to ensure the quality of service, which leads to poor utilization of the overall network. A packet switching network cannot guarantee dedicated bandwidth for each individual pair of transmissions, but it improves overall resource utilization. The Internet, which is the most popular communication media and the infrastructure of networked telerobots, is a packet switching network.

1.3.1 The Internet

The creation of the Internet can be traced back to U.S. Department of Defense (DoD)'s APRA NET network in 1960s. There are two features of the APRA NET network that enabled the successful evolution of the Internet. One feature is the ability for information(packets) to be re-routed around failures. Originally this was designed to ensure communication in event of a nuclear war. Interestingly, the dynamic routing capability of also allows the topology of the Internet to grow easily. The second important feature is the ability for heterogeneous networks to interconnect with one another. Heterogeneous networks, such as X.25, G.701, Ethernet, can all connect to the Internet as long as they can implement Internet Protocol (IP). The IP is media, OS, and data rate independent. This flexible design allows a variety of applications and hosts to connect to the Internet as long as they can generate and understand IP.

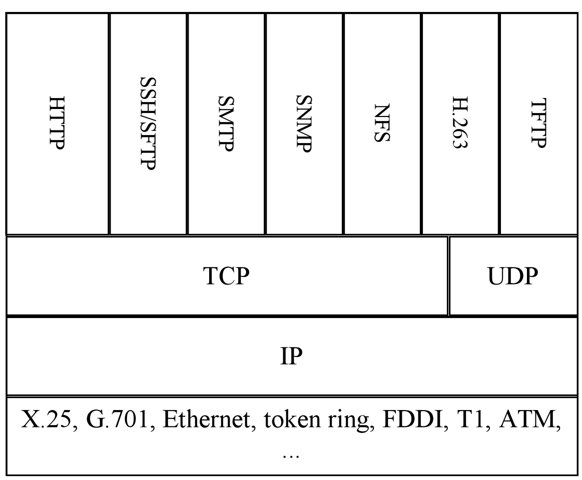

Figure 1.3: A 4-layer model of Internet protocols from [57].

Figure 1.3 illustrates a 4-layer model of the protocols used in the Internet. On the top of the IP, we have two primary transport layer protocols, Transmission Control Protocol (TCP) and User Data Protocol (UDP). TCP is an end-to-end transmission control protocol. It manages packet ordering, error control, rate control, and flow control based on packet round-trip time. TCP

Types	bits per second
Modem (V.92)	Up to 56K
ISDN BRI	$64 - 128$K
HDSL	1.544M duplex on two twisted-pair lines
ADSL	$1.544 - 6.1$M downstream, $16 - 640$K upstream
Cable modem	$2 - 4$M downstream, $400 - 600$K upstream
Fiber to the home(FTTH)	$5 - 30$M downstream, $2 - 5$M upstream[2]
Internet II/III node	≥ 1G

Table 1.1: *"Last-mile" Internet speed by wired connection type. If not specified, the downstream transmission and the upstream transmission share the same bandwidth.*

guarantees the arrival of each packet. However, the excessive re-transmission of TCP in a congested network may introduce undesirable time-delays in a networked telerobotic system. UDP behaves differently. It is a broadcast-capable protocol and does not have a re-transmission mechanism. Users must take care of error control and rate control themselves. UDP has a lot less overhead compared to TCP. UDP packets are transmitted at the sender's preset rate and change the rate based on the congestion of a network. UDP has great potential, but it is often blocked by firewalls because of a lack of a rate control mechanism. It is also worth mentioning that the widely-accepted term "TCP/IP" refers to the family of protocols that build on IP, TCP, and UDP.

In the application layer of the Internet protocols, Hyper Text Transmission Protocol (HTTP) is one of the most important protocols. HTTP is the protocol for World Wide Web (WWW). It allows the sharing of multimedia information among heterogeneous hosts and OSs including text, image, audio, and video. The protocol has significantly contributed to the boom of the Internet. It also changes the traditional Client/Server (C/S) communication architecture to a Browser/Server (B/S) architecture. A typical configuration of the B/S architecture consists of a web server and clients with web browsers. The webserver projects the contents in Hyper Text Markup Language (HTML) format or its variants, which is transmitted over the Internet using HTTP. User inputs can be acquired using Common Gateway Interface (CGI) or other variants. The B/S architecture is the most accessible because no specialized software is needed at client end.

1.3.2 Wired Communication Links

Even during peak usage, the network backbones of the Internet often run at less than 30% of their overall capacity[1]. The average backbone utilization is around $15 - 20\%$. The primary speed limitation for the Internet is the "last-mile," the link between clients and their local Internet Service Providers (ISP).

