

Smart Telerobotic Surveillance System via Internet with Reduced Time Delay

Ashesh Vasalya, Rohit Agrawal

School of Electrical Engineering, VIT University, India

Article Info

Article history:

Received Oct 3, 2012

Revised Dec 27, 2012

Accepted Jan 10, 2013

Keyword:

Mobile Robot
Surveillance
Teleoperation
Fuzzy System
Grid Scanning
Internet
Matlab Simulink

ABSTRACT

This paper presents a distributed platform that allows the special group of user to control a gadget (possibly a robot) through internet as a medium. An advanced version of this technology is capable of transmitting graphic images and other surrounding information as required, via internet back to the user to facilitate the effective monitoring of the existent situation using appropriate software tools. The project uses the SRV-1 Mobile Surveillance Robot which is a fully integrated system standard designed and other related technology for surveillance purposes. To resolve this issue, a fuzzy logic controller is used to control the robot's motion along a predefined path with the necessary manipulation of the normal course. The robot was first modelled in Matlab Simulink and the fuzzy logic rules were optimized for the best results possible. In accordance with the fuzzy rules developed the fuzzy interference system generates the output map for operating IR ranger sensor data. This system was developed to actuate as an auxiliary intelligence in the teleoperation system developed for the mobile robot, and the Grid Scanning algorithm was induced to enhance accuracy to cope up with the unexpected delays from the internet data communication. The logic in the internet based controlling of robotics can be expanded to a very large field, like speed control, trajectory control, obstacle avoidance and so on.

Copyright © 2013 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Ashesh Vasalya,
School of Electrical Engineering,
VIT University,
Vellore-632014, Tamil Nadu, India.
Email: asheshvasalya@yahoo.in

1. INTRODUCTION

We have designed this teleoperated server in order to allow users throughout the world to visit actual remote spaces and exhibits. It also serves as a useful scientific tool by promoting discussion about the physical specimens in the browser such as insects, live reptiles, rare museum collections, and recently discovered artefacts.

Modern telepresence systems address in addition to visual, also auditory and haptic senses to provide information to a human teleoperator. This active area of research in remote control analyzes the integration of multimodal sensor inputs, the compensation of time delay effects, and the synchronization of teleoperation/autonomous control reactions.

Training operatives to work with expensive equipment, which is needed for carrying out profitable work tasks (e.g. production line robots) or when the work is of a complex and safety critical nature (e.g. nuclear environments, explosive placement, surgery) has long proved problematical. A method used in some instances is that of immersive telepresence, but different approaches are needed for different sectors and equipment is often expensive and does not release high cost capital equipment for production line usage. In

all cases, the feedback of rich data from the physical hardware site to the operator site, via Internet or Mobile Communications links is bandwidth limited and contains uncertain delays.

It must be understood that the robot executes a predefined surveillance operation under a given fuzzy logic set. The user can interrupt the normal functioning by imposing their own commands on the robot via internet after providing necessary bandwidth solution. The robot returns live video feeds in terms of packets via the same channel. The Live video mechanism is also used in grid scanning algorithm to provide extra accurate movement controls to the robot apart from the regular sensors [1].

2. REQUIREMENTS AND OUTLINE OF OUR APPROACH

The first requirement is to do with delays in communications over the Internet. Our robot teleoperation system provides an asynchronous operation method to reduce the number of communication over the Internet. Undoubtedly Internet provides the most acceptable solution; the medium is cheap and readily available over most parts of the world. To execute our approach, we substitute a virtual space for real data from the robot. The virtual space is a reproduction of the place where the robot is. The operator generates a sequence of instructions by using the virtual reproduction, and then sends the sequence to the robot rather than each operation command. We term this data packaging.

