

PAPER • OPEN ACCESS

Web-Based Robot Control Interface

To cite this article: Zixuan Xiao and Yulin Xu 2019 *IOP Conf. Ser.: Earth Environ. Sci.* **252** 042112

View the [article online](#) for updates and enhancements.

Web-Based Robot Control Interface

Zixuan Xiao^a, Yulin Xu^b

School of Mechanical and Engineering Shanghai University, Shanghai, China

^a849048256@qq.com, ^bxuyulin@shu.edu.cn

Abstract. A web-based robot control system interface is designed. This interface can be used by users to control the motion of target robot, observe 3D model, and monitor the real-time value of the robot joints through PC or mobile device. The advantage of this system lies in its versatility and portability, allowing users to open the interface to control target robot in any place. This system mainly uses robot operating system (ROS), Html 5, C++, JavaScript and PHP technology. JavaScript library provided by ROS authority is used to build the front-end interface. PHP is used to create a user registration login system. Ros3djs is used to realize the establishment of dynamic model of robot simulation. The C++ library provided by ROS is used to synchronize the communication between the ROS node and the robot. Finally, the human-computer interaction system of the 16-DOF robotic hand is successfully implemented. According to the results, the system has good interactivity, versatility and portability.

1. Introduction

The "Third Industrial Revolution" since the 1950s has promoted the rapid development of space technology and electronic computer technology. In the 21st century, computer information technology has stepped into a prosperous era. Nowadays, thanks to the vigorous development of computer technology, people have strong scientific technology and hardware power to realize their ambition in the field of robotics, making world enter the smart factory era of "Industry 4.0". In industrial developed countries, industrial robots have been widely used in many fields such as automobile, auto parts manufacturing, mechanical processing, electrical and electronics, rubber and plastics, food, logistics, and manufacturing, becoming a core force for the country to develop high-end manufacturing and achieve high automation in manufacturing [1].

With the advancement of intelligent robot technology, the demand for civilization and generalization of robots has increased, and easy-to-use human-computer interaction system has gradually become an important research topic. So far, many robot control interfaces have to be used on computers or special devices due to operating system limitations, without good versatility and portability. In order to improve this problem, the literature [2] adopts the Android phone, which uses the OTG interface to connect the joystick handle, and wireless video transmission scheme to achieve real-time visual control of the mobile robotic hands; the KUKA smartPad device is designed for the KUKA robot that has been put into industrial production, and the KUKA smartHMI operating system is equipped to carry out control operations, while increasing portability of the control interface, it also limits the user's application platform. In addition, it is also an important issue to establish model for robots. In literature [3], a 3-DOF mechanical leg is proposed, and MATLAB is used to build a 3D simulation model which can only run on a computer platform. Literature [4] builds a 3D model of each



joint of the robot by using the 3D computer aided design software Remo3D, uses standard COLLADA to describe the interconnection between the joints of the robot, and carries out Real-time graphical modeling of robots using HTML5 language.

Combining the above factors, this article is developing under ROS operating system, using URDF modeling, a unified robot description format. HTML5 technology is used to design web based robot control interface. Functional modules provided by ROS operating system are used to implement interface-host-robot communication, so that the robot can be controlled by frontend interface. Through the robot control system proposed by this article, the target robot can be remotely controlled by browser on both mobile device and PC, and dynamic motion of the target robot can be observed in real time. In addition, the fact that host is set under the ROS operating system is helpful to further development of robot function.

2. Technology Principle of Robot Control System

2.1. ROS Robot System

As an open source, ROS (Robot Operating System) system is a simulation operating system for robots. It provides functions similar to those provided by computer operating system, including hardware abstraction description, underlying driver management, execution of shared functions, message passing between programs, and program distribution package management. Some tools and libraries are also provided to acquire, build, write, and run multi-machine integrated programs [5-6]. The structure diagram is shown in Fig.1, the basic unit of ROS system is node, and the core structure is topic subscription and message publishing model. This structure enhances the distribution of ROS system, weakens the coupling and dependencies between modules, and improves the efficiency of robot system development. When a node subscribes to a topic, once the topic posts a new message, the node can receive the message and take action.