Table 1.1 lists typical bitrates for different connection types. It is interesting to note the asymmetric speeds in many cases, where upstream bitrate (from client to the Internet), are far slower than downstream bitrates (from the Internet to the client). These asymmetries introduce complexity into the network

[1] http://www.telegeography.com/

model for teleoperation. Since the speed difference between the slowest modem link and the fastest Internet II node is over $10,000$, designers of a networked telerobotic system should anticipate a large variance of communication speeds.

1.3.3 Wireless Links

Table 1.2 compares speed, band, and range of wireless standards as of 2006. Increasing bit rate and communication range requires increasing power. The amount of RF transmission power required over a distance d is proportional to d^k, where $2 \leq k \leq 4$ depends on the antenna type. In Table 1.2, Bluetooth and Zigbee are typical low power transmission standards that are good for short distances. WiMax and MWBA are currently under development.

Types	Bit rate (bps)	Band (Hz)	Range (m)
Zigbee (802.15.4)	$20-250$K	$868-915$M/2.4G	50
3G Cellphone	400K$-$1.15M	≤ 3.5G	15k
Bluetooth	732K	2.4G	100
MWBA (802.20)	1M	$\leq$3.5G	15k
WiFi (802.11a,b,g)	$11-54$M	2.4G or 5.8G	100
WiMax(802.16)	70M	$2-11, 10-66$G	50k

Table 1.2: *Survey of wireless technologies in terms of bitrate and range.*

By providing high speed connectivity at a low cost, WiFi is the most popular wireless standard in 2006. Its range is approximate 100m line of sight and the WiFi wireless network usually consists of small scale inter-connected access points. The coverage range usually limits these networks to an office building, home, and other indoor environments. WiFi is a good option for indoor mobile robots and human operators. If the robot needs to navigate in outdoor environment, 3G cellphone network can provide the best coverage available. Although obvious overlap exists among wireless standards in coverage and bandwidth, there are two import issues that have not been covered by Table 1.2. One is mobility. We know that if a RF source or receiver is moving, the corresponding Doppler effect causes frequency shift, which could cause problems in communication. WiFi is not designed for fast moving hosts. WiMax and 3G Cellphone allows the host to move at a vehicle speed under 120km/hr. However, MWBA allows the host to move at a speed of 250km/hr, which is the only protocol that works for high-speed trains. Both WiMax and MWBA are designed with a transmission latency less that 20ms. However, 3G cellphone networks have a variable latency ranging from 10ms to 500ms.

1.3.4 Properties of Networked Telerobotics

As defined by Mason, Peshkin, and others [40, 46], in "quasi-static" robot systems, accelerations and inertial forces are negligible compared to dissipative forces. In quasi-static robot systems, motions are often modeled as transitions between discrete atomic "configurations".

We adopt a similar terminology for networked telerobots. In Quasistatic Telerobotics (QT), robot dynamics and stability are handled locally. After each atomic motion, a new state report is presented to the remote user, who sends back an atomic command. The atomic state describes the status of the robot and its corresponding environment. Atomic commands refer to human directives, which are desired robotic actions.

Several issues arise:

- *State-Command Presentation*: How should state and available commands be presented to remote human operators using the 2D screen display?
- *Command Execution/State Generation*: How should commands be executed locally to ensure that the desired state is achieved and maintained by the robot?
- *Command Coordination*: How should commands be resolved when there are multiple human operators?

1.3.5 Building a Networked Telerobotic System

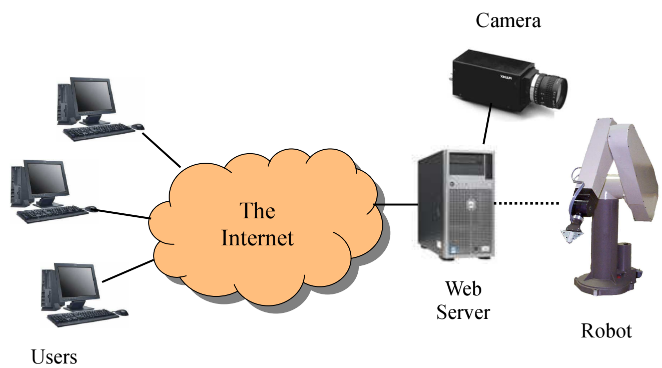

Figure 1.4: Typical system architecture for a networked telerobot.

As illustrated in Figure 1.4, a typical networked telerobotic system typically includes three components:

- users: anyone with an Internet connection and a web browser,
- web server: a computer running a web server software.
- robot: a robot manipulator, a mobile robot, or any device that can modify or affect its environment.

Users access the system via their web browsers. Any web browser that is compatible with W3C's HTML standard [3] can access a web server. In 2006, the most popular web browsers are Microsoft Internet Explorer, Netscape, Mozilla Firefox, Safari, and Opera. New browsers and updated versions with new features are introduced periodically.

