KhepOnTheWeb: One Year of Access to a Mobile Robot on the Internet

Patrick Saucy and Francesco Mondada
Microprocessor and Interface Lab (LAMI)
Swiss Federal Institute of Technology
Lausanne, Switzerland
saucy or mondada@epfl.ch
http://diwww.epfl.ch/lami/team/saucy or mondada

Abstract

Working in the field of mobile robotics, we are interested in sharing robots with other people via a network. In the 3 last years, there was already some work done mainly with robot arms but rarely with a mobile robot. In spring 1997, a first attempt was done with KhepOnTheWeb. Its purpose is only to demonstrate some possibilities of remote control of a Khepera mobile robot. It is accessible daily to everybody. After one year of access we performed an analysis of the log-files in order to understand the behaviour of the public in front of such an installation. This analysis was rather difficult because of the large amount of data involved. Specific software has been developed in order to extract the relevant information and to put it in a form that can be significative. Results show that people come mainly once and the few who return do it rapidly. The second visit is made on average after less than 15 days. Only a few of the visitors really operated the setup.

During this year, another benchmark (TeleRoboLab) was built. It is reserved only to researchers and will be available at the end of 1998. It takes into account the results presented in this paper but adds a scientific dimension to telerobotics.

This paper focuses on interface design, hardware and particularly on user behaviour. The robot is still accessible at: **http://KhepOnTheWeb.epfl.ch/**.

1 Introduction

In the research community we have been used for years to remotely access, via standard communication networks, unique or expensive structures, such as supercomputers, important databases, unique information resources or the World Wide Web (shortly, the WEB), the email box or the ftp servers full of software. With the growing of the Internet you find more and more devices such as coffee machines, cameras, telescopes, manipulators or mobile robots connected to it. Despite the fact that you can spy on other people with hundreds of cameras you can interact only with a few robots, and they often have a restricted access [1].

Why ? There are several explanations. An installation with a robot is very expensive. You have to do regular maintenance. Things are much easier with a camera. The interaction with the user is very poor. He just sits and watches. Sometimes he has the possibility to choose a different orientation of the camera [7]. With a robot you have a strong interaction. For instance, you can move on the floor and grasp objects with a mobile robot equipped with an arm [8]. Discovering the control interface, the user has to understand rapidly the goal of your site and what are the possibilities of the robot in order to achieve them. A very famous example is the "Mercury project" [2]. This kind of experiment is very useful in the sense that it gives important information about the reactions of the users, the kind of equipment needed and the constraints of Internet. More information about devices connected to the net can be found at [9], [10].

2 Project Goal

The goal of our project[1] is very similar to those of the experiments described in the introduction, namely to pro-

1. This project, called "Sharing of Unique or Expensive Equipment for Research and Education : The Remote Manipulation Paradigm", is part of the Swiss Priority Program SPP ICS of the Swiss National Research Foundation.

vide the access, through network communication facilities, to a complex and unique mobile robotics setup. The specificity of our project is that we want to provide a setup mainly to the scientific community carrying out research in mobile robotics control.

The main motivations of our project are both economic and scientific. On one side, this kind of equipment is very expensive and could be shared between several research laboratories to reduce the expenses. On the other side, the comparison between different approaches in the field of mobile robotics can be made only on robots placed in perfectly identical conditions . This does not only means that the robot has to be the same, but also that the environment has to be identical, in every detail like lighting conditions or colour and type of the floor.

It is therefore scientifically a great advantage to share a common environment for the understanding of the actual control approaches.

The goal and the motivation of the project fits well with the activity of our lab. The LAMI (Microprocessor and Interface Lab) is specialised in the development of mobile robots and tools for the research in control algorithms. The remote experimentation is the logic extension of this activity.

In the first part of this project, our goal was to understand the possibilities available with the current technology, the reactions of users and the requirements for the equipment of such a setup. In comparison with other projects on the WEB, described in the introduction, our installation has some additional features :

- The controlled device is a mobile robot equipped with an on-board camera. This is also the case of some other setups that are partially available [4].
- Unlike other mobile robot setups, ours is running daily without the need of an external support.
- The interface has a live video feedback and runs in a standard WEB environment.
- Everybody has access to our robot. There is no distinction such as registered user/guest. But only one person at a time can control the robot for a maximum of 5 minutes.

3 System Architecture

Our setup consists of a mobile robot accessible via the Internet and moving in a wooden maze, see figure 1. The labyrinth is made such a way that the visitor can see various effects. In a general way, the walls are higher than the robot, so that the visitor has to move around to explore the maze. Only some centre walls are lower than the others. On the right there is a mirror allowing the user to watch the robot he is controlling. On the bottom left of figure 1 there is a ramp giving the robot access to a view outside the maze.

In this experiment we use the mobile robot Khepera developed in our lab in collaboration with K-Team, the company producing and selling it. Khepera is equipped with an on-board CCD camera and is connected to a PC via a wired link. The camera sends the video signal to a framegrabber also placed in the same PC. This computer is also our WEB server and is therefore connected to Internet. The user can access this installation via the WEB browser Netscape. Internet Explorer^{TM} is not yet supported.

A virtual model of the setup has also been developed using VRML and Java. By this way the user has the possibility to train himself by running a simulation locally on his machine and without time delays constraints. A complete description of this implementation can be found in [5].

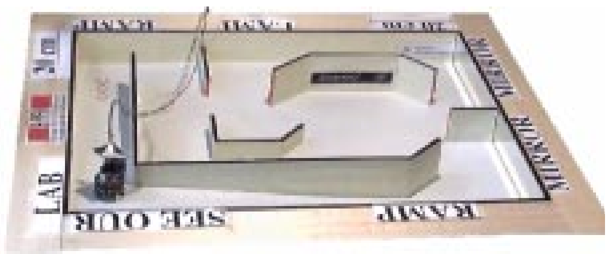

Figure 1: Khepera with its on-board video camera watching out of the maze, whose size is 65 X 90 cm.

3.1 The mobile robot Khepera

Khepera is a small cylindrical robot, 55 mm in diameter and of variable height. Its small size allows us to employ a suspended cable for power supply and other signals without disturbing its movements. A camera observing the environment can be placed without the need for wide-angle lenses . In its configuration of figure 1 Khepera is made up of three layers corresponding to :

- the sensory-motor board equipped with 8 infrared proximity sensors and two motors
- the CPU board equipped with a Motorola 68331 microcontroller
- the video board carrying a color CCD camera with 500 X 582 pixels

3.2 Hardware

The host computer communicates with the robot via a RS232 link at the speed of 38’400 bits/s. The video signal is sent from the robot to the framegrabber in the PC, in a differential mode on two additional wires.

An external camera (Canon VC-C1) is mounted on the ceiling above the maze in order to give the user a global view. This is an interesting help for planning a long displacement. A RS232 link between the PC and the camera allows the visitor to control the camera orientation and the zoom factor in order to have a better look of the current situation. The video is also wired to the framegrabber of the PC and multiplexed with the video signal coming from the robot camera. Therefore, it is possible to switch between the robot view and the external view. All these connections are summarized in figure 2.

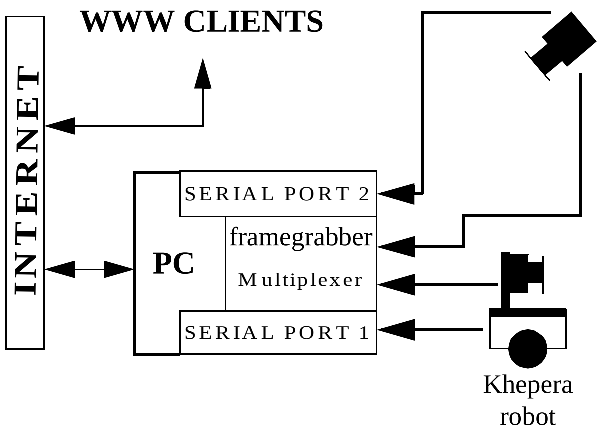

Figure 2: Connections between the devices of the setup

3.3 Software

The PC operating system is Windows 95™. The WEB server is the Personal Web Server™ from MicroSoft™. This server launches several CGI (Common Gateway Interface) scripts to perform the tasks. They communicate between them through shared named memory. Two programs run continuously :

- The first one grabs the images and puts them in shared memory in JPEG format (160 X 120).
- The second one puts the IP address of every new user in shared memory in such a way that others CGI scripts have the possibility to control the user identity. A timeout variable is regularly decremented in memory. After 5 minutes, if the user has not disconnected, this program puts the user IP address in a list forbidding him during 30 seconds to get control again.

When a client performs an access to the control page, the server starts a program that sends continuously the images stored in the memory. This image feedback is based on the server push technique supported by Netscape but not by Internet Explorer™ at the time of writing. Others techniques can be used but have not been tested in this experiment [6].

On the client site, the user has access to a control page made mainly in plain HTML and using clickable images. There are three types of commands available :

- Commands to control the robot movements (speed/position). The click coordinates are sent to the server, where a CGI script decodes and builds the corresponding order for the robot. The orders are sent to Khepera via the RS232 serial link.
- Commands to control the external camera movements (orientation/zoom). Here too, the orders are sent to the camera via the RS232 serial link.
- Commands to switch the camera. A CGI script on the server acts on the multiplexer present at the input of the framegrabber.

A Java applet running on the client side sends regularly some requests for information about the state of the robot and the time left to the user. A CGI script on the server answers these requests collecting the information from the robot and the shared memory.

There is no local “intelligence” on the robot such as obstacle avoidance. This type of mechanism is not necessary because of the light weight of the robot. This also means that there is no risk to destruct a wall or the robot itself. The advantage of having a direct control is that the user can see the result of his own action without any external contribution. The drawback is that the control of the robot is more difficult without help and under important delays.

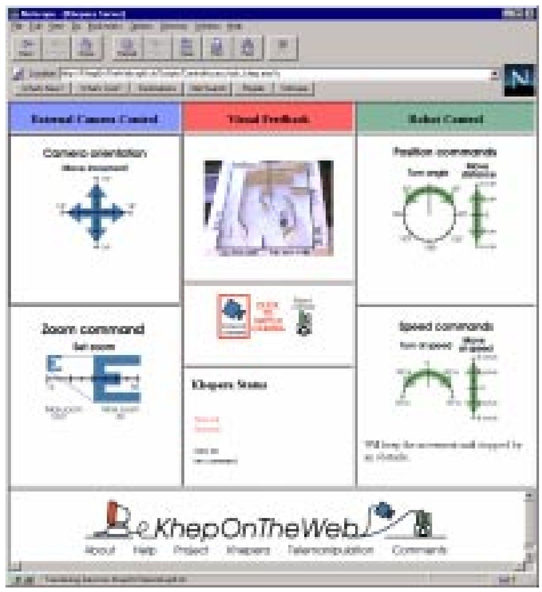

Figure 3: The remote control interface provided under Netscape

3.4 Interface

The interface at the level of the client includes all possible operations that can be made on the robot, the external camera and the multiplexer. The complete window available under Netscape is shown in figure 3.

The interface is composed of three columns, corresponding to the three types of commands that the user can perform :

- On the left column there are the controls of the orientation (pan and tilt) and of the zoom factor of the external camera. The values of the parameters are given by clicking on the graduations.
- On the middle column there are the visual video feedback, the switch control of the two cameras and the display of the robot status.
- On the right column, two different control panels allow to send orders to the robot. The upper one provides displacement commands. The lower one gives speed commands. Each of these control panels has two parts. One for the rotation and one for the straight movement of the robot.

This interface is not the first one we tried. We started in 1995 developing a control interface based on a videoconferencing software (CuSeeMe™). The results were interesting but the control software was running only on MacIntosh™, and therefore we moved to a pure WEB interface. The constraint is to send complete images to the client but the performances are still good. It is also possible to use animated GIF and to send only the part that changes in the images but a “caterpillar effect” [6] can be observed.

The image size is small. In order to give more comfort to the user it is possible to display the image larger than it is in the browser without loss of bandwidth.

Finally, we have spent more time on the design of the interface than in all other engineering parts.

4 Log Access Analysis

The period between May 1997 and May 1998 is considered for this analysis. As explained above, the access to the virtual setup is not taken into account. All our statistics are based on the IP addresses. Therefore several persons from the same site will appear as a single one. Moreover, some Internet addresses are allocated dynamically and the same person visiting on separate occasions may appear as another person. In this paragraph, IP addresses, users, machines and visitors have the same meaning. Next we define three terms : *action*, *session* and *nip*.

Action is a script run by the user to control the robot or the camera. A *forbidden action* is an action launched by a user who has not asked for the control of the robot from the presentation page. He is not registered by the system and his action is refused. A *0 action* is due to a visitor who does nothing after having loaded the control page. If he did not load the control page we do not count it in *0 action* but in *no control page*.

A *session* or visit is defined as an uninterrupted access by the same machine with a maximal break of 10 minutes between two accesses.

In *nip* are gathered IP addresses which the name could not be found in a DNS (Domain Name Server).

4.1 General Statistics

Based on theses definitions, we had 27’498 visits performed by 18’408 unique machines. Only 3’178 machines did more than one access. Their average return time is about 23 days with a typical delay between two actions of 13.6 seconds.

The most active session was issued from the United-Kingdom. The user performed 630 actions distributed as follows :

- 344 actions for the control of the robot
- 243 actions for the control of the camera
- 43 actions on the switch of the panoramic and the embedded camera

This particular visit had a duration of one hour and ten minutes. This corresponds to an average of 1 action every 7 seconds. This visitor came back 13 times but he never did so much actions again.

4.2 Actions Distribution

All other sessions were not as active as the one discussed above. Figure 4 depicts the usage of the site.

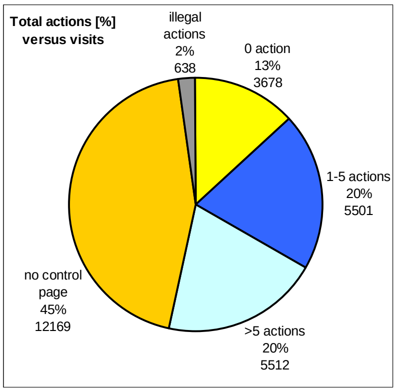

Figure 4: Distribution of actions versus visits

45% of users visited only the welcome page at our site. They did not request the control of the robot. There are two possible explanations for that :

- The browser type. Microsoft Internet Explorer™ does not allow visual feedback using the *server push* technique. The welcome page contains a warning about this problem. Statistics in [11] show that the distribution between the two main browser is : 50% Netscape, 45% Internet Explorer™ and 5% others.
- The type of access. The user has the possibility to control the real or the virtual robot. The welcome page of the virtual setup was loaded by 11'283 machines.

20% of the users had access to the control page but did not perform any actions. The reasons for that could be :

- The robot was already controlled by other people.
- The user could not understand the goal of the site.
- The image rate was too low (too big delays) or zero (browser type).
- There were problems while loading the clickable maps or the Java applet.

The previous graphic (figure 4) represents very well the actions distribution in general. We found approximately the same distribution in other analyses not illustrated here :

- action distribution versus months
- action distribution versus number of visits
- action distribution versus domain of the machines
- action distribution versus our own use (demonstrations and verifications)

Figure 5 is focussed on the effective use of the site, represented by the actions.

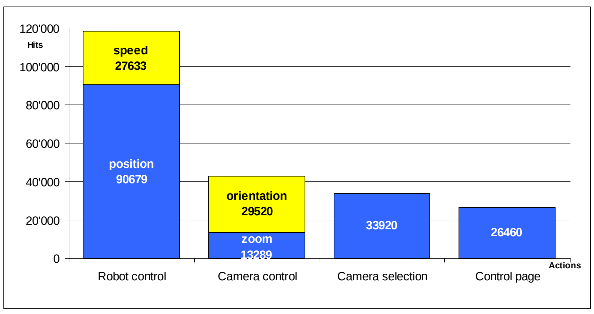

Figure 5: Actions to control the robot and the camera

The robot is mainly controlled in position. This emphasizes the "wait and see" strategy developed by the users. Although the goal of the site is to control the robot, the camera is strongly solicited. Generally, a camera view is requested every second robot action. The visitor needs different view points in order to correctly understand the robot's location before deciding the next move. The importance of the column "camera selection" shows also the necessity to have both a panoramic camera and a camera on the robot.

The relationship between delays on the network and the number of actions performed is emphasized by figure 6. Only domains with more than 100 users who performed more than 5 actions are taken into account. There is a clear relation between the time/action and the number of actions/user.

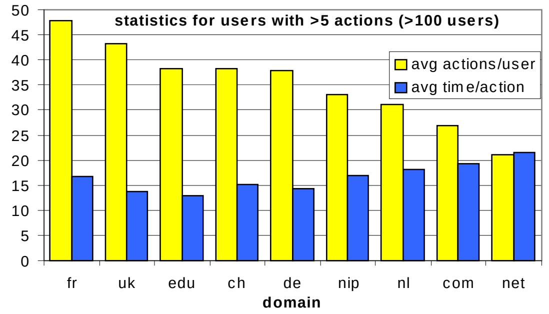

Figure 6: Relationship between delays and actions

Figure 7 depicts the influence of external causes on the evolution of the number of accesses along the 12 months considered.

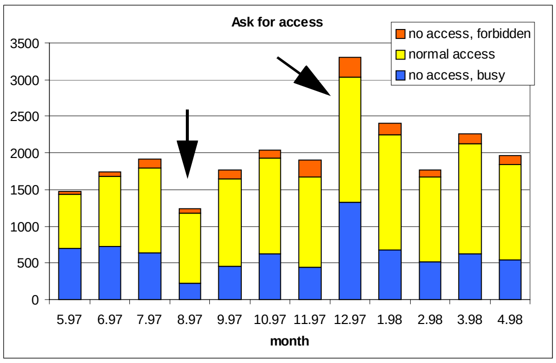

Figure 7: Accesses during 12 months

The two months pointed by the arrows are interesting. August 1997 shows a minimum due to an holiday period. August 1998 (not illustrated here) shows the same singularity. December shows a maximum. The 8th December, the site was selected "Cool Robot of the Week" by the Nasa [12]. Already on the following month, the number of accesses decreased and stabilized around 2'000 per month. People visit mainly once and do not return or come back fast a second and last time. We think that this behaviour is based on the "surf effect" of the WEB. This result is verified by the statistics of the Australian Telerobot [13].

4.3 Returning Visitors

Only 3'178 machines came back to our site. They made 12'268 of the 27'498 visits. Different analyses about the returns are shown in the next two graphics. Although return average is about 23 days, figure 8 exhibits an important peak of returns centred on 18 minutes.

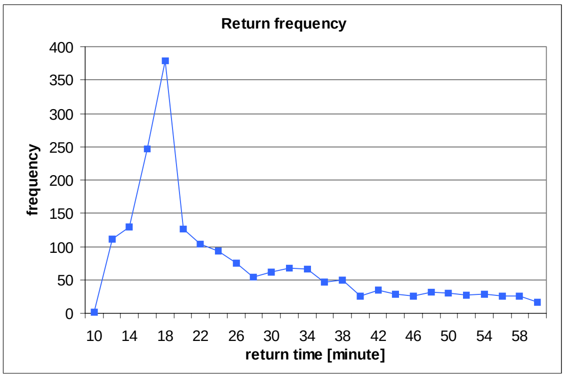

Figure 8: Return frequency

1'096 of a 9'090 total returns are made after 12 to 22 minutes. This confirms the "surf effect". This peak is mainly due to people who come a second time to our site because they could not control the robot the first time, see figure 7. So they retry later. There is no return below 10 minutes because two hits from the same machine made in an interval of 10 minutes belong to the same session by definition.

How do the returns influence the user behaviour ? Figure 9 tries to answer this question.

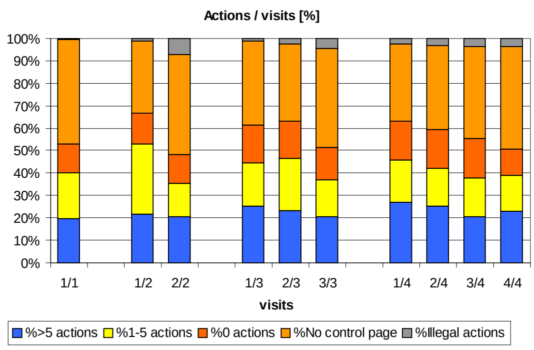

Figure 9: User behaviour versus visits number

This figure describes the distribution of the actions for each visit for 4 user categories performing a different total number of visits. The analysis is restricted to the first 4 visits in order to maintain a representative sample. Users who come back have acted during their first visit much more than the ones who did only one visit. We could expect that from their second visit users take advantage of their knowledge to do more actions. This is not the case. The number "of more than 5 actions" even decreases slightly. People who come back seem to only have a look at the presentation page. The reason could be that they want only to show the site to some other people, get information or simply verify that the server still exists. The number of illegal actions increases, as well. It is possible that between two visits users have left their browser open on the control page. In this case there is a timeout which makes the following action illegal. Most of the people come back after a short time, see figure 8.

4.4 Geographic Location Influence

All these figures are established on the accesses of 18'408 unique machines. This is less than the 60'000 hosts that visited in 6 months the Mercury Project site [3]. One of their graphics shows that 58% of the machines were located in North America and only 14% in Europe. In our case, this is the opposite, only 7% of the machines were located in North America and 37% in Europe[1]. It could be because Internet is less well developed in Europe than in North America. Another explanation is that our site runs differently. It sends continuous video feedback and only one person at a time can have the control on the robot and this until the timeout.

Switzerland and its neighbours (France, Germany, United-Kingdom) made most actions. The geographic proximity (reduced access time) favours the use of our site. In other words, our site is less attractive by its functionality for users far away because of unacceptable response time. The domains com, nip and net did most accesses to our site but they made fewer returns than Switzerland and its neighbours.

5 Conclusions

The experience accumulated in this first test shows some important aspects. Such a physical environment is technically feasible with commercial parts. Everything from the robot to the camera, from the server to the framegrabber, are very standard commercial products. But there are still problems. One is to introduce a complex setup in a WEB where the rule is more "click and wait" than "read and click". Another one is obviously the delay which prevents people from having a good interac-

1. The domains com, nip and net have no geographic meaning. 51% of the machines come from these domains. We do not know how the Mercury Project processes takes them into account.

tion and to take interest in the site. This is not the only reason why users do not come back. The site is frozen. There is nothing new to see in a second visit. Once you controlled it, there is no reason to come back. As said above, you have to catch the netsurfer's attention the first time he comes. It is interesting to observe that the behaviour of the users is generally independent of the country or the time the site is available, which shows that there is a global and stable Internet culture.

But the analysis of the Internet user behaviour is not simple. The graphics and the analysis of the previous paragraphs show well that a good understanding is difficult to get. Only few of our analyses are included in this article. Significative graphics are very difficult to calculate because of the huge quantity of data of the log-files. These were analysed with our own software due to the specificity of the analysis made. It took two days of non-stop processing to scan the log-files and to get the data in a form that could be handled to create a graphic. Even with such an amount of data there were still categories with not enough representative samples. When a graphic exhibits a feature, it is rather difficult to explain it with the general knowledge about the Internet. Moreover you have very little feedback from the users themselves. We received only 1 mail for 1000 accesses and it contained very few information.

The reliability of the setup is good but we had to face two main problems. As said above the PC server is running Windows 95™. This system is not stable enough to allow the machine to operate without being regularly reseted to prevent a crash. The Personal Web Server™ was not well adapted for our usage. Sometimes it froze in a given state. Then it is impossible to access the welcome page. This kind of test has been useful for a new project called *RobOnTheWeb* [14]. The project team has an additional reason to use Linux as operating system and Apache as server.

The other problem is rather mechanical. Sometimes the robot is prevented from moving by an edge of a wall. Khepera is very light and its two motors are powerful. Therefore, against a wall, it is possible that it rises a little and that one of its wheels hardly keeps contact with the floor. In the ramp, with the weight of the cables or due to the dust, wheels can have less adhesion making difficult to control the robot.

6 Further work

As described in paragraph 2, a new site has been designed and implemented. It provides the complete remote control of a much bigger robot. The user will be able to download C code on the robot . Therefore its access has to be restricted to only scientific users who have been registered. The setup can be more complex because the users are familiar with computers and robotics but there is still a need for a good interface in order to let them concentrate on the algorithm and not to understand the site functionalities.

Currently, there is no tool available to face the delays on the Internet (in the future RSVP ?) but it is widely accessible. The idea of our site is to be still accessible from Internet but to use another network (ISDN) to control in real time the robot. For people who do not have ISDN, the robot can be controlled via the Internet. ISDN has poorer performance than other networks like ATM but it is less expensive and easier to install.

Acknowledgements

We would like to thank Edo Franzi and André Guignard for the important work in the design of Khepera. We thank very much also Prof. Jean-Daniel Nicoud and all the LAMI assistants who have tested our interfaces for their useful comments.

References

[1] R. Simmons. Where in the world is Xavier, the robot ? *Robotics and Machine Perception, 5,* issue 1:5-9, 1996. Special Issue : Networked Robotics.

[2] Golberg K. A telerobotic garden on the world wide web. *Robotics and Machine Perception,* 5, issue 1:4, 1996. Special issue : Networking Robotics.

[3] Goldberg K. Desktop Teleoperation via the World Wide Web, *IEEE International Conference on Robotics and Automation,* p. 654-658, 1995.

[4] J.A. Fryer. Remote-control experiment using a networked robot. *Robotics and Machine Perception,* 5, issue 1:12, 1996. Special Issue : Networked Robotics.

[5] O. Michel, P. Saucy, F. Mondada. "KhepOnTheWeb": An Experimental Demonstrator in Telerobotic and Virtual Reality. *Proceedings of the International Conference on Virtual Systems and Multimedia (VSMM'97),* IEEE Computer Society Press, 1997

[6] http://rr-vs.informatik.uni-ulm.de/rr/

[7] http://graco.unb.br/robwebcam.html

[8] http://freedom.artm.ulst.ac.uk/~antonh/research/Lynxmotion.html

[9] http://www.yahoo.com/Computers_and_Internet/Internet/Entertainment/Interesting_Devices_Connected_to_the_Net/Robots/

[10]http://queue.ieor.berkeley.edu/~goldberg/art/telerobotics-links.html

[11]http://www.cen.uiuc.edu/bstats/latest.html

[12]http://ranier.oact.hq.nasa.gov/telerobotics_page/coolrobots97.html

[13]http://telerobot.mech.uwa.edu.au/

[14]http://diwww.epfl.ch/lami/team/michel/RobOnWeb/