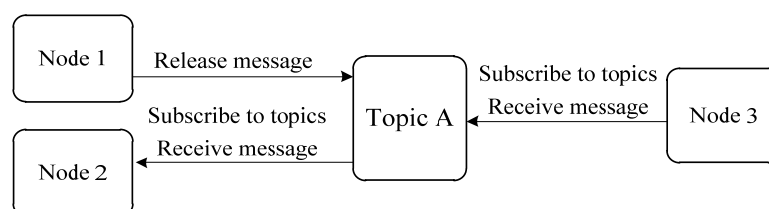


Figure 1. Topic-notification example graph

As the perfect solution for robot system development, the ROS system can run on the Linux operating system or embedded system, with rich toolkit and multiple language library [7].

Rosbridge is used by us to communicate with the front-end interface. Rosbridge (Rosbridge_Suite) is a function package provided by ROS authority for developers to communicate interactively in both non-ROS systems and ROS systems. Rosbridge consists primarily of the Rosbridge Protocol and the Rosbridge Implementation. Where, the Protocol part provides a specific format for communication between the non-ROS system and the ROS system, including topic subscription, message release, service call, parameter setting and acquisition, picture information transfer, etc., which are all in JSON format. The Implementation part is a concrete implementation of Rosbridge, including Rosapi, Rosbridge_library, Rosbridge_Server and other packages.

2.2. WebSocket Protocol

WebSocket protocol is used by Rosbridge to communicate between ROS systems and non-ROS systems. A big drawback of the traditional HTTP-based B/S communication is its one-way request response model that only supports the client to send a request to the server, and then the server returns the result. When the client needs to establish a long-term connection with the server, and want to receive dynamic data changes at the server side in real time, the client can only repeatedly send HTTP

requests to the server, which not only wastes resources, but also makes efficiency low [8]. The above problem is solved by WebSocket, which is a full-duplex communication protocol based on TCP. The implementation on the server side is relatively easy, with relatively lightweight data format, the small performance overhead and the efficient communication, which guarantees the real-time performance and saves a large amount of bandwidth.

2.3. WebGL Technology

As a JavaScript API, WebGL (Web Graphics Library) has emerged in the last decade, rendering interactive 3D and 2D graphics in any compatible web browser without the need for plugins [9]. This is the most compelling advantage of WebGL, because JavaScript and OpenGL ES 2.0 are allowed to be combined. By adding a JavaScript binding of OpenGL ES 2.0, hardware 3D accelerated rendering is provided for HTML5 Canvas. So without the need for any 3D rendering plugins, web developers can just use system graphics card to display 3D scenes and models smoothly in the browser directly [10], and create complex navigation and data visualization. For the robot web control interface of this system, WebGL is used to build a 3D simulation model.

3. Architecture Design of Robot Control System

The system architecture is mainly composed of five modules. The server side includes four modules, which are user registration login module, 3D simulation model display and control module, serial data communication module, robot and single chip module; the browser side contains one module: the robot control web interface module. The structure is shown in Fig. 2.

