Exploration Robot with Stereo vision

Vladislav Richter, Miroslav Skrbek
richterv2@fit.cvut.cz, skrbek@fit.cvut.cz
Faculty of Information Technology, Czech Technical University in Prague

***Abstract*—This article presents a four-wheeled robot equipped with two cameras for stereoscopic vision. The robot is remotely controlled from a mobile device running the Android operating system. On the display, an operator sees 3D anaglyph image from stereoscopic cameras and the depth map that allows determining the distance to nearby objects in front of the robot. The robot has rich set of advanced sensors like inertial sensors, infrared low distance sensors, microphone array for sound localization, speech recognition and voice output. Wi-Fi is used for connection between the robot and the mobile device. Control commands are sent via the TCP/IP protocols to the robot. A robot control application on the mobile device has been developed for the Android operating system. This is a layered application consisting of the communication and robot control API and the graphical user interface. On start, the application retrieves a list of available commands in the XML format describing capabilities of the robot. The application can adapt its user interface according to this list. We also tested the Kinect sensor as a replacement of cameras and have developed a control application receiving all data available on Kinect.**

I. Introduction

At present the mportance of exploration robots grows. They are used in many fields especially in aerospace, waste disposal after disasters, military, police, but also in common life. Depending on deployment they vary in degree of their autonomy. For instance fully autonomous pipe explore robot is presented in [1].

Embedding 3D vision in a robot provides significant space information for a robot controller in case of fully autonomous robot, or it provides a 3D stereoscopic image to the operator. Several approaches were used for viewing 3D space in front of the robot. In [4] and [5] two cameras mounted at certain distance are used to map 3D space. Another way is to use 3D Laser Scanner as described in [3]. Tablets and mobile phones are cheap and easy to use peripherals that the operator can use to control the robot. A solution applied to a spider robot can be found in [2].

In this article we present a four-wheeled robot equipped with two cameras for stereoscopic vision. The robot is remotely controlled from a mobile device running Android operating system as shown in Fig. 1. On the display, an operator sees 3D anaglyph image from stereoscopic cameras and the depth map that allows determining the distance to nearby objects in front of the robot. The robot has rich set of advanced sensors like inertial sensors, infrared low distance sensors, microphone array for sound localization, speech recognition and voice output.

Wi-Fi is used for connection between the robot and the mobile device. Control commands are sent via the TCP/IP protocols to the robot. The sensor data including images are sent back to the mobile device and shown on its display in numerical and graphical form.

Both software and hardware of the robot is constantly improved. We also tested the Kinect sensor as a replacement of cameras. Latest version of the server running on a small notebook on the robot transmits all Kinect data including the high quality depth map to the mobile device.

II. Hardware

The four-wheeled robot has two pairs of motors, each pair is connected in parallel. The module with two pulse-width modulated H-bridges controls the left and the right side pair of motors. Wheels with 11cm diameter provide the speed of up to 60cm per second. The robot is powered by NiMH battery packs with tens cells and nominal 12V voltage.

A. Control boards

Electronic control boards are split into several layers. A PWM module and a sensor board are on the lowest level. A control board is on the intermediate level. A main board is on the top level. The computational power grows from the bottom to the top of the hierarchy. The sensor board has ATmega88 controller installed that is the heart of the odometer module. The control board, which is responsible for collecting sensor data and motor control, has 16-bit dsPIC33FJ256GP506 microcontroller. The main board is based on the Roboard RB110 which is the pentium based x86 compatible computer system with 256MB RAM running at 1GHz. The main board has Gentoo Linux installed and provides sufficient power for basic image and voice processing and communication.

B. Sensors

The robot is equipped with a rich set of sensors. The rear wheels have incremental quadrature encoders mounted on its motors. Impulses from the encoders are processed by the odometer unit that provides the distance and the velocity via the I^2C bus. Battery voltage and current sensors provide analog voltages measured by the A/D converter of the control board microcontroller. The 3-axis accelerometer and 3-axes gyroscope is mounted in the center of the robot. They allow to observe the terrain profile. The system temperature sensor, the odometer module and gyroscope share the same I^2C bus.

