



AC13 Controller: Extendable Telepresence Robotics for Education and Beyond

Victor O. Santos Uceta¹, Dr. Elio Lozano Inca²

PhD. Student, Dept. of Computer Science, Purdue University, West Lafayette, Indiana, United States²

Associate Professor, Dept. of Computer Science, University of Puerto Rico, Bayamón, Puerto Rico¹

ABSTRACT: In recent years, mobile technology has increased exponentially. This paper focuses on the design and implementation of an application controller interface for a Rover Spy Tank using Java for the Android platform. Communication between controller and rover is via ad-hoc wireless. It uses the RoverOpen library to provide command parsing and execution to control the rover movements and video streaming. The application uses the OpenCV library to provide basic computer vision algorithms. An open source application, called AC13 Controller, was developed. It is composed of different packages that provide communication with the rover, control its movements, perform basic computer vision, store and crop images, and share data. It is characterized by its simplicity, effectivity, reachability, accessibility, and usability. Finally, we give a number of ideas on how to extend the application to add new functionalities.

KEYWORDS: Rover Robotics, Software Engineering, Telepresence, Machine Learning, AC13 Controller

I. INTRODUCTION

A. Android Platform

Android is a powerful mobile computer platform and it has become popular on smartphones and tablets used worldwide. It has also become popular for software developers because of its powerful capabilities and open architecture; also, it operates on the Java Virtual Machine. Android API is open and allows easy access to the hardware components. Android devices provide numerous communication interfaces like USB, Wi-Fi, and Bluetooth that can be used to connect to robots. It is a great platform for robotics system control because it is much cheaper than any other ARM-based processing unit.

B. Research Objectives

The objective of this research was to implement a simple and effective application controller for the Rover Spy Tank unit. It will provide to the users, educators, scientists, and companies an open-source application controller software in order to manipulate the Rover Spy Tank robot by means of a mobile device with an Android operating system. We expect that the controller serves as a basis for future research and may provide academic enrichment. The reasons for selecting this rover and a mobile device with an Android operating system are its simplicity, great number of users, and availability of low-cost devices. The rover cost about \$99 and a typical tablet cost under \$100. With these two devices, the academic and research communities could have a rover with functions such as telepresence to guide the rover in an environment with obstacles and object recognition using computer vision algorithms provided by the OpenCV library. This rover with our controller application constitutes an excellent topic for research, in addition to a great contribution to education and exploration. In addition to the growth and popularity of mobile devices, we provide the community with an application controller with broad functionality that is available to the entire world through Google Play at no cost.

II. RELATED WORK

Soul et al. [9] wrote: “The incorporation of robots in the Computer Science classroom (CS) can increase excitement, improve enrollment and retention, and bring relevance to a variety of study areas.” The main objective of this research is to attract the students’ attention and offer a strong research tool that is both accessible and economical. He provides a table which depicts examples of such robots [9].



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Falaki et al. expected that the sales of smart phones exceed those of desktop computers. This data was an important factor at the moment of deciding which platform was given to be used for implementing the application controller for the rover.

Marta [6] indicated that the IDC (International Data Corporation) revealed that the growth of tablets is 311.76%, a very high number considering that it occurred in a term of three years. The following is the number of tablets sold for each of the three years: 17 million (2010), 44 million (2011) and 70 million (2012). According to these numbers, using an Android mobile device with an application to control the rover is a very effective option to attract the attention of users and researchers because the cost is low, it is accessible, and its research and education applications are endless.

Luimula et al. [5] studied three interfaces for the effective creation of user interfaces to control robots: i) telepresence interface, in which a camera is placed on the robot which will remotely control the robot through a computer. ii) Use radio frequency identifiers (RFID) along the path the robot will follow in order to identify the obstacles. iii) Use a combination of the previous two which shows the plane with the robot's position and the obstacles identified with the RFID sensors, plus a front and rear view of the robot by means of cameras attached to it.

