Automated Puppetry—Robo-Puppet

Article · January 2015

DOI: 10.1007/978-81-322-2229-3_49

Aravind Ma
Indian Institute of Science

Automated Puppetry—Robo-Puppet©

M.A. Aravind, N.S. Dinesh, Nori Chalapathi Rao and P. Ram Charan

Abstract Many types of puppet theatres and automation systems exist. The paper presents a fully automated marionette theatre. The system presented consists of robot manipulators each with sixteen degree freedom which can be adapted to different degree of freedom. The manipulators are linked through a WIFI network to a master controller (Linux PC). Software developed provides for the user interface at the front end and a back-end module which synchronizes motion data, audio and video streams in real-time. The theater provides for a puppet motion capture in three modes. The motion programming modes are—direct human body motion capture, motion programming through joystick and use of motion library modules. The user interface designed provides for an easy and efficient way for the puppeteer to record the motion in sync with the audio and video. The theater can support simultaneous operation of six marionettes with automated stage lighting and curtain (Patented as Robo Puppet).

Keywords Robo puppet · Manipulator · Marionette · Puppet theatre

1 Introduction

Puppetry is one of the most ancient forms of entertainment in the world. Over the centuries, puppetry has developed into a powerful media of communication. Traditionally, India has a rich heritage of puppetry. Puppeteers from different regions conduct their shows in their colloquial languages, limiting the effectiveness of communication to the audience who know the language. The objective of the present work is to develop an automated Puppet Theater system which derives

M.A. Aravind (✉) · N.S. Dinesh · N.C. Rao · P.R. Charan
DESE, Indian Institute of Science, Bangalore, India
e-mail: aravindma1990@gmail.com

A. Chakrabarti (ed.), *ICoRD'15 – Research into Design Across Boundaries Volume 2*,
Smart Innovation, Systems and Technologies 35, DOI 10.1007/978-81-322-2229-3_49

benefits from modern technology. A conscious effort has been put to realize a system which adapts to conventional puppets without modifying the traditional construction of the puppets.

There are many forms of puppets and they differ in their construction and manipulation techniques. Most types of puppets in use in India today fall into five broad categories [1]: hand puppets, rod puppets, marionettes, rod-string puppets, and shadow puppets. We have chosen stringed puppet (Marionette) for our study. From literature we find that Marionettes have strings ranging from 2 to 120 depending upon the complexity of motion required. It is observed that most of the puppet manipulations can be achieved with 10 strings and the puppets with more strings are considered special puppets and may need highly skilled puppeteer to manipulate them. Therefore we have presently considered a marionette with up to 12 strings for the automation. A robot manipulator with a provision to manipulate up to 16 strings has been developed to manipulate the puppet. The puppet theater developed by us is able to accommodate 5 robots which can simultaneously manipulate 5 puppets, playing the part of 5 puppeteers. The system has provision for the automated stage lighting, curtain controls, video effects and possible special effects.

The motions of the puppets are precise sequence of choreographed motions as decided by the puppeteer and importantly all the motions have to be in synchronization with the audio/video used in the show. This calls for a real-time audio/video and motion synchronization capability for the electronic system used for the control of the entire show. Effective programming of the puppet theater is the key to the success of the technology envisaged.

Many ways of motion capturing for puppetry has been explored earlier [2, 3–4]. Most of them use elaborate methods of using markers and multiple angle capture cameras. The proposed technology provides for cost effective means of automation and helps create sophisticated shows with puppets built with traditional technology. The developed system may emerge as a new form of art for the artists to exploit.

2 Puppet Construction

The puppet theatre was constructed for marionettes. These marionettes were constructed out of papier-mâché so that they are of light weight. The typical weight of the constructed marionette was around 200 g. This was an important parameter to decide the specifications of the actuators and construction of the manipulator. The dimensions of the puppet were kept to the general standards of a marionette puppet. Figure 1 shows the puppet construction.