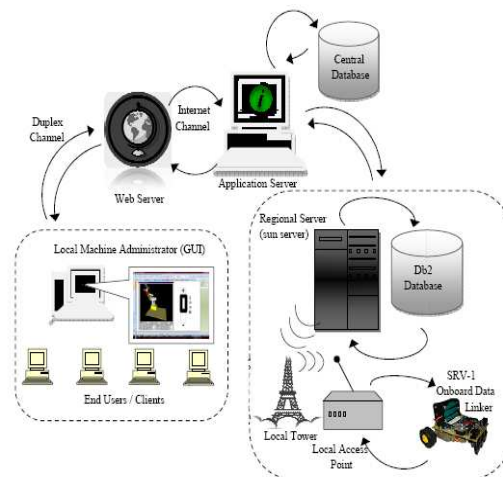


Figure 1. The basic module of the teleoperated robot over the internet

With asynchronous robot teleoperation, the operator has to estimate robot's movement before the robot actually works. But it is difficult since the movement would probably include some errors in real world. Secondly, there are many crackers and pranksters on the Internet. They might do damage to the robot or the objects in the place where the robot is;

Thirdly, the system should give good quality images from the video camera if the robot teleoperation system is used for an Internet museum. However, this conflicts with the first requirement since we have to reduce the number of transferred images and/or the size of the images to minimize the delays in communications. A solution for quality of images is making a movie file. We can also maintain a satisfactory quality of the images irrespective of the Internet's condition since the images need not to be streamed on the fly.

The fourth point is availability. It is impossible to provide as many robots as would-be users. Our solution for availability is based on indirect communication.

Finally, we must take account of interoperability. Diverse computer platforms are connected with the Internet. A robot teleoperation system for the Internet should therefore support multiplatform-interoperability.

3. BASIC MODULES

We have classified the entire system into four basic modules for the ease of understanding:

- The web based user interface which shall accept the task commands from the user and return projected outputs from the robot via the Solaris Workstation as the web linker.
- The link between the regional web server and the robot using access nodes in the specified region.
- The SRV-1 Surveillance robot which works under the given web commands and also independently of its own algorithm governed by the fuzzy rules.
- The live video feed mechanism with the advanced delay reduction technique.

First of all, it must be understood that the robot is well capable of performing its own surveillance operations according to a sequence of fuzzy rules in absence of any auxiliary control commands. This capability is triggered through an onboard controller with a preprogrammed algorithm consisting of a fuzzy set of logic. This will be described in detail later in the paper [3].

User could either use high definition hardware tools like hi-def multifunction joy stick to drive the robot or he could use highly developed graphics user interface (GUI) to do the same movements of joystick or toolbar application in GUI will trigger a set of commands which are used to control the robot. These commands will be directed from client machine to the desired regional server as a java applet; from there it will be redirected to our central server and an instruction code corresponding to these set of commands will be stored in our central database. These instructions will be redirected from our central server to regional server near to our host machine. Each host machine will be having one stand alone high-end server completely dedicated to that host machine where all the processing will be done and data received from and sent to the host machine will be stored.

Besides storing instruction code in central database we will also store these instruction codes in regional database server.

Finally these set of commands will be sent to end machine/host machine where they are run and action is performed. Host machine will be a standalone machine which is capable of storing these instruction commands and running them along with other important features like collecting the data of near and far objects in the forms of images, live videos, chemical environment, surrounding environment info like humidity, temperature, dryness life sign, light availability and so on [2].

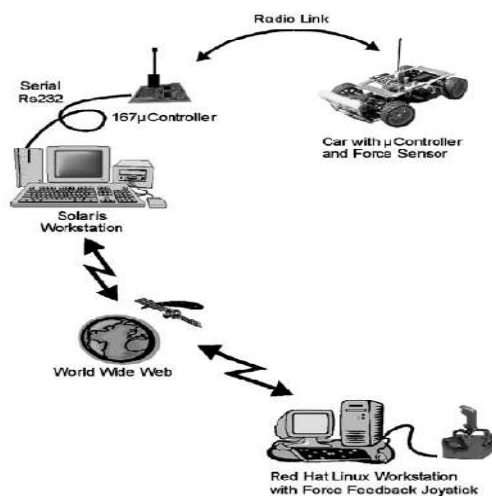


Figure 2. Motion Control of the SRV-1 using web commands

4. THE ASYNCHRONOUS COMMUNICATION

We have applied asynchronous communication and mobile agent computing to meet the above requirements for robot teleoperation. Figure 1 illustrates the basic scheme of an asynchronous system for robot teleoperation, built around mobile agents.