Holz et al. [4] developed two theses for creating gender-sensitive and activity learning environments. In one of this activities the students programmed an Android app in Java to remote control a Lego Mindstorms NXT Robot. In the other activity they programmed in Java a wheeled Arduino microcontroller. Their evaluation results showed that using smartphones can have a highly motivational effect on students of both sexes.

Tamzarian [10] analyzed the rover spy tank in deep detail. He located, at the bottom of the rover; three-pin interface port to access the serial console on the device using any TTL capable hardware. The main booting log information of the serial console were: W90P745 Boot Loader, Linux version 2.4.20-uc0, GCC version 3.0, Processor: Winbond W90N745, Video Audio Access Protocol: MO_O Camera operation protocol, MO_V Camera AV transmission protocol, and several CGI scripts such as decoder_control.cgi, wifi_car_control.cgi, camera_control.cgi, get_params.cgi, set_params.cgi, and check_user.cgi. Upon connecting to the ad-hoc Wi-Fi that the rover creates, the rover can be accessed by the app-specified IP address (<http://192.168.1.100/>) and log on with user/pass: AC13/AC13. The in-built web server of the rover responds to the CGI scripts listed previously at boot log information.

Rossum [8] used nmap and tcpdump tools to reveal how to access the rover server. This server can be accessed issuing an HTTP GET request http://192.168.1.100/check_user.cgi?user=AC13&pwd=AC13. The author also discovered that the binary protocol used to command the camera and the rover movements are MO_O and MO_V protocols. The binary stream starts with these protocols and with a sequence of bytes which represents a command. In order to send these commands from the Android device to the robot is necessary to open and close TCP sockets at the same port after authentication. The library is implemented in the AC13 Rover Spy Tank controller, which can be found in the Google Play Store. As the RoverOpen app [2], is open source. The application independent Java library is also available on GitHub.

III. EXPERIMENTAL SETUP

In this research, we used several software, hardware tools, and resources.

A. Rover Spy Tank

The rover used in this research was based on Rover Spy Tank from Brookstone [1]. The rover was equipped with a sensor camera (320x240 – 640x480), Wi-Fi (802.11b/g), Infrared Control Night vision lenses (850 nm), two motors, a microphone, and 6 AA batteries. The rover also had features such as 2.30 pounds, dimensions with antennae of 10/25” w x 7.75” l x 7.75” h, a battery operation time of 2.5 hours, and Wi-Fi control distance of 200 feet without obstruction and 100 feet with obstruction.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016



Figure 1. Spy Rover Tank from Brookstone [1]

Figure 1 shows the Spy Rover Tank which is the simplest, low-cost platform and the most suitable for the small-sized, light, battery-driven vehicle. The vehicle has two driving wheels and the angular velocities of the two wheels are independently controlled. The client application connects to the mobile robot with the Bluetooth module. We implemented a controller program for the rover. This controller provides the following capabilities: i) telepresence experience when using the rover's front camera, ii) control of the rover's movements in two dimensions, iii) changes in the resolution of the images provided by the rover, iv) night vision by means of the infrared camera, v) capture images for recognizing and learning about objects, and vi) detection of faces. The velocity of the video can vary when an Android tablet is being used instead of an Apple iPad. The reason why we implemented an open-source application controller is because of the difficulty of the researchers to perform reverse engineering of the communication protocol of the proprietary hardware and its organization performance.

B. Hardware Tools

A standard size tablet (7"- 10.1") with Android OS or similar mobile device; battery pack with 9 volt for a maximum operating time of 2 and half hours; Wi-Fi connectivity with 802.11b/g; sensor camera (320x240 – 640x480); a desktop PC with the capacity of using the development tools and other elements such as the virtual Android machine, which is included in the Android SDK.