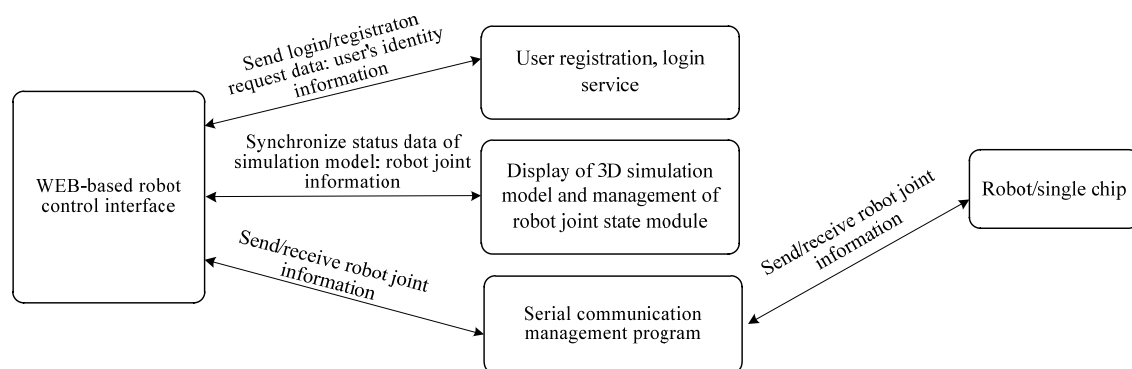


Figure 2. Topic-notification example graph

3.1. Core Program of Simulation Model

The core program of simulation model contains logic for reading robot model files, building 3D simulation models, and updating the state of the model in real time. It carries out the communication to the browser-side robotic web control interface through Rosbridge, and display the 3D model in real time on the web interface. Rviz tool is used to connect it to simultaneously display 3D models. Developed on ROS system, the module uses Rosbridge to communicate with the web interface. It subscribes to a topic of robot joint state, receives robot joint state messages, and updates the robot's joint state according to the message data. In addition, it contains a robot simulation model URDF file, which is read by the ROS simulation model program to build the simulation model.

3.2. User Registration/Login Service

This service runs on the Apache 2 server on the host side. Apache can run on most computer operating systems. Widely used due to its wide platform operation and security, it is one of the most popular web server software [11]. The registration and login service of this system uses the PHP 5 language. Because PHP is a server-side language, cooperating with the Apache server, it can be used to write server programs directly [12], with convenience and security. After receiving user's login HTTP

request, the server accesses the whitelisted user information file stored in the host. If the matching identity information is found, the server redirects to the robot control web interface. For the registration request, considering that the robot cannot open its control access to the public, the relevant person in charge should review the registration information and write the legitimate user to the white list.

3.3. Web-based Robot Control Interface

The web-based robot control interface is user-oriented. Rosbridge and ROS core programs are used at the web interface to carry out the communication, with high communication efficiency and real-time performance. Because web pages have great versatility, the civility and usage scenarios of robot control systems can be fully promote. From the perspective of user, there is no need to set up a complex software environment to run the control system. After opening the browser and inputting address, the user can log in to access to carry out the operation of the robot. And the state feedback of the robot can be learned through the WebGL-based simulation model. In addition, users don't even need a computer, and they can also access the robot control interface using a browser on a smart device such as a mobile phone or tablet PC, which avoids the user's geographic restrictions. The robot control interface of the system can control all the joints of the robot, and display the real data of the robot joint by reading the serial port.

3.4. Management Program of Serial Communication

This module is mainly responsible for processing data of communication data cable connected to the server side and the robot hardware module. The serial communication library Rosserial that comes with the ROS system is used in this module to manage the serial data cable. Rosserial is a ROS serial communication protocol that realize the P2P communication of ROS through the serial transmission medium. As a ROS node, the serial communication management program is responsible for reading the serial port data. When new data comes in from the serial data cable, the node reads the content and releases the data to the serial data topic. The node also subscribes to the joint state topic. When new joint information is posted to the joint state topic, the joint data is written into the serial communication data cable. Sample structure diagram is shown in Fig.3. There is a corresponding JS module that subscribes to the serial data topic in the web control interface module. When the serial communication node issues a serial data message to the serial data topic, JS will read the message and analyze the message content. The joint data is obtained from the serial port and displayed on the web interface.

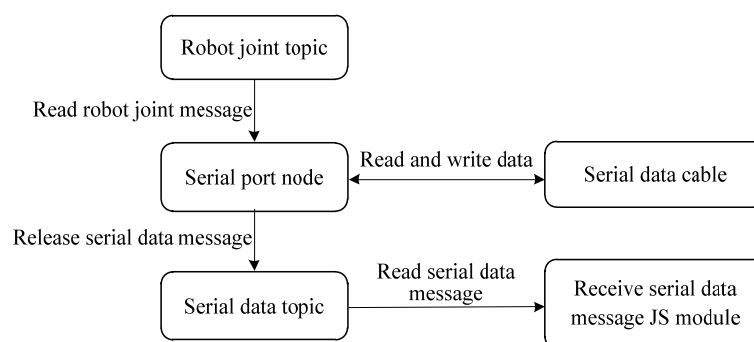


Figure 3. Serial communication model structure

3.5. System Workflow

The user opens the browser through the computer or mobile device. The robot web control interface is popped up by entering the URL, and automatically transferred to the user login interface. The login interface has a registration link through which the user can access to the registration page. The login and registration interface are based on the PHP5 service. Since PHP runs on the server side, the client cannot know the specific logic of the registration/login service code, which ensures the security of the identity information. After the registration application is approved, the user can log in to the system