Fig. 1 Puppet stage with reference

3 Manipulator Design

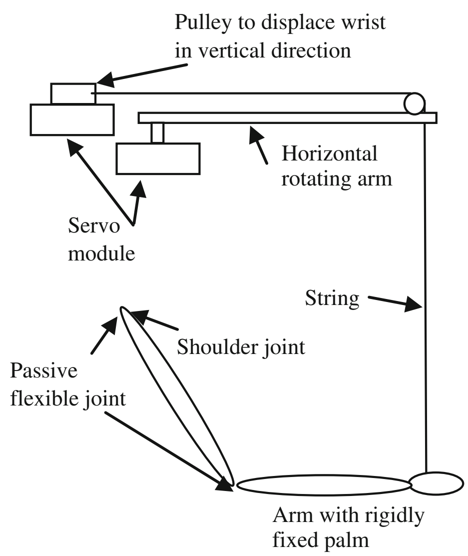
The puppet manipulator was designed keeping in mind the weight and speed consideration of the marionette. A typical puppet theatre can house around 4–5 puppets at a time. The frame of reference of the whole puppet systems is as shown in Fig. 2.

Movements of different limbs of puppet:

The puppet is given 9° decoupled of motion. The entire puppet can move in the X direction from one end of the stage to the other. The wrist is given two degrees motion in X and Y as shown in Fig. 3. While only two degree freedom is given to the wrist depending upon the position of the Y axis, the loose passive joints of the arm will take different positions due to gravity and give the illusion of more than two axes motion. The motion of the wrist is the largest in the puppet. For the puppet designed by us it is 30 cm from end to end in Y axis. The other motions that were given are the following: The leg is given one degree motion along Y direction. The entire body is movable in Y direction and rotatable around puppets vertical axis. The actuation is done by DC servo motors. These servo motors were housed in a manipulator along with an assembly of gears and pulleys to bring about the various effects of the system. Servo motor specifications were chosen depending on the load characteristics of the limbs of the puppet. Other similar designs existed like [5, 6], but these systems have a static manipulator with pulleys and gears. In our system

Fig. 2 Puppet stage with reference

Fig. 3 Arrangement for hand motion with two axes decoupled motion



the whole manipulator could rotate about an axis as well as laterally on a rail mechanism. This gave two extra degrees of freedom compared to other designs.

The commercially available servos generate a maximum of 180° of total motion. Therefore we used gear train in order to achieve small size of the pulley and get amplified motion (1.5 turns). In our prototype system, the puppets weighed 200 g with each limb weighing roughly 10 g. We could actuate the arm and legs with a

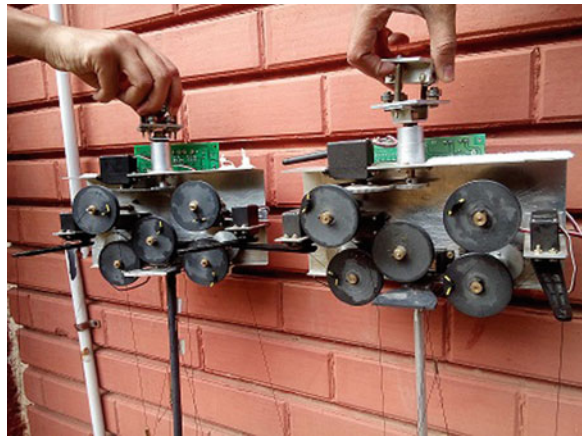


Fig. 4 Robot manipulator

required speed of 50 cm/s with servos which provided 34 N-cm torque. The entire puppet body was actuated with a servo of 100 N-cm at 16 cm/s speed. The manipulator construction is shown in Fig. 4.