C. Software Tools.

Java programming language; Android Development Toolkit version 18; Indigo Eclipse; OpenCV2.4.8; Android operating system 3.0; WiresShark; and GIMP. The rover's software features are listed in [10]. The rover serves as a Wi-Fi access point and a video streaming server. This server can transmit real-time video and accept control command from a client through Wi-Fi. It controls Rover's movement, the camera streaming, audio, and infrared light. Communicator library used in this research is based on the RoverOpen [2].

D. Communication Resource.

Communication requirements consist of an Android device and a GUI designed for performing a particular operation. For enabling the communication of the rover with the android device the ad-hoc Wi-Fi (802.11b/g) communication technology is used. This communication device is attached to the rover that receives the data from the mobile and can also send the data. The ad-hoc wireless communication protocol, running at the speed of 2.4 GHz with the architecture of client-server, is suitable for performing image capturing, video recording and rover movement operations. This communication protocol is designed for mobile devices and gives the connectivity between two devices using their MAC address.

IV. APPLICATION SOFTWARE IMPLEMENTATION

In this section, we present the implementation of an Android client software interface called AC13 Controller for the Rover Spy Tank unit. This controller was written in java using Eclipse as the IDE. It has basic intuitive tools that provide the user with an easy way to control the movements of the Rover and visualizing the real-time streaming video data. This software communicates with the server running in the rover. We used the open source library provided at [2] for sending and receiving commands as well as receiving video data. This section provides a technical explanation of the implementation of some of the components that constitute the AC13 controller. The implementation is organized in



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

a structured and scalable manner and it is divided into different packages according to their functionality. The application is made up by a group of Java classes that cover all the functions of the application. These functions include: user interface, functionality, and communication with the rover. OpenCV library was used to implement basic advanced image processing functionalities.

AC13Communicator. This package is the most crucial of the controller. It contains all the communication implementation with the Rover Spy Tank unit. It was designed to function as a multiplatform library that serves as a communication interface with the rover. The rover contains a web server with CGI scripts through which it transmits all its parameters, sends images, receives commands and transmits sound. To interact with this server, we used an open source library called RoverOpen [2]. This library creates a socket to send and receive a vector of bytes. By decoding the bytes, it was possible to identify some reactions in the rover; for example, moving to the left, stopping the wheels, and receiving the image that the rover is observing through its camera. Among the capabilities of this communication library is the constant flow of the image data observed by the rover and the ability to move the rover in all directions. With these basic functions, practically any telepresence interaction between the rover and other mobile device or a computer, can be attained. It is important to mention that this library is still in the development stage, and it can continue to grow in order to provide a more enriching experience. Among the functionalities that are still missing from this library are included the flow of sound coming from the rover and moving the front lens up and down.

AC13Controller. This package contains all the basic implementation with the main methods of a typical application for Android and makes reference to all the other packages. This package consists of all the initial animation and the main activities that are in charge of handling all the events of the controller. It initiates communication with the rover for sending the movement commands to manipulate the rover. We used an enhanced version of the RoverOpen [2] based library for sending and receiving commands as well as receiving video data. The basic functionality of the communication is based on a multi-thread mechanism which extracts the image from the library that communicates with the rover at least eight times per second. After the image is obtained, it is displayed on the interface to obtain a live video effect. Another basic functionality of the controller is the manipulation of the rover's movements. To obtain this movement, it is necessary to send a series of commands to the rover through the communication library, but these commands should be sent according to the direction desired by the user. We use an interface similar to a joystick that provides Cartesian coordinates to indicate the direction to move the rover. In order to find the direction where to move we used the inverse tangent formula to determine the bearing angle: $\theta = \tan^{-1} \frac{y}{x} * \frac{180}{\pi}$. This angle is evaluated in a range of values to determine the direction. These ranges are defined in the following manner: 45° - 125° (FORWARD), 136° - 225° (LEFT), 226° - 315° (BACKWARD), and 316° - 405° (RIGHT). Once the direction was found, a command was sent to the rover to move in the specified direction.