with a username and password. After receiving the login identity information of the user, the login service will read the legal user whitelist Json file, and check whether the received username and password pair exists in the whitelist. If so, jump to the robot web control interface.

The robot web control interface is mainly composed of the simulation model display JS module, the slider-based joint control JS module, and the serial port data information display JS module. WebGL technology is used to display the 3D robot model of the simulation model on the page, and monitor the joint state of the robot. The simulation model is updated accordingly with the changes of the robot joint state. The slider-based joint control module reads the robot model, extracts all the robot joints and creates a one-to-one correspondence between the slider and the joint. The serial port data information display module subscribes to the serial port data topic. When a new message is released, the module parses the message content and updates the data information of the joints on the web control interface. When the user slides the slider on the control interface to generate an action, the JS code reads the new joint information and posts the message to the joint state topic. After receiving the message, the ROS simulation model program updates the joint information of the simulation model, and send the data to the web interface, so that the simulation model of the web interface also changes.

The serial communication management ROS node also subscribes to the joint state topic. When the new joint state information is received, the information is parsed and the Rosserial library is called to write the data to the serial port. The data packet is serially sent by the serial data cable to the microcontroller that controls the robot. The microcontroller has its own control logic to change the joints of the robot, realize remote control of the robot on the web interface. At the same time, the serial communication management node will publish the joint state message to the serial port data information topic when reading the joint information about the robot hardware from the serial port data cable. The serial port information JS module on the client web control interface reads and parses the message content. The joints and values are updated on the web interface to realize the function that the real joint data information on the robot side is displayed on the client interface.

4. System Implementation and Experimental Results

According to the web-based robot control interface proposed in this paper, the control system of 16 DOF robotic hand is realized in this experiment.

4.1. Software Architecture

The experimental system is built using Html, JS, PHP, C++ language according to the above system architecture. The specific structure is shown in Fig. 4.