A web server is a computer that responds to HTTP requests over the Internet. Depending upon the operating system of the web server, popular server software packages include Apache [4] and Microsoft Internet Information Services (IIS) [5]. Most servers can be freely downloaded from the Internet.

[3] http://www.w3.org/
[4] http://www.apache.org
[5] http://www.microsoft.com/WindowsServer2003/iis/default.mspx

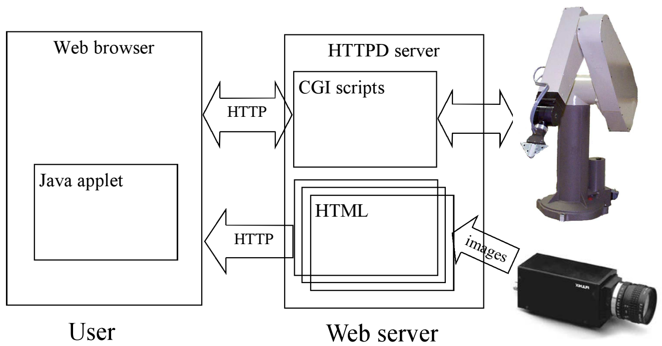

Figure 1.5: A sample software architecture of a networked telerobot.

To develop a networked telerobot, one needs a basic knowledge of developing, configuring, and maintaining web servers. As illustrated in Figure 1.5, the development requires knowledge of HTML and at least one local programming languages such as C, CGI, Javascript, Perl, Php, or Java.

It is important to consider compatibility with the variety of browsers. Although HTML is designed to be compatible with all browsers, there are exceptions. For example, Javascript, which is the embedded scripting language of web browsers, is not completely compatible between Internet Explorer and Netscape. One also needs to master the common HTML components such as forms that are used to accept user inputs, frames that are used to divide the interface into different functional regions, etc. An introduction to HTML can be found in [37].

User commands are usually processed by the web server using CGI, the Common Gateway Interface. Most sophisticated methods such as PHP[6], JSP, and socket programming, can also be used. CGI is invoked by the HTTP server when the CGI script is referred in the URL. The CGI program then interprets the inputs, which is often the next robot motion command, and send commands to the robot via a local communication channel. CGI scripts can be written in almost any programming language. The most popular ones are PERL[7] and C.

A simple networked telerobotic system can be constructed using only HTML and CGI. However, if the robot requires a sophisticated control interface, a Java applet is recommended. Java applets run inside the web browser on the client's computer. Information about Java can be found at Sun Microsystems' official Java home page [8].

Most telerobotic systems also collect user data and robot data. Therefore, database design and data processing program are also needed. The most common used databases include MySQL [9] and PostgresSQL [10]. Both are open source databases and support a variety of platforms and Operation Systems. Since a networked telerobotic system is online 24 hours a day, reliability is also an important consideration in system design. Web site security issue is critical. Other common auxiliary developments include online documentation, online

[6]http://www.php.net/
[7]http://www.perl.org/
[8]http://www.java.com/
[9]http://www.mysql.com/
[10]http://www.postgresql.org/

manual, and user feedback collection.

1.3.6 State-Command Presentation

To generate a correct and high quality command depends on how effectively the human operator understands the state feedback. The state-command presentation contains three sub-problems: the 2d representation of the true robot state (state display), the assistance provided by the interface to generate new commands (spatial reasoning), and input mechanism.

Browser Displays

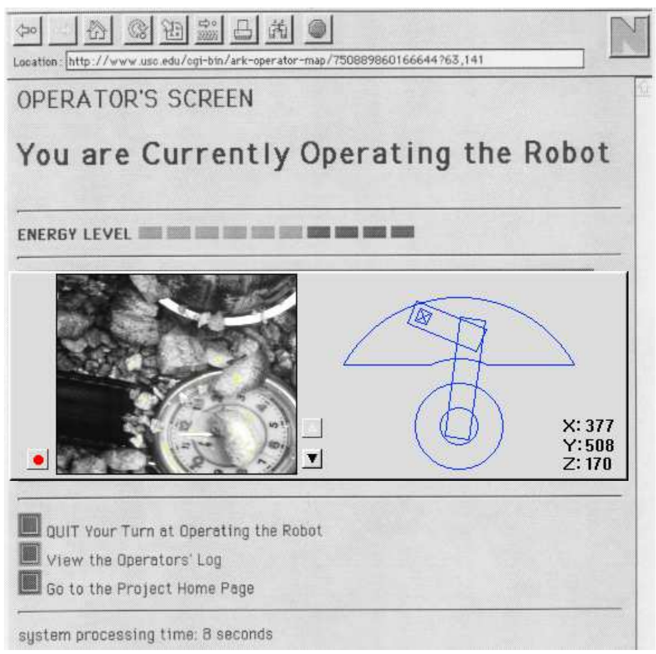

Figure 1.6: Browser's view of the first networked telerobot interface [21]. The schematic at lower right gives an overhead view of position of the 4-axis robot arm (with camera at end marked with X), and the image at lower left indicates the current view of the camera. The small button marked with a dot at the left directs a one-second burst of compressed air into the sand below the camera....The Mercury Project was online from August 1994 - March 1995.