Near front and back obstacles are detected by two IR sensor with the range limited to 80 cm distance. They are mounted on the rear and front side of the robot chassis. For longer distance measurement, two cameras with 640x480 resolution are mounted on the top of the robot. The cameras create a

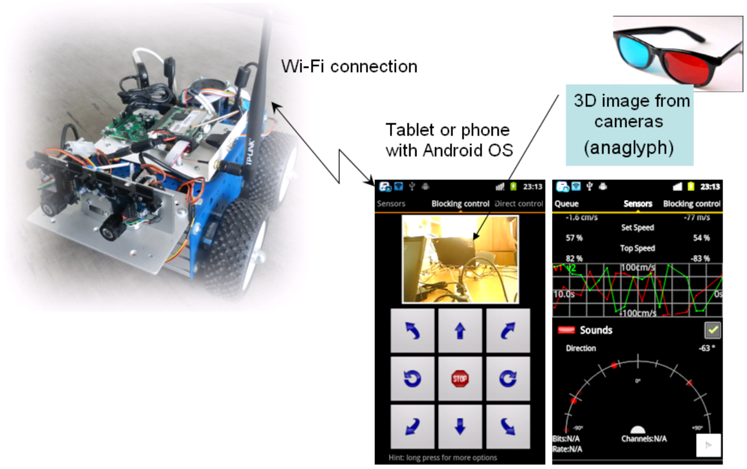

Figure 1. Exploration robot with stereoscopic vision

stereo pair that enables the depth map calculation. The cameras are rotably mounted and vertical angle is controlled by a servo drive.

The cameras have four-microphone array integrated. It enables sound source localization and can provide high quality voice for speech recognition.

C. Voice output

The robot is equipped with a USB connected sound card and speaker. In combination with Festival speech synthesizer, it provides voice output in Czech and English languages. It is an important feature for user-robot interaction and robot diagnostics because no display is mounted on.

III. Software

A. Robot side

The main board operates as a Wi-Fi access point. The server application running under the Linux operating system accepts connetions from remote control applications. For easy debugging we use textual communication protocol defining a collection of commands for wheel control (for example `set_wheel_speed`, `forward`, `backward`, `rotate`), reading camera images (`get_eye_image`, `get_depth_map`), reading status information (`get_info`). The server integrates a module for stereo vision and sound localization that process audio and video data from cameras directly attached to the main board. The server reads the sensor data from the control board which is designed as an USB CDC device and mapped as a Linux tty device. The communication protocol is mostly composed from the single letter commands decreasing the communication overhead.

Images from the cameras are processed on the robot and depth map is generated on demand the control application. Depending on algorithm and output quality selected, the depth map calculation takes from 0.3 to 3 seconds. Therefore the server is implemented as a multi-thread application to avoid unnecessary blocking of camera image and audio streams by depth map calculation. The stereo vision is based on OpenCV algorithms including an integrated camera calibration procedure.

B. Android Control Application

To remotely control the robot, three distinct application were developed. Each at a different development stage of the robot with different goal and more experience. But none of the later two applications is simply an upgrade. There are significant difference among them hence they are described separately.

All developed applications have a set of common features. They are Android based applications capable to run on small mobile devices like smartphones or tablets. Mobile devices in general provide convenient way to remotely control the robot moving around. A stationary control center seems to be very impractical in this case. Additionally the Android operating system allows to run applications on various devices. All applications shows the robot's telemetry to the operator in textual and graphical form and provides touch elements on the display to directly control the robot motors in real-time.

C. First Control Application

The earlier application is based on the Android 2.3 which is compatible with any number of devices but it cannot take advantage of advanced Android features of later versions. The structure of the application is shown in Fig. 2. This application is designed in two main layers. The robust robot's commands and data processing API (written in pure Java) are at the bottom. The data processing API provides an interface for accessing all functions of the robot. At the start this API establishes TCP/IP communication with the robot and requests a XML file describing all robot sensors and commands. After parsing this file, the API exposes appropriate interfaces to the higher GUI layer.

The GUI layer is based on Android SDK and exposed interfaces. The GUI provides 3D model of the Robot rotated in space according to the robot's accelerometer data, current reading and history graph of the accelerometer data, gyroscope data, wheel speed, motor speed, distance from obstacles, battery voltage and current and system temperature. It can also receive and localize sound from robot's microphone and display the sound source direction on a handy display. It also receives and displays images from robot's cameras, depth map. The anaglyph provide 3D view when operator has blue-red glasses. A virtual joystick and motion keypad both drawn on the display are provided to control robot's movement. The virtual joystick is handled by user's fingers which directly sets the speed of motors in real-time. The motion keypad consists of a set of buttons representing clearly defined movements like move forward, backward, rotate left/right etc. These commands are defined at the robot side and can be chained and executed in a sequence. If a command has not a corresponding graphical control on the display, it may be also executed. The GUI shows a full list of available commands parsed form the XML file. An operator may select any command, enter its parameters and execute it.