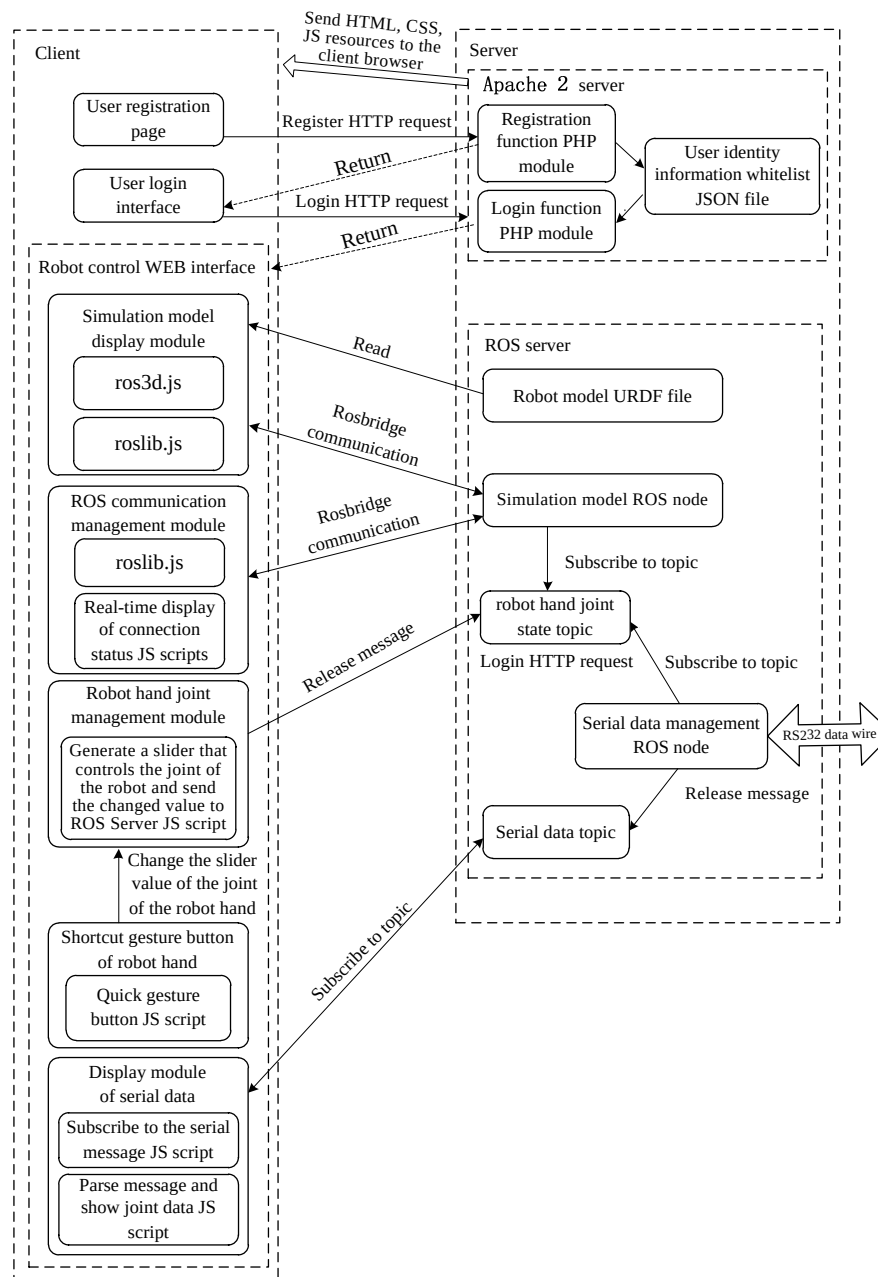


Figure 4. Software system architecture graph

4.2. System Environment

Client environment: Html5, jQuery, WebGL browser are supported; Ros3d and related packages; ROS system JS support package Roslibjs.

Host-side environment: Ubuntu 14 operating system; Apache 2 server; PHP 5; ROS system indigo version; Rosbridge and related packages; Rosserial and related packages; rs232 serial communication data cable; 16 DOF robotic hand.

4.3. Experimental Result

According to the software system architecture, a WEB-based robotic hand control system is successfully built. When the correct username and password are entered, the user can log in to the main interface of the robot control, as shown in Fig.5.



Figure 5. Robotic hand control UI

Zone 1 is the connection status display bar, displaying in real time whether the control interface is successfully connected to the host's ROS system. If the connection is successful, "Connected to ROS system" will be displayed. If the connection is disconnected, "The ROS system is not connected, please refresh the page and try again" will be displayed. If an error occurs in the connection, "an error occurs while trying to establish a connection with the ROS system, please refresh the page and try again" will be displayed.

Zone 2 is a 3D model display area. The robot's simulation model is displayed in real time, which is used to remotely monitor the robot's dynamics and capture feedback from the user's robotic hand adjustment.

Zone 3 is the display area of sensor data of robotic hand, where the data of data cable connected to the lower computer is displayed in real time. This area shows the joint data of robotic hand sent from the lower computer to the host, where the user knows the actual joint data from robotic hand.

Zone 4 is the slider control area of bending degree of robotic hand finger, which lists the controllable joints of all robots. The adjustment slider is used by the user to change the value of the joint. The host receives the joint data sent by the web interface, and sends the data through the serial port to the lower computer, so as to control the robotic hand to make corresponding changes. As shown in Fig 6.