Unlike traditional point-to-point teleoperation, where specialized training and equipment are available to operators, networked telerobots offer wide access to the general public. Designers cannot assume operators have any prior experience with robots. As illustrated in Figure 1.6, networked telerobotic systems must display the robot state on a 2D screen display.

The states of the teleoperated robot are often characterized in either world coordinates or robot joint configuration, which are either displayed in numerical format or through a graphical representation. Figure 1.6 lists robot XYZ coordinates on the interface and draw a simple 2D projection to indicate joint

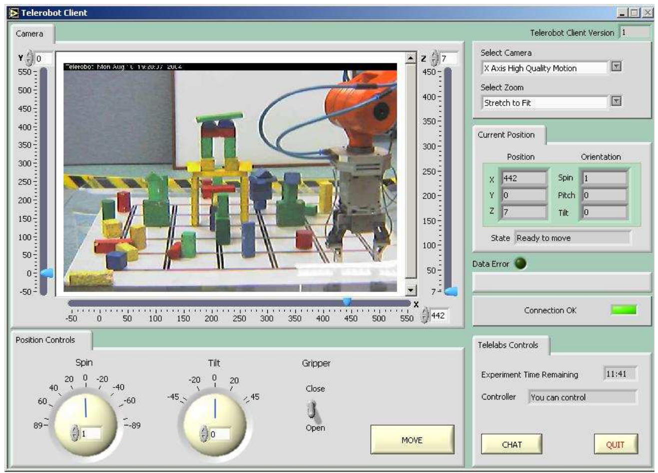

Figure 1.7: Browser Interface to the Australian networked telerobot which was a 6-axis arm that could pick up and move blocks. [26].

configurations. Figure 1.7 illustrates another example of teleoperation interface that is developed by Taylor and Trevelyan [54]. In this interface, XYZ coordinates are presented in a sliding bar near the video window.

The state of the robot is usually displayed in a 2D view as shown in Figure 1.6 and Figure 1.7. In some systems, multiple cameras can help the human operator to understand the spatial relationship between the robot and the objects in the surrounding environment. Figure 1.8 shows an example with four distinct camera views for a six degree of freedom industrial robot.

Figure 1.9 demonstrate an interface with a pan-tilt-zoom robotic camera. The interface in Figure 1.9 is designed for a mobile robot.

More sophisticated spatial reasoning can eliminate the need for humans to provide low level control by automatically generating a sequence of commands after it receives task-level commands from the human operator. This is particularly important when the robotic system is highly dynamic and requires very fast response. In this case, it is impossible to ask the human to generate intermediate steps in the robot control. For example, Belousov et al. adopts a shared autonomy model to direct a robot to capture a moving rod [29]. Fong and Thorpe [14] summarize vehicle teleoperation systems that utilize these supervisory control techniques. Su et al. develops an incremental algorithm for better translating the intension and motion of operators to remote robot action commands [31].

Human Operator Input

Most networked telerobotic systems only rely on mouse and keyboards for input. The design problem is what to click on in the interface. Given the fact that user commands can be quite different, we need to adopt appropriate interface for inputs. For example, inputs could be Cartesian XYZ coordinates in world coordinate system or robot configurations in angular joint configurations.

For angular inputs, it is often suggested to use a round dial as control interface as illustrated in bottom left of Figure 1.7 and right hand side of

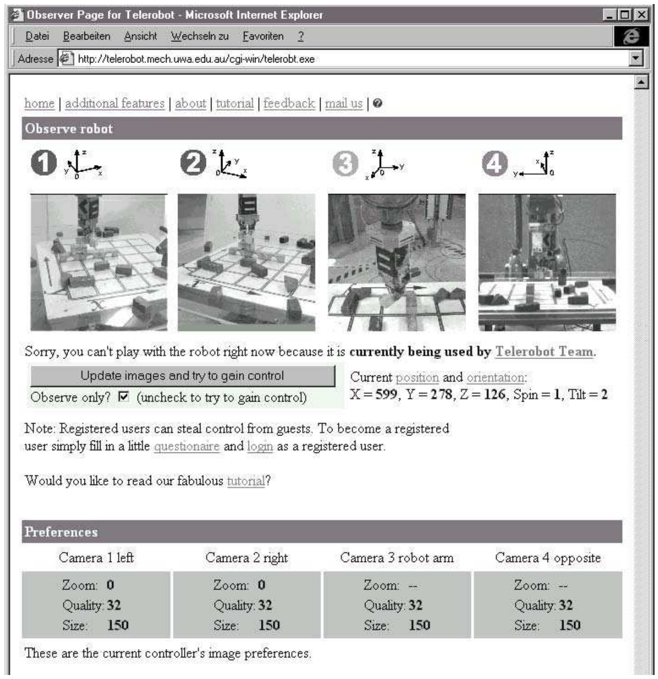

Figure 1.8: Use of multi-camera system for multi-view point state feedback [15].