4 Controller for the Manipulator

The manipulator controller is a system which receives motion command stream from a master controller and controls the servo motors accordingly. In literature [7] we find some controllers are built with wireless trans-receivers working in master slave mode. But in the present work (Fig. 5) the master controller and many manipulator controllers are networked using WIFI link. This provides for real-time streaming of motion, audio and video on a single network. The master controller is a Linux PC which synchronizes all the three streams within 30 ms. The synchronizing is done at

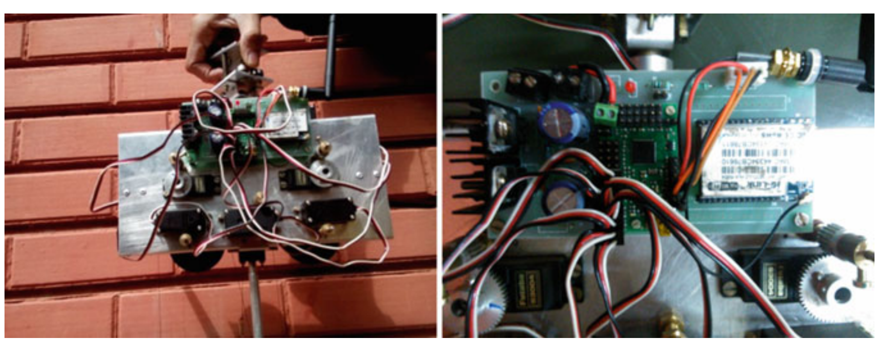


Fig. 5 Controller for manipulator

the video frame rate of 30 FPS. The manipulator controller is an ARM cortex based system which can control up to 16 servo motors. There is provision to monitor the health of each controller in real time and transmission error detection and correction.

There is a stage management module (SMM) which is also WIFI linked to the master controller. The SMM is used for the stage lighting, curtain control and special effects. The SMM works like a Programmable logic controller which works in synch with the puppet motion.

5 Software Design

The software has following two main modules: Front-end module and the back-end module.

The front end module forms the Graphic User Interface (GUI) which interfaces the user to the back end functions for the real time controlling of the puppets.

5.1 GUI Design

A GUI based program was created to interface to all the puppets. Five main modes defined the whole system: Test mode, Record mode, Playback mode, Edit mode, Real-Time mode.

5.1.1 Test Mode

This mode is designed to test all the functionalities of the puppet (Fig. 6). This is also used to do any calibrations that might be necessary for the initial setup of the puppet. Some pre-defined motion patterns could be played through selected axis for testing. Provision is made for testing and calibrating joysticks and Microsoft Kinect which form input devices for motion data.

5.1.2 Record Mode

Record mode is used to generate motion and stage management data in sync with recorded audio/video stream. Three types of inputs were used for capturing motion data.

Microsoft Kinect

Kinect is a 3D motion capture gaming device. Many open source tools allow the developer to access the 3D view of the Kinect. Human motion can be detected

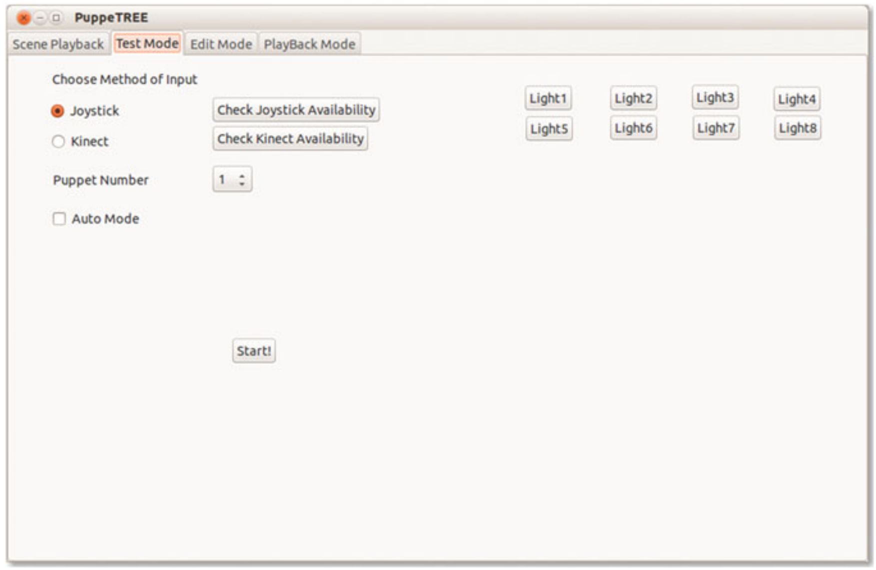


Fig. 6 Test mode

through some image processing and various points can be accessed on the skeletal frame of the human. Motion capturing techniques were used to extract the user skeleton from the images and apply the required positions onto the puppet. These points were scaled and mapped to the joints of the puppet respectively. Thus generated motion data is transmitted in real time to various puppets. During our studies, it was found that the motion data had noise. Averaging techniques were used on the values sent, to give a smooth transition between values as shown in Fig. 7. This reduced a lot of jitters that came from image processing of the user's gestures from Kinect. Also, the system achieved minimum of one frame delay from the human body motion to the puppet motion.

It was required to scale the human gestures to the size of a puppet. In the case of Kinect, the size of the body captured on the camera depends on how far the user is from it. Therefore, direct mapping of limb positions cannot be done in the Cartesian co-ordinates. The problem was overcome by mapping the angles between the limbs to the puppet's respective limbs. With this, the motion data is made insensitive to the position of the user within play area of the camera.

Joysticks

Two 2-axis joysticks can be interfaced to this system. Provision is made in GUI to map virtually, the joystick axes to any limbs of the puppet as the user requires.

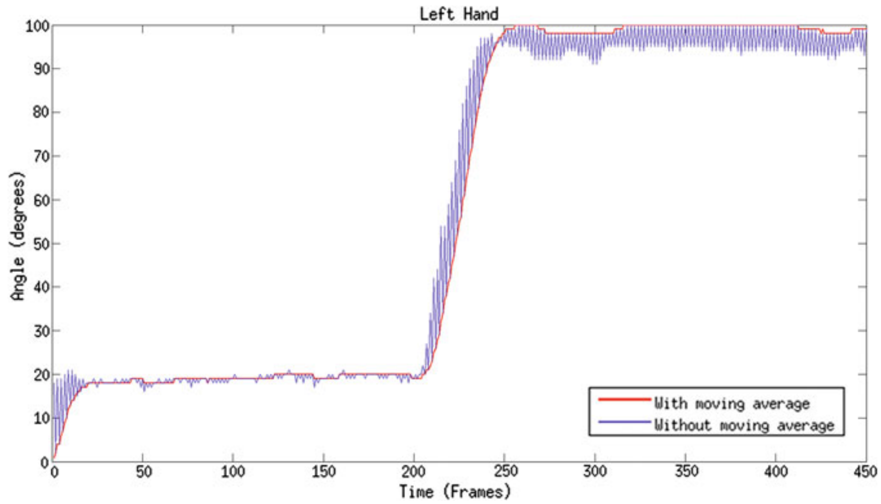


Fig. 7 Moving average for left hand (angles vs. time (frames))

Motion Library Modules

In order to simplify motion programming, pre-defined motion library files were created for general moves like walking, lifting and waving hands, sitting down or standing up etc. These can be directly used for the motion programming. The motion file is created compatible to Microsoft excel format which provides for easy file manipulation.

Playback Mode

This mode provides an interface for the user to playback an earlier recording (Fig. 8). The interface provides for selecting the start and end time as well as the number of puppets to be played simultaneously.

5.1.3 Edit Mode

This mode allows the user to edit the pre-recorded motion file (Fig. 9). Provisions have been made to copy motions of one puppet to another puppet. User can cut and copy motion from one part of the play to any other puppet in any other part of the play. User can also mirror this motion while copying with the enable of a button. This brings about complete sync between puppets. This provision helps puppeteer easily sync multiple puppet motions which otherwise needs highly skilled puppeteers to achieve the same in manual puppetry. This feature of cutting copying the motion and mirroring of motion help create dance sequences effectively and easily.