Once the required TCP/IP or UDP network connection has been set to a robot server, a mobile agent of the robot is moved to the user's computer. The mobile agent carries a virtual reproduction of the place where the robot is. The user gives control instructions to the robot in the virtual space.

Control instructions from the user are stored via the application server to the regional platform. The mobile agents which have a stored record of the requested operations are moved to the robot server where it can operate the robot via the Internet. This form of storage is primarily done on two level database systems which provide access to itself to the user under the permission of central administrator.

After the mobile agent has moved to the robot server, the agent acquires the current condition of the robot, and issues control instructions to the robot to enact user's instructions.

As a result, images are obtained and stored in the server. After teleoperation of the robot according to the user's control instructions has been completed, the images acquired are transmitted to the user's computer. A client application then displays the images on the user's screen.

4.1. Software requirements:

1. Db2 for database development.
2. DSP for server side programming
3. An Internet RMI interface
4. 2D graphical transformation map is visualized in GUI.

4.2. Hardware requirement:

1. The integrated robotic parts of the SRV-1 Surveillance robot including a chassis, a set of Sonar and IR sensors, actuators and the controller board.
2. A mounted video camera.
3. An embedded wireless data linker.
4. Server dedicated to one host machine.

5. THE SRV-1 SURVEILLANCE ROBOT

5.1. The Chassis

The chassis has to accommodate a microcontroller board, sensors, batteries and the radio link. An extra accommodation is made for a video camera. The chassis for an experimental robot car, based on a differential drive system and four wheels, was selected as the most versatile mobility platform. The wheels, made of hardened rubber have a diameter of 9 cm, providing sufficient clearance to handle typical terrain roughness. The chassis is made of ABS-fibre plastic, being lightweight, hard to break, and very stiff. Thus it can carry a payload of more than 2 kg. The stretch in front of the suspension can be varied, or different springs can be built into the suspension of the car, allowing even more load to be carried. The major mass contribution is from the Li-ion batteries which have a long life in order to allow driving sequences in the order of hours.

5.2. Electrical Architecture and the Sensor Set

The onboard controller segment carries an ATmega32L microcontroller development board with both serial and parallel interface, supporting interfaces to a broad spectrum of sensor configurations. The microcontroller also provides a pulse width modulated output, which is used for the drive control and steering. The sensors on-board the rovers have to provide the information to the teleoperator to characterize the rover's working environment. The standard sensor system includes one Sonar sensor mounted on the front end of the bot. Directly mounted below it and separated by a shielding diaphragm to avoid sensor interferences, is an IR sensor emitter-detector pair with an optimal distance calibration. The controller also contains a comparator IC such as LM324 and a motor driver IC such as L293D [4].

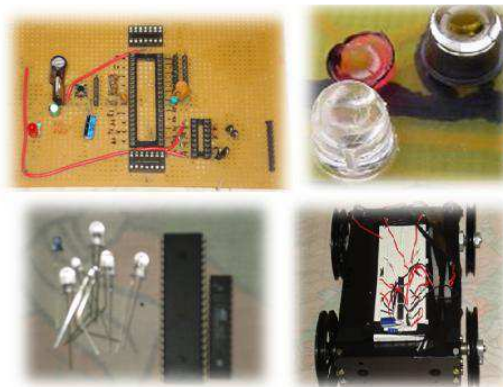


Figure 3. Hardware constituents of the SRV-1 Module

5.3. The Mounted Telecamera Portal

The RAM/PRO – 843 camera module was selected.

5.4. The Radio Link

For teleoperated mobile robots the telemetry and telecommand data are to be transferred in a short distance link to a nearby fixed workstation, which then provides the access to long-distance telecommunication systems. As this short-distance link is critical for the overall performance, several alternative link implementations on basis of Bluetooth, DECT, and wireless LAN have already been analyzed, before a radio package controller (RPC) was selected for this task.