Figure 1.9. For linear motion in Cartesian coordinate, arrows operated by either mouse click or keyboard are often suggested. Position and speed control are often needed as illustrated in Figure 1.9. Speed control is usually controlled by mouse click on a linear progress bar for translation and dial for rotation.

The most common control type is position control. The most straight forward way is to click on the video image directly. To implement the function, the software needs to translate the 2D click inputs into 3D world coordinates. To simplify the problem, the system designer usually assume the clicked position is on a fixed plane. For example, a mouse click on the interface of Figure 1.6 assumes the robot moves on X-Y plane. The combination of a mouse click on the image can also allow abstract task level command. The example in Figure 1.11 uses mouse clicks to place votes on an image to generate a command that directs a robot to pick up a test agent at task level.

1.3.7 Command Execution/State Generation

When a robot receives a command, it executes the command and a new state is generated and transmitted back to the human operator. However, commands many not arrive in time or may get lost in the transmission. Also, Because users are often inexperienced, their commands may contain errors.

Belousov and his colleagues demonstrate a system that allow a web user to capture a fast rod that is thrown at a robot manipulator [29]. Over the limited communication channel, it is impossible to ask the human to control the manipulator directly. Computer vision-and augmented reality based local intelligence is required to assist the human operator. The rod is on bifilar suspension, per-

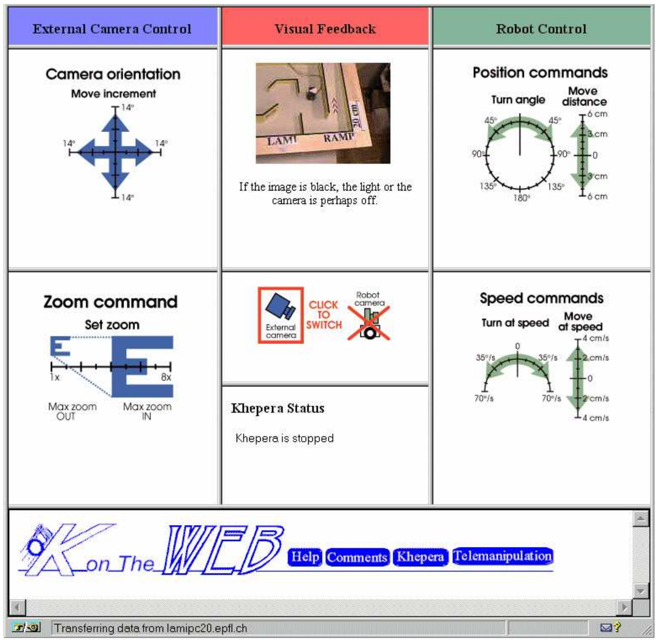

Figure 1.9: Camera control and mobile robot control in Patrick Saucy and Francesco Mondada's Khep on the web project.

(a) (b)

Figure 1.10: A web-based teleoperation system that allows robot to capture a fast moving rod [29]: (a) User interface and (b) System setup.

forming complicated oscillations. Belousov et al. design a shared autonomy control to implement the capture. First, an operator chooses desired point for capture on the rod and capture instant using a 3D online virtual model of the robot and the rod. Then, the capturing operation is performed automatically using a motion prediction algorithm that is based on the rods motion model and two orthogonal camera inputs, which perceive the rod's position locally in real time.

1.3.8 Collaborative Control

When more than one human is sharing control of the device, command coordination is needed. According to [20], multiple human operators can reduce the chance of errors, cope with malicious inputs, utilize operators' different expertise, and train new operators. In [50], a *collaborative telerobot* is defined as a telerobot simultaneously controlled by many participants, where input from each participant is combined to generate a single control stream.

When group inputs are in form of direction vectors, averaging can be used as aggregation mechanism [19]. When decisions are distinct choices or at abstract task level, voting is a better choice [24]. As illustrated in Figure 1.11, Goldberg and Song develop the Tele-Actor system using Spatial Dynamic Voting. The Tele-Actor is a human equipped with audio/video device and controlled by a group of online users. Users indicate their intensions by positioning their votes on a 320×320-pixel voting image during the voting interval. Votes are collected at the server and used to determine the Tele-Actor's next action based on the most requested region on the voting image . (See: http://www.tele-actor.net).