AdvancedAlgorithms. This package contains algorithms that use OpenCV (Open Computer Vision) library. This package performs analysis and processing of images. Some of the algorithms in this package includes face detection, feature extraction, object recognition, and supervised learning using different techniques such as support vector machines and neural networks.

DataAccess. The data access package contains classes that serve as an interface for the storage and reading of data in secondary memory. There are various situations in the controller that require storage, such as the case of storing images taken from the rover, adjustments and user preferences. An important future implementation could be the storage of binary data for image recognition, among other algorithms of supervised learning.

Global. This package contains constants and variables that are used to share references and data between the other packages in a centralized manner whenever necessary and perform global changes in the controller's behaviour.

ImageCrop. This package contains multiple classes that support image-cropping activity that is used in the controller. This functionality allows for the interactive cutting of an image for further processing.

Widgets. This package contains various components for the graphic interface of the application that the Android does not include by default such as the Joy Stick component.

V. RESULTS

The result of the implementation of the controller includes a simple and intuitive design with the capability of expansion to much more functionalities. Functionality and effective communication was possible with the Rover Spy Tank rover unit, thanks to the implementation of the RoverOpen library that is used to connect to the rover's server. The creation of a communication library opens the doors to future research with this rover since it was written in Java,

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

thus the implementation will not be tied to any particular platform can be used in a great variety of operational systems and devices, such as computers, intelligent phones, tablets, and any other machine that has a Wi-Fi connection and can run the Java code.

A. Application Software Design

The user will handle the rover and its actions via a GUI designed for android systems. The interface of the controller was designed for large screens. The controller application has three main activities. The first is a simple activity that demonstrates an entrance animation in which data is changed from secondary memory to primary memory. The second and main activity demonstrates the interface of the controller and provides its options and functionalities. This activity contains a frame in which the live video received from the robot is displayed. It also includes a joystick that serves to control the movements of the robot. The activity demonstrates other controls such as the infrared camera switch, image recognition menu, and image capture, which is stored in secondary memory afterwards. Finally, the activity provides a button that triggers an adjustment dialog in which the different parameters of the controller can be set. The last, but not least important activity allows for the cropping images. The goal of this activity is to allow the user an intuitive interface to cut the sections of the image received by the rover for processing in functions such as save in secondary memory, object detection, and supervised learning.



Figure 2. User interface of the AC13 controller

The user interface of the AC13 controller (Figure 2) presents the main activity of the application. The interface shows the different controls that the controller provides. The main panel in which the image is displayed, in this specific case, it is the image of a video game control. The joystick that serves to control the movements of the rover is located to the left of this GUI. At the lower part of the screen contains the button that opens the adjustment dialog of the controller. Finally, to the right the buttons for different functionalities are located such as the infrared startup, taking photos from the screen, and image recognition.



Figure 3. Recursive view of the rover's camera

The recursive view of the rover's camera (Figure 3. Recursive view of the rover's camera) presents a view of the robot observing the tablet, which is at the same time using the controller. When the rover observes an image of its own vision, this causes a mirror effect in which the image of the vision is reflected recursively.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016



Figure 4. Adjustment panel of the AC13 controller

Figure 4 presents the adjustment panel of the controller. This GUI shows two options that allow setting the resolution of the image received by the robot and adjusting the sensitivity of the joystick when moving the rover.



Figure 5. Infrared vision of the rover in the AC13 controller

Figure 5 presents the infrared vision of the robot. It was activated by pressing the blue colored button located at the right side. This functionality of the robot allows for a very clear vision in environments with no or little sources of light.