This RPC is an intelligent transceiver module, which enables a radio network/link to be implemented between digital devices. The selected Radiometrix RPC-433-A is a low-energy-demanding device. It combines a UHF radio transceiver and a 40 Kbit/s packet transfer. A reliable range indoors is around 30m and outdoors about 120 m. The module provides all the RF circuits and processor functions for intensive low-level packet formatting and packet recovery, required to interconnect several micro-controllers in a radio network.

5.5. Remote control of mobile robots including haptic interfaces

The remote control of mini-rovers via the Internet is determined by the network performance composed of

- The radio link between the rover and the local PC and
- The Internet link between the local server computer and the client computer.

Crucial issues in that context are related to

- Bandwidth for transferring sensor data from the rover to the teleoperator and
- Latency in passing the control commands from the teleoperator to the bot.

5.6. The teleoperator workstation

We may either use a joystick or an advanced GUI for the main input portal for the teleoperator commands in the telematic system configuration. It is connected by a PCI interface board to the client's computer. With PC/104 we have a form factor small enough to be accommodated within the overall physical requirements of a small mobile robot: a standard PC/104 card measures 90 mm × 95 mm: More significantly PC/104 provides us with a micro-controller based upon the Intel 386, 486 or Pentium processor within an *IBM Personal Computer* compatible architecture. Clearly a card measuring about 14in² cannot accommodate a complete PC and so the PC/104 card set provides a modular architecture in which major functional blocks are all implemented as different PC/104 cards, all interconnected via the PC/104 bus. This allows a high data transfer rate between the programs on the client computer utilizing typically a Red Hat Linux operating system. In order to provide fast and easy access to the board, an interface in the object-oriented programming languages of the JAVA platform was implemented because of the common interface shared all over the world.

5.7. The data link between client/server

Data transfer between the client and the server computer over the Internet has been realized by a communication based on sockets, using the Internet user datagram protocol (UDP). In contrast to the TCP/IP Protocol, used as Internet standard, no acknowledgement of received data packages is provided. The internet bandwidth selected must support a duplex transfer channel for the to and fro channelling of the data from the end user and the server with respect to web commands and video feeds. Thus in case of lost packages also no retransmission is initiated. The data transfer reaches with UDP better real-time performance at the cost of a potentially undetected loss of data.

5.8. System Software Design

Two threads were created in the main function for transferring and receiving data packets separately over the Internet. On either workstation the same methods are used to send and receive data packets. In this application, platform independence is not realizable, because the joystick as well as the robot is a special device, which can only be used with dedicated hardware (the appropriate PCI board, respectively, the SRV-1 vehicle). The added feature in this function is that it can directly trigger the existing fuzzy rule set and can manipulate the microcontroller input through auxiliary rule set. Thus the end user can interfere in the normal motion algorithm and reprogram the fuzzy logic set to redefine the input and output variables.

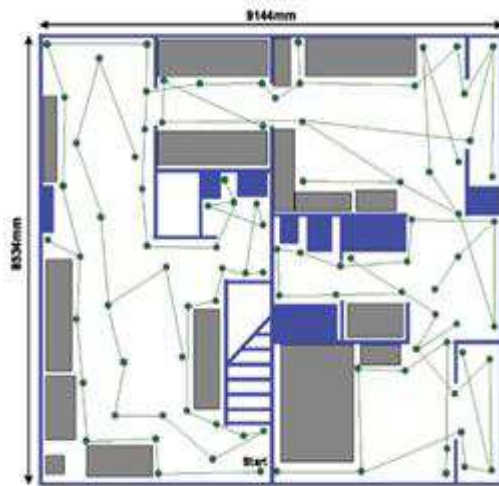


Figure 4. Map of Environment with node location



Figure 5. Configuring IP address for accessing the appropriate web server

5.9. The Server Computer

On the server (in this case a workstation with Solaris operating system) the wireless link to the SRV-1 which is a RS232 serial interface specifically designed to decode the host link, is installed. An ATmega32L microcontroller placed in serial connection pre-processes the data received by the radio modem and writes them to a RS232 interface (serial port) of the server computer. The maximum data transfer rate obtained on experimental ambience is around 7 Kbit/s.