Figure 1.11: Spatial Dynamic Voting interface for Tele-Actor system [24]. The Spatial Dynamic Voting (SDV) interface as viewed by each user. In the remote environment, the Tele-Actor takes images with a digital camera which are transmitted over the network and displayed to all participants with a relevant question. With a mouse click, each user places a color-coded marker (a "votel" or voting element) on the image. Users view the position of all votels and can change their votel positions based on the group's response. Votel positions are then processed to identify a "consensus region" in the voting image that is sent back to the Tele-Actor. In this manner, the group collaborates to guide the actions of the Tele-Actor.

Song and Goldberg [51, 52] develop a controllable camera that allows many clients to share control its camera parameters as illustrated in Figure 1.12. Users indicate the area they want to view by drawing rectangles on a panoramic

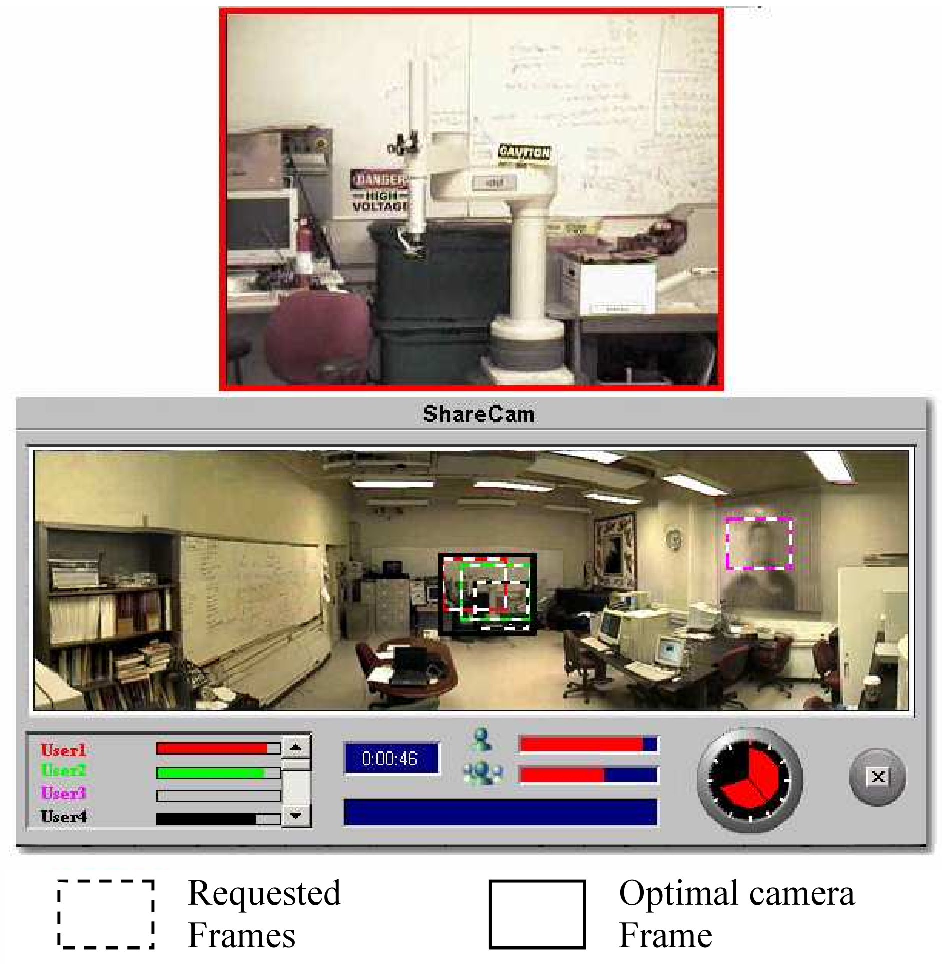

Figure 1.12: Frame selection interface [52].The user interface includes two image windows. The lower window displays a fixed panoramic image based on the camera's full workspace (reachable field of view). Each user requests a camera frame by positioning a dashed rectangle in the lower window. Based on these requests, the algorithm computes an optimal camera frame (shown with solid rectangle), moves the camera accordingly, and displays the resulting live streaming video image in the upper window.

image. The algorithm will compute an optimal camera frame with respect to the user satisfaction function, which is defined as Frame Selection problem.