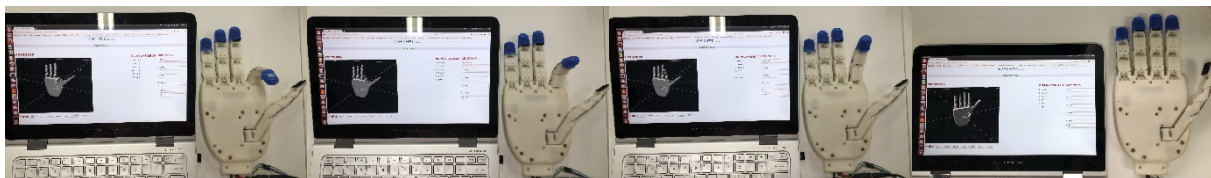


Figure 6. Robotic hand control UI

Zone 5 is a shortcut button area for specific gestures of robotic hand, providing shortcut buttons for several common gestures. When the user presses the button, the robotic hand will make the corresponding posture. As shown in Fig. 7.

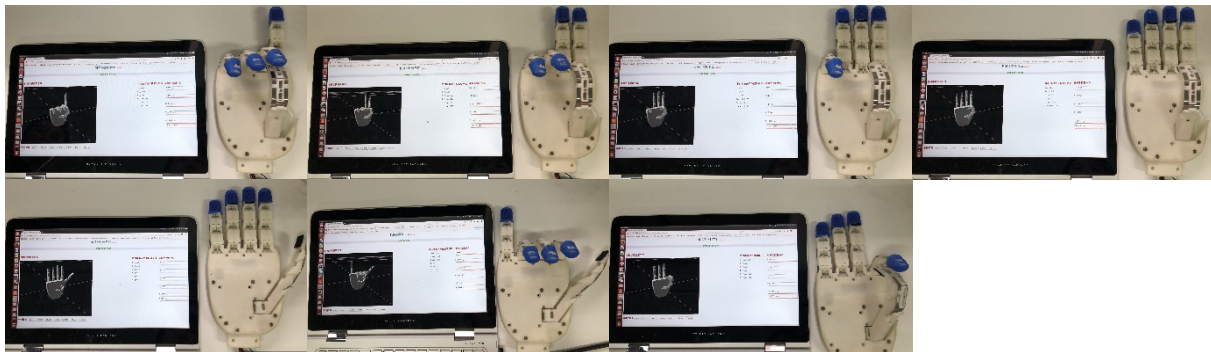


Figure 7. The gestures of the shortcut button area

The system can be used on any mobile device or PC with regular browsers such as IE, Firefox, Safari and Chrome. The Safari browser that comes with the iPhone 6 is used to run the experiment, as shown in Fig. 8.

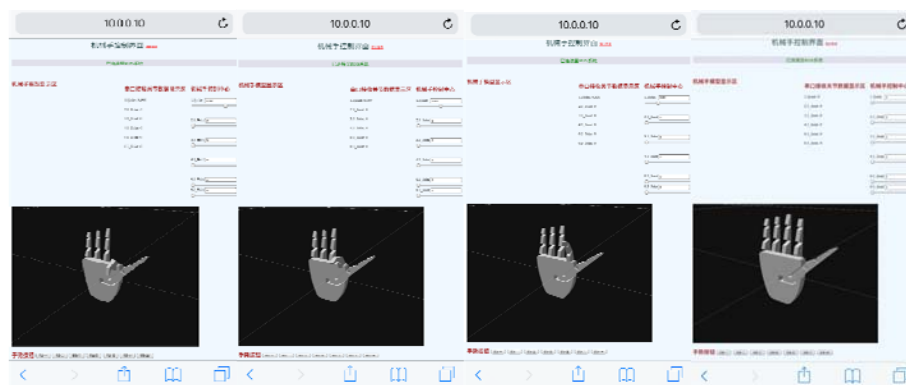


Figure 8. Experiment diagram with mobile phone

According to the experimental results, the control of the bending degree of each finger of the robotic hand is achieved by the system. The update frequency of 3D simulation model meets the real-time requirements. The joint data from the robotic hand sensor in real time is displayed in the serial data display area. The user can press the button in the gesture button area to update slider value of the joint, and the robotic hand immediately puts out the corresponding gesture.