A cycle to transfer one data package from the server to the car and back takes approximately 37ms. Typical delay periods measured for the Internet data transfer to locations on different continents ranged between 70 and 120ms.

5.10. The Robot Server software

We have successfully ported a full Linux implementation including all networking components, a C compiler and its libraries, onto the mobile robot hardware described above, comprising an Intel 386SX CPU card running at 25 MHz, with 4 MB RAM and an 80 MB solid-state disk drive. Linux also lends itself to optimisation. The Linux kernel may, for instance, be re-compiled to omit the drivers or services not required in a particular implementation.

5.11. The SRV-1 robot on-board data processing

All sensors and actuators on the car are connected to the analog and digital input and output ports of the ATmega32L micro-controller. The C programming language provides sufficient real-time performance.

During those instances when the onboard controller fails to establish a communication link to the server or there is a discontinuity of the auxiliary commands from the user's end, the controller switches to a self toggle mode and uses the predefined fuzzy rules to continue the surveillance operations.

The X and Y coordinate values from the joystick can be transmitted directly to the server and these inputs are then transformed into control commands for the drive and the steering motor. They are assigned to the pulse width modulated output of the micro-controller, used to control the drive and steering angle.

6. FUZZY RULES

The fuzzy rules are based in the common sense, as the following sentence: If the sensor is far and distance is far then the speed is fast. The table 1 resumes the fuzzy rules. The data got from the sonar sensors applies only to the control speed, not for the obstacle avoidance. The collision monitoring is mainly realised by the infrared sensor that monitors at any time the received signal from the near obstacles. If a very near object is detected by the IR sensor the robot is instructed to stop. For practical purposes, the number of input ports of the system was limited to two i.e. inputs from the respective sensors *Distance sensor (Sonar Sensor)* and *Obstacle sensor (IR Sensor)*

The logic was modelled using the MATLAB 7.1 Simulink software. According to the logic, five linguistic fuzzy logic sets were selected and the obstacle categorizes them into five optimal length ranges. Thus the variables are:

Table 1. IR sensor output response of the input Membership function through MATLAB 7.1 Simulink.

Variables	IR Range sensor cm	Distance in cm
VN	15	200
N	30	400
M	45	600
F	60	800
VF	75	1000

Accordingly, the output variables to be randomly set to the controller in run time were also designated five linguistic fuzzy variables. These variables are:

Table 2. Speed output response of the input Membership function through MATLAB 7.1 Simulink

Variables	Speed in cm/s
VS	8
S	16
M	24
F	32
VF	40

The details of the range of membership function for the input and output variables are shown below.

Table 3. The 25 rules of the fuzzy logic sets

Rule No.	IR sensor	Distance	Speed
1.	VN	VN	VS
2.	VN	N	VS
3.	VN	M	VS
4.	VN	F	VS
5.	VN	VF	VS
6.	N	VN	VS
7.	N	N	S
8.	N	M	S
9.	N	F	S
10.	N	VF	S
11.	M	VN	VS
12.	M	N	S
13.	M	M	M
14.	M	F	M
15.	M	VF	M
16.	F	VN	VS
17.	F	N	S
18.	F	M	M
19.	F	F	F
20.	F	VF	F
21.	VF	VN	VS
22.	VF	N	S
23.	VF	M	M
24.	VF	F	F
25.	VF	VF	VF