1.4 Conclusion and Future Directions

As the technology gets matured, networked telerobots gradually go beyond university laboratories and find its application in real world. A new project, Collaborative Observatory for Nature Environments (CONE) project that is proposed by Song and Goldberg[11], aims to design networked robotic camera system to collect data from the wilderness for natural scientists. The fast development of networked telerobot system is not limited to North America. Japan's ATR Intelligent Robotics and Communication Laboratory announces its Networked Robot Project Led by Norihiro Hagita (ATR). Its mission is to develop network-based intelligent robots for applications such as service, medical, and safety. Hideyuki Tokuda (Keio University) chaired "Networked Robot Forum" in Spring 2005. It promotes R&D and standardization on network robots through activities to support awareness campaigns and verification experiments in collaboration among wide-ranging parties, which includes over 100 industry and academic members. Korea's Ministry of Information and Communication announces the Ubiquitous Robotic Companion (URC) Project to develop network-based intelligent robots.

Networked telerobots have allowed tens of thousands of non-specialists around the world to interact with robots. The design of Networked telerobots presents a number of engineering challenges to build reliable systems that can be operated by non-specialists 24/7 and remain online for years. Many new research challenges remain.

- *New Interfaces*: As portable devices such as cellphone and PDA becomes more and more power in computation, networked telerobotics should be able to adopt them as new interface. As a computer becomes more and more powerful, it is capable of visualizing more sophisticated sensor inputs. New interface designer should also keep track of new developments in hardware such as haptic interface and voice recognition. New software standards such as Flash, XML, XHTML, VRML, and WML will also change the way we design interface.

- *New Algorithms*: Algorithms determines performance. Scalable algorithms that are capable of handing large data such as video/ sensor network inputs and utilize fast evolving hardware capability such as distributed and parallel computation will become more and more important in the networked telerobotics.

- *New Protocols*: Although we have listed some pioneering work in changing the network environment to improve teleoperation, there are still a large number of open problems such as new protocols, appropriate bandwidth allocation [38], QoS [16], security, routing mechanisms [58], and many more. Network communication is a very fast-evolving field. To incorporate/modify network communication ideas in networked telerobotic system design will continue to be active research area. The Common Object Request Broken Architecture (CORBA) or Real Time CORBA [2,7,32,33] have great potential for networked telerobotics.

- *Applications*: Many new applications are emerging in areas such as security, inspection, education, and entertainment. Application requirements

[11] www.c-o-n-e.org

such as reliability, security, and modularity will continuously pose new challenges in system design.

Bibliography

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

[2] M. Amoretti, S. Bottazzi, M. Reggiani, and S. Caselli. Evaluation of data distribution techniques in a corba-based telerobotic system. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NV, US*, October 2003.

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

[4] B. Ricks, C. W. Nielsen, and M. A. Goodrich. Ecological displays for robot interaction: a new perspective. In *International Conference on Intelligent Robots and Systems (IROS)*, volume 3, pages 2855 – 2860, 2004.

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

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

[7] S. Bottazzi, S. Caselli, M. Reggiani, and M. Amoretti. A software framework based on real time COBRA for telerobotics systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, EPFL, Lausanne, Switzerland*, October 2002.

[8] 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*, volume 4, pages 3138–3343, April 2000.

[9] L. Conway, R. A. Volz, and M. W. Walker. Teleautonomous systems: Projecting and coordinating intelligent action at a distance. *IEEE Transactions on Robotics and Automation*, 6(20):146–158, Appl. 1990.

[10] C. Cosma, M. Confente, D. Botturi, and P. Fiorini. Laboratory tools for robotics and automation education. In *IEEE Internetional Conference on Robotics and Automation (ICRA), Taipei, ROC*, pages 3303–3308, Sep. 2003.

[11] D. Aarno, S. Ekvall, and D. Kragi. Adaptive virtual fixtures for machine-assisted teleoperation tasks. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1151–1156, 2005.

[12] D. Ryu, S. Kang, M. Kim, and J. Song. Multi-modal user interface for teleoperation of robhaz-dt2 field robot system. In *International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 168–173, 2004.

[13] K. W. Dorman, J. L. Pullen, W. O. Keksz, P. H. Eismann, K. A. Kowalski, and J. P. Karlen. The servicing aid tool: A teleoperated robotics system for space applications. In *The Seventh Annual Workshop on Space Operations Applications and Research (SOAR 1993), Johnson Space Center, Houston, TX*, volume 1, Jan. 1994.

[14] T. Fong and C. Thorpe. Vehicle teleoperation interfaces. *Autonomous Robots*, 11:9–18, 2001.

[15] H. Friz. *Design of an Augmented Reality User Interface for an Internet based Telerobot using Multiple Monoscopic Views.* PhD thesis, Institute for Process and Production Control Techniques, Technical University of Clausthal Clausthal-Zellerfeld, Germany, 2000.

[16] W. Fung, N. Xi, W. Lo, B. Song, Y. Sun, Y. Liu, and I. H. Elhajj. Task driven dynamic qos based bandwidth allcoation for real-time teleoperation via the internet. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NV, US*, October 2003.

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

[18] Raymond Goertz and R. Thompson. Electronically controlled manipulator. *Nucleonics*, 1954.

[19] K. Goldberg and B. Chen. Collaborative teleoperation via the internet. In *International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2001.

[20] 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.

[21] K. Goldberg, M. Mascha, S. Gentner, N. Rothenberg, C. Sutter, and J. Wiegley. Desktop tele-operation via the world wide web. In *IEEE International Conference on Robotics and Automation, Nagoya, Japan.*, May 1995.

[22] 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/.

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

[24] 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.

[25] 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.

[26] **http://telerobot.mech.uwa.edu.au/**.

[27] **http://www.cl.cam.ac.uk/coffee/qsf/timeline.html**.

[28] H. Hu, L. Yu, P. W. Tsui, and Q. Zhou. Internet-based robotic systems for teleoperation. *Assemby Automation*, 21(2):143–151, May 2001.

[29] I. Belousov, S. Chebukov, and V. Sazonov. Web-based teleoperation of the robot interacting with fast moving objects. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 685–690, 2005.

[30] I. Toshima and S. Aoki. Effect of driving delay with an acoustical telepresence robot, telehead. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 56–61, 2005.

[31] J. Su and Z. Luo. Incremental motion compression for telepresent walking subject to spatial constraints. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 69–74, 2005.

[32] 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), Washington, DC, United States*, 2002.

[33] S. Jia and K. Takase. A corba-based internet robotic system. *Advanced Robotics*, 15(6):663–673, Oct 2001.

[34] K. Goldberg, M. Mascha, S. Gentner, N. Rothenberg, C. Sutter, and J. Wiegley. Robot teleoperation via www. In *IEEE International Conference on Robotics and Automation*, May 1995.

[35] A. Khamis, D. M. Rivero, F. Rodriguez, and M. Salichs. Pattern-based architecture for building mobile robotics remote laboratories. In *IEEE Internetional Conference on Robotics and Automation (ICRA), Taipei, ROC*, pages 3284–3289, Sep. 2003.

[36] 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.

[37] Eric Ladd and Jim O'Donnell. *Using Html 4, Xml, and Java 1.2*. QUE Press, 1998.

[38] P. X. Liu, M. Meng, and S.X. Yang. Data communications for internet robots. *Autonomous Robots*, 15:213–223, 2003.

[39] 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.

[40] Matthew T. Mason. On the scope of quasi-static pushing. In O. Faugeras and G. Giralt, editors, *The Third International Symposium on Robotics Research*. MIT Press, 1986.

[41] T. Mirfakhrai and S. Payandeh. A delay prediction approach for teleoperation over the internet. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2002.

[42] R. S. Mosher. Industrial manipulators. *Scientific American*, 211(4), 1964.

[43] 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.

[44] N. P. Jouppi and S. Thomas. Telepresence systems with automatic preservation of user head height, local rotation, and remote translation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 62–68, 2005.

[45] Eric Paulos, John Canny, and Francesca Barrientos. Prop: Personal roving presence. In *SIGGRAPH Visual Proceedings*, page 99, August 1997.

[46] Michael A. Peshkin and Art C. Sanderson. Minimization of energy in quasi-static manipulation. *IEEE Transactions on Robotics and Automation*, 5(1), February 1989.

[47] C. Pollak and H. Hutter. A webcam as recording device for light microscopes. *Journal of Computer-Assisted Microscopy*, 10(4):179–83, 1998.

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

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

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

[51] D. Song and K. Goldberg. Sharecam part I: Interface, system architecture, and implementation of a collaboratively controlled robotic webcam. In *IEEE/RSJ International Conference on Intelligent Robots (IROS), Las Vegas, NV*, volume 2, pages 1080 – 1086, Oct. 2003.

[52] 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), Las Vegas, NV*, volume 2, pages 1087 – 1093, Oct. 2003.

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

[54] K. Taylor and J.P. Trevelyan. Australias telerobot on the web. In *26th Symposium on Industrial Robotics, Singapore*, pages 39–44, Oct. 1995.

[55] N. Tesla. Method of and apparatus for controlling mechanism of moving vessels or vehicles. *http://www.pbs.org/tesla/res/613809.html*, 1898.

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

[57] J. Walrand and P. Varaiya. *High-Performance Communication Networks, 2nd Edition*. Morgan Kaufmann Press, 2000.

[58] Z. Cen, A. Goradia, M. Mutka, N. Xi, W. Fung, and Y. Liu. Improving the operation efficiency of supermedia enhanced internet based teleoperation via an overlay network. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 691–696, 2005.