D. Second Control Application

The second application was developed with slightly different goals. Although the earlier application has been excellent in presentation of telemetry data and very flexible for all aspects of control, it was somewhat cumbersome for remote robot navigation. A large number of display pages and control elements caused cluttered GUI and the operator was not able to move robot and see relevant data at the same time. The second application is based on Android 4.0 and expects the robot with the default configuration for the sake of simplicity. It is highly modular so that new functions may be easily added. All the control elements and telemetry information are now displayed at a single screen with the big camera image in the center. The telemetry data overlaid the image as a head-up display (HUD). Each side of the display has a slider for one thump that controls speed of one side the robot's wheels. On the top of the image, there are small icons which can switch to anaglyph display, disable HUD and perform other commands. Android 4 allows this part to be very easily editable and extended for other configurations. In order to improve the performance, the camera images received in YUY2 format are decoded directly on the graphics chip by means of the OpenGL 2.0. This approach has proved very successful since it was tested by a wide range of users with minimal experience, including small children.

E. Kinect

The user interface of the last application is based on the second one with minor improvements and different color theme. But it is differently implemented. This application has been developed from grounds up to cooperate with the robot equipped by a Kinect sensor. The Kinect sensor is now focal point of the entire application which can receive and display all Kinect sensor data including color image stream, infrared image stream, depth map stream, skeleton tracking data, audio data, and accelerometer data. For the first time this application also comes with it's own server application which runs on robot. Communication has been improved by introducing the UDP transport protocol which is more efficient for uni-directional data steams and increases throughput. Besides simple implementation it allows to stream the same data to multiple receivers at once.

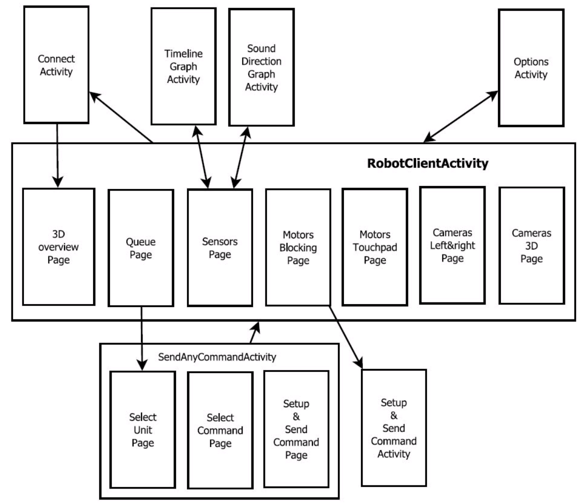

Figure 2. Internal structure of earlier application

Acknowledgment

Authors of this article thanks to many students who worked on hardware and software of this robot in the past and created the background for this final work. Namely we thank to Robert David, Petr Petrouš and Tomáš Klinský that implemented the video, audio and server modules as their bachelor thesis.

This research has been supported by grant *SGS12/096/OHK3/1T/18 Advanced perceptive methods for robotic platform.*

References

[1] Jong-Hoon Kim, G. Sharma, and S.S. Iyengar. Famper: A fully autonomous mobile robot for pipeline exploration. In *Industrial Technology (ICIT), 2010 IEEE International Conference on*, pages 517–523, 2010.

[2] Sung Wook Moon, Young Jin Kim, Ho Jun Myeong, Chang Soo Kim, Nam Ju Cha, and Dong Hwan Kim. Implementation of smartphone environment remote control and monitoring system for android operating system-based robot platform. In *Ubiquitous Robots and Ambient Intelligence (URAI), 2011 8th International Conference on*, pages 211–214, 2011.

[3] J. Ryde and Huosheng Hu. 3D laser range scanner with hemispherical field of view for robot navigation. In *Advanced Intelligent Mechatronics, 2008. AIM 2008. IEEE/ASME International Conference on*, pages 891–896, 2008.

[4] Kwang soo Kim, Wook Bahn, Changhun Lee, Tae jae Lee, M.M. Shaikh, and Kwang soo Kim. Vision system for mobile robots for tracking moving targets, based on robot motion and stereo vision information. In *System Integration (SII), 2011 IEEE/SICE International Symposium on*, pages 634–639, 2011.

[5] Zhao Yong-guo, Cheng Wei, Jia Lei, and Ma Si-le. The obstacle avoidance and navigation based on stereo vision for mobile robot. In *Optoelectronics and Image Processing (ICOIP), 2010 International Conference on*, volume 2, pages 565–568, 2010.