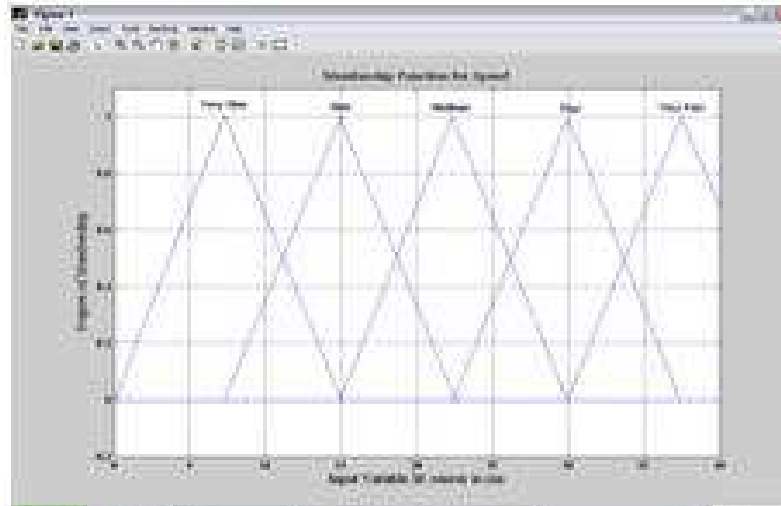


Figure 6. The output map of the sensor response with respect to speed

7. LIVE VIDEO FEED

Live Video Feed (LVD) can serve end users with the possibility of viewing the ambience of the robot surrounding and hence perform the surveillance operation. The mounted camera gathers the images and displays them. Its main job in the robot server is the capturing of images from the video camera. The camera agent also studies user demand, the condition of the network, and user's computer power, then converts, compresses, edits the captured images accordingly if necessary. The camera agent is moved to the user's computer and displays the images after the robot agent has finished all of its programmed operations.

We used VRML for the museum agent because of its popularity for 3D computer graphics and the plenitude of available viewers and tools. Most parts of the system are written in the C programming language for speed, but the user interface and processing module for navigation in virtual space were written in Java, because Java has useful libraries for handling VRML.

Video transmission is propagated in the same manner as the command transmission was done i.e. via the same channel, however the existing bandwidth provided by the UHF frequency cannot handle the bit rate required. Many researchers have made tremendous efforts to reduce the server bandwidth required. Previous works have shown that periodic broadcasting schemes can significantly improve the efficiency of LVD by periodically transmitting video segments in a proactive way and guarantee service latency for a certain period of time. The method that we have selected is termed Harmonic Broadcasting [5].

7.1. Harmonic Broadcasting

This scheme divides a video into n segments of n segments of equal size, S_1, S_2, \dots, S_n each of which is distributed over n channels, C_1, C_2, \dots, C_n . The bandwidth of these channels will be equal to the video consumption rate b and so the total bandwidth consumed by the scheme will be nb . A time slot is defined as a duration in which a segment is consumed or viewed at the consumption rate which is equal to $d = D/n$, or equivalently to maximise n .

As we know, each segment S_i would be broadcasted with the periodicity of $i \times d$ at most, to keep playing out continuously for each user. The first channel transmits segment S_1 at frequency $1/d$ i.e. it repeats the segment continuously. If each segment S_i is transmitted with a periodicity of $i \times d$, the scheme can acquire the best performance. It can be noted that the segment S_5 is transmitted at frequency $1/4d$ which wastes some bandwidth.

7.2. Grid Scanning Algorithm

In spite of using the technologies of Asynchronous Communication and Harmonic Broadcasting there may still be a spot for a margin of error. Consider the surveillance bot to be present at coordinates at (x, y) at time $t=0$ and the telecommand sent to it at that time instant instructs it to traverse through a specific path to coordinates at (x_1, y_1) . But due to the existent factors of time delay due to network congestion, the telecommands reach the bot after a lag of a small time $t=t_1$. During this time the bot, moving under its own predefined fuzzy rules travels to some random coordinate (x', y') . After receiving the packet of instruction at

time $t=t_1$, the bot follows the instructed path at ends up somewhere at coordinates (x_1', y_1') whose deviation from the actual destination (x_1, y_1) is proportional to the deviation of originating point of coordinates.



Figure 7(a). A camera's image of the equivalent real view

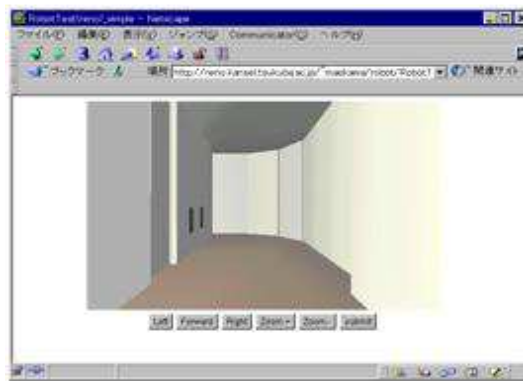


Figure 7(b). A view of a virtual space for the system

To prevent this error, we have devised a new algorithm called the Grid Scanning Algorithm which utilizes a robust mechanism to make the bot controller to keep a track of its traversed path. The path of the robot is defined by a set of coordinates of the reference points which must be known. This is done by mapping the entire area under surveillance, dividing it into an optimum number of grids recognizable by the onboard camera and storing it in the memory. Ranger finder data is collected in the robot's frame of reference by scanning a space around the robot's instant position and storing in a structured file.

When the robot recognizes the incoming telecommand, it pauses for the moment, estimates the current coordinates, analyses the path needed to be taken to the required destination point and moves to the new position and the process returns to the normal mechanism.

In case the onboard hardware doesn't support sufficient memory the scanned coordinates can be stored by the local regional server as the time taken for the data transfer between the bot and the server is considered almost negligible. In this case, the server analyses the path taken by the bot to the required destination. In a randomly changing environment the tracking of coordinates could be done by using the polar coordinate method to designate each point by using a central RFID tower to monitor the random position of the bot.

8. CURRENT PROSPECTS OF VARIOUS TELEROBOTIC APPLICATIONS

8.1. Space

The primary current example of a space teleoperator capability is the 20 m long remote manipulator system (RMS) built by the Canadian firm SPAR and carried aboard the U.S. space shuttle. It has six DOF and is controlled directly by a human operator viewing through a window or over video and using two three-axis Variable-rate-command joysticks, one for three translations, and one for three rotations.

8.2. Undersea

Undersea teleoperator, in that community called "remotely operated vehicles" (ROVs), had come into extensive use by the offshore oil and gas industry in well-head completion operations, monitoring of pipelines, placing of anodes and inspection of welds on subsea structures, and other tasks.

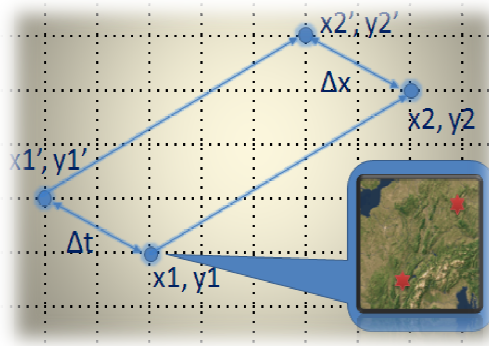


Figure 8. The plotting of the bot environment and analysis of path traversal

8.3. Nuclear power

The nuclear "hot laboratories" continue as the application most wedded to force-reflecting master-slave systems which use pulleys rather than gears to reduce motor speed and increase torque but yet make the manipulator back-drivable. Such teleoperators, however, are now becoming sufficiently mobile and dexterous for doing

8.4. Aids for the handicapped

An alternative to arm prosthesis, i.e. a device actually worn on the body, is a teleoperator separate from the body, attached to a wheelchair, or able to wheel itself around the room.

8.5. Surgery

One form of teleoperator is the endoscope, a coherent fibre-optic bundle with added tubes for conveying fluids to or from the distal end point, plus wires for modifying its curvature so it can be manoeuvred around corners, plus additional wires operating an end-point gripper or snare or cauterize. Such devices when inserted into the gastrointestinal tract (from either end) have proven very successful for inspection, biopsy and simple surgery.

8.6. Terrestrial mining

Mining is the sensing of where coal seams are and the decision of where to cut; radio or sound waves are occluded. Assuming operating energy can be stored on board the teleoperator, this could be a good application for disposable optical fiber line unreeled and laid down as the vehicle moves into the mine.

8.7. Fire fighting

This form of teleoperation can serve a great purpose, even if it merely performed the task of extending a fire fighter's vision and hearing to within a hazardous fire situation.

8.8. Military operations

The utilization of telerobotics can be extended upto inspection and deactivation of mines and other explosives; observation of enemy operations without exposure to enemy fire; remotely piloted aircraft.

9. CONCLUSION

This paper has described application of internet to the teleoperation of a robot. An algorithm for the construction and visualization a map of the environment is developed based on information acquired by a telerobot range finder. Users are involved in finding the optimal trajectory in that environment. Remote Path Planning of a different kind of robots is possible with specific hardware by binding them to a Network connection through the server. The proposed fuzzy proxy system ensures the smooth running of the

interactive process on the client side. The reflexive module of the robot ensures its autonomous moving in situations.

Many stages and varieties of teleoperator control, from direct manual to sophisticated Telerobotics should emerge. Indeed, if one considers all the machines which may be programmed by human operators to sense and manipulate objects one has to include airplanes, ships and industrial systems of many kinds, systems that we would not ordinarily call manipulators. Yet the current interest in telerobotics has encouraged us to think hard about the relation of man to machine in doing useful work.

REFERENCES

- [1] R. Dawkins, O. Holland, A. Winfield, P. Greenway, A. Stephens, An interacting multi-robot system and smart environment for studying collective behaviours.
- [2] P.T. Davies, C.R. McGuffin, Wireless Local Area Networks, McGraw-Hill, New York, 1995.
- [3] P. Wong, D. Britland, Mobile Data Communications Systems, Artech House Publishers, 1995.
- [4] K. Dobrovodsky, P. Andris and P. Kurdel, "A Virtual Reality Robot Workcell Simulator", *Proceedings of 9th Int. Workshop Robotics in Alpe-Adria-DanubeRegion RAAD 2000*, pp.331-336, 2000.
- [5] C. P. Pfleeger, *Security in Computing*, Prentice-Hall, second edition, 1997. [1] G. Mair, "Transparent Telepresence Research", *Industrial Robot*, Vol. 26, No 3, pp. 209-215, 1999

BIOGRAPHIES OF AUTHORS



Ashesh Vasalya has completed his B.Tech degree in Electronics and Instrumentation Engineering from Vellore Institute of Technology University (VIT-U), Vellore, Tamil Nadu, India, 2008-12.

He has worked for Mini Baja SAE India 2011 and has been completed his academic internship on the Implementation of (Media Oriented System Transport) MOST25 Protocol at TIFAC-CORE in automotive Infotronics, Department of Science and Technology, Govt. of India. He is the board member of ISOI (Instrument Society of India) VIT-U branch and the member of SAE (Society of Automotive Engineers).

He is also the IEEE author of a paper on Robotics and Int. Journal author of the papers on Wireless Communication and Telecommunication. He has done his final project thesis on 'OFDMA/CDMA Based Systems Analysis and their Contribution New Access Technique'. He is currently placed in HCL as Software Engineer. His areas of interests are Advanced Autonomous Robotics, Advanced Embedded Device Driver, wireless communication Networks and Automotive Communication Protocols etc.



Rohit Agrawal is currently working as a Process Technology Engineer in Magneti Marelli Powertrain India Ltd. in the Manufacturing Department involved mainly in the automatic optical soldering inspection process optimization. He completed his engineering graduation from VIT University, Vellore in the year 2011 in Electronics and Communication Department. He did an internship project on Power Optimization in Numeric LCD's at India Institute of Technology, Bombay. He carried out his final year project in Delphi Automotive – Technical Center India on Automotive Diagnostics. He is the author of four papers, 1 Springer, 2 IEEE and one paper in the International Journal of Distributed and Parallel Computing illustrating his various project works. He has also completed an incubated project under Technopreneurship Promotion Programme (TePP) by Department of Science and Technology applied for a patent. His areas of interest include Embedded Systems Design, In-vehicle Networking and Wireless Communication.