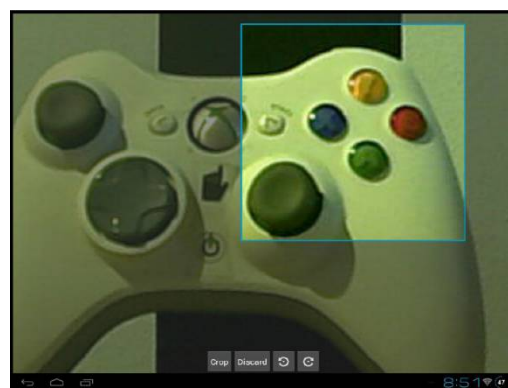


Figure 6. Image-cropping functionality of the AC13 controller

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Figure 6 shows the image-cropping functionality. It serves as a tool for preparing images to be processed for different purposes such as image recognition and storage of the selected region in selected image. In this case, a region of the image is being cut to be processed at a later time by the image-recognition algorithms.

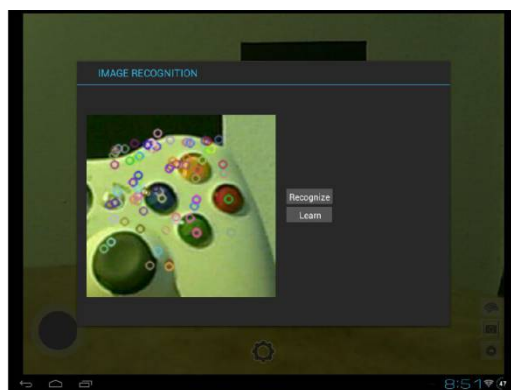


Figure 7. Image recognition algorithm of the AC13 controller

In Figure 7 the result of the cropped image is shown after it is processed by one of the image-recognition algorithms. The colored circles above the image represent the key points of the image which can be said to be the key characteristics. The points can be used to match future images with the same object presented.

B. Software License

The implemented software was published in Google Play without any cost, under the name “Rover Spy Tank Controller” and username UCETA. There is a short video demonstrating the controller in action. In order to allow widespread use by the research community as well as practitioners, we released the communicator and controller library for the Rover Spy Tank as open-source software under a GNU General Public License. The project is available in [7]. We invite the community to share extensions via this platform

VI. CONCLUSION AND FUTURE WORK

A. Conclusion

Communicating with robots by means of mobile devices with wireless technologies are increasing research and education opportunities in robotics. Best of all, these devices can be economically accessible to many people. One of the main purposes of this research was to demonstrate the endless possibilities of new mobile technologies and the integration with technologies such as robotics. In this paper, an open source application controller for the Rover Spy Tank from Brookstone was developed and made publicly available under a GPL license. The application controller was written in Java. It used RoverOpen library to communicate with the rover's server. A feature of the application controller is that users can interact with real rover motion and that it provides real time video streaming by using Android mobile devices. Key features of the library are: the AC13Communicator package that provides the communication interface with the rover; AC13Controller package that initiates the communication with the rover and sends the movement commands to manipulate the rover; advanced algorithms that enable OpenCV library to execute basic computer vision algorithms; DataAccess package that performs the basic storage of image data for further processing. Global Package that contains elements that other packages share between them; ImageCrop package to perform image-cropping activity in the controller. The library can be downloaded by students, researchers, and enthusiasts and continue to be developed by the open source community. New features will be added and made publicly available through a publicly-accessible code repository.

B. Future Developments

Although the rover application controller was designed on the basis of a number of different user usability studies, not every aspect of possible features is covered. Therefore, the library could be adapted and further developed by future researchers for additional applications design and development. There are already a number of ideas on how to extend the library. It is expected that the same be optimized as soon as possible. Some optimizations and additional functionalities to the controller are the following:



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

- Being able to connect via WebSocket directly from the browser. Move the UI from an Android app to a more accessible *HTML5* web app.
- Increasing the video frame rate and resolution.
- Including Virtual Reality Handsets API to enhance telepresence experience.
- Allowing Gyroscope-based movements.
- Reverse engineering the rover's audio protocol.
- Including advance object detection using pre-trained classifiers and rich deep learning tools and services [11, 12].
- Routing recording and replay, autonomous driving via supervised training.