The robot control interface can be used on the PC side and the mobile side, achieving cross-platform ability. The client does not need to install a complex software environment, which meets the versatility and feasibility requirements of system.

5. Conclusion

The popular Html and JS technologies and the JS library supported by the ROS system are adopted in this system to realize the robot web control system. Compared with the traditional robot control system, due to the separation of the client and the server, the system can be used without the configuration of complicated software and hardware environment, which is easy to use and suitable for widespread popularization. Due to the platform independence of only web technology, and the perfect compatibility with computers and mobile devices, the system has cross-platform features, which increases the geographical situation to use robot control systems.

The current web-based robot control system is still in the emerging stage. With the advancement of science and technology, efficient web technologies will emerge one after another, and the robot operating system is constantly updating new versions. In the near future, robot control systems designed by new technologies will contain more functions. For example, the front-end technology can

be combined with human tracking technology, enabling the camera to control the robot by tracking the human body. For example, a JS library for speech recognition could be developed to realize the voice control of robot. How to enable users to control robots more intelligently using a simpler software environment is the research direction that promotes the civilianization and generalization of robot control systems.

Acknowledgments

This work is supported by Municipal Science and Technology Commission of Shanghai (No.15411953500).

References

- [1] Tianmiao Wang, and Yong Tao, Current status and industrialization development strategy of industrial robot technology in China. *Journal of Mechanical Engineering*, 2014, 50 (09): 1-13.
- [2] Zichang Lai, Bin Yang. Visual Real-time Control to Remotely Control Mobile Robotic hand. *Application of Microcontrollers & Embedded Systems*, 2014, 14 (12): 14-17.
- [3] Hasrul Che Shamsudin, Mohammad Afif Ayob, Wan Nurshazwani Wan Zakaria, Mohamad Fauzi Zakaria. Modeling and Simulation for One Leg of Quadruped Robot Trajectory. Hong Kong Society of Mechanical Engineers (HKSME). *Proceedings of 2015 International Conference on Mechanical Engineering and Automation Science (ICMEAS 2015)* [C]. Hong Kong Society of Mechanical Engineers (HKSME), 2015: 6.
- [4] Alexey Demin, Yuxing Dai, Xin Li, Wanshan Wang. Kinematics simulation of robot based on HTML5 and WebGL Under 3D Environment. *Journal of Northeastern University (Natural Science)*, 2014, 35 (04): 564- 568.
- [5] Zhengwan Cao, Xueliang Ping, Shenglong Chen, Yi Jiang. Method and Research on Construction Method of ROS-based robot model. *Combined Machine Tool & Automatic Processing Technology*, 2015 (08): 51-54.
- [6] Shaonan Meng, Yanbing Liang, Heng Shi. Motion Planning of Six DOF Robotic hand Based on ROS Platform. *Journal of Shanghai Jiaotong University*, 2016, 50 (S1): 94-97.
- [7] Liangliang Bai, Xueliang Ping, Shenglong Chen, Yi Jiang. Design and Realization of Distributed Mobile Robot Control System. *Mechanical Design & Manufacturing*, 2015 (10): 180-183.
- [8] Zaichang Xiao, Wenhui Yang, Bing Liu. Real-time technology based on WebSocket. *Computer and Telecom*, 2012 (12): 40-42.
- [9] Hao Wang, Feng Tian, Wenjun Zhang. Design and Realization of Interactive Platform Based on WebGL. *Electronic Measurement Technology*, 2015, 38 (08): 119-122+128.
- [10] J.Y. Xia, B.J. Xiao, Dan Li, K.R. Wang. Interactive WebGL-based 3D visualizations for EAST experiment. *Fusion Engineering and Design*, 2016, 112.
- [11] Yana Ma, Huanyan Qian, Yamin Sun. Apache Module Analysis and Service Customization. *Journal of Nanjing University of Science and Technology (Natural Science)*, 2002 (02): 171-174.
- [12] Huaishan Qiu, Qunxiong Zhu, Chenghui Shi. Design and Realization of Networked Vehicle Management System Based on PHP. *Journal of Beijing University of Chemical Technology (Natural Science Edition)*, 2003 (06): 72-76.