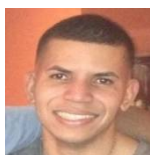
ACKNOWLEDGEMENTS

The authors would like to thanks, Anne van Rossum (author of the RoverOpen library) for the guidance to develop our communication library, which the rover application controller depends on. Also, we thank Antonio Huertas, professor of Computer Science Department at University of Puerto Rico at Bayamón for advising us on technical aspects of this work.

REFERENCES

1. Brookstone. "Rover App-Controlled Spy Tank with Night Vision", Retrieved from URL: [<http://www.brookstone.com/rover-remote-control-spy-tank-for-ipad/719302p.html>], April 12, 2016.
2. DoBots. "RoverOpen", Retrieved from URL: [<https://github.com/mrquincle/roveropen>], April 12, 2016.
3. Falaki,H.,Mahajan,R.,Kandula, S., Lymberopoulos, D., Govindan,R. and Estrin, D., "Diversity in smartphone usage", In Proceedings of the 8th international conference on Mobile systems, applications, and services (MobiSys '10), ACM, New York, NY, USA, pp. 179-194,2010.
4. Holz,J.,Leonhardt,T. and Schroeder, U., "Using smartphones to motivate secondary school students for informatics", In Proceedings of the 11th Koli Calling International Conference on Computing Education Research (Koli Calling '11), ACM, New York, NY, USA, pp. 89-94, 2011.
5. Luimula, M., Säskilähti, K.,Partala, T., Pieskä, S. and Alaspää, J., "Remote navigation of a mobile robot in an RFID-augmented environment", Personal Ubiquitous Comput.14, pp. 125-136,2010.
6. Marta, R., "Mobile documentation: Usability guidelines, and considerations for providing documentation on Kindle, tablets, and smartphones", Professional Communication Conference (IPCC), IEEE International, pp.1-13, 17-19, 2011.
7. Mastayoda01. "AC13 Java Library: Open source Java library to control the AC13 Rover", Retrieved from URL: [<https://sourceforge.net/projects/ac13javalibrary/>], April 12, 2016.
8. Rossum, A. V., "Brookstone Rover AC13", Retrieved from URL: [<https://dobots.nl/2012/01/09/brookstone-rover-ac13/>], April 2012, 2016.
9. Soule, T. and Heckendorn, R. B., "COTSbots: computationally powerful, low-cost robots for Computer Science curriculums", J. Comput. Sci. Coll. 27, 1, pp 180-187, 2011.
10. Tamzarian,A., "Topic: AC13 Brookstone Rover", Retrieved from URL: [<http://www.openipcam.com/forum/index.php?topic=261.0>], April 12, 2016.
11. DeepDetect. "DeepDetect: Open Source Deep Learning Server", Retrieved from URL: [<http://www.deepdetect.com>], April 17, 2016.
12. Google. "TensorFlow", Retrieved from URL. [<https://www.tensorflow.org>], April 17, 2016.

BIOGRAPHY



PhD. Student **Victor Santos Uceta** received B.S. in Computer Science from the University of Puerto Rico at Bayamón in 2012. Since 2013 has been pursuing a Ph.D. degree in Computer Science at Purdue University with focus on Distributed Systems, Software Engineering, and applications of machine learning in interdisciplinary fields.



Dr. **Elio Lozano Inca** received B.S. in mathematics from National University of San Antonio Abad of Cusco, Perú in 2000. He received a M.S. in Scientific Computing and a Ph.D. in Computer and Information Science and Engineering at University of University of Puerto Rico at Mayaguez Campus in 2003 and 2006 respectively. Dr. Lozano is an Associate Professor at the department of Computer Science, University of Puerto Rico at Bayamón, San Juan, PR, 00959 USA. He has developed several software and firmware for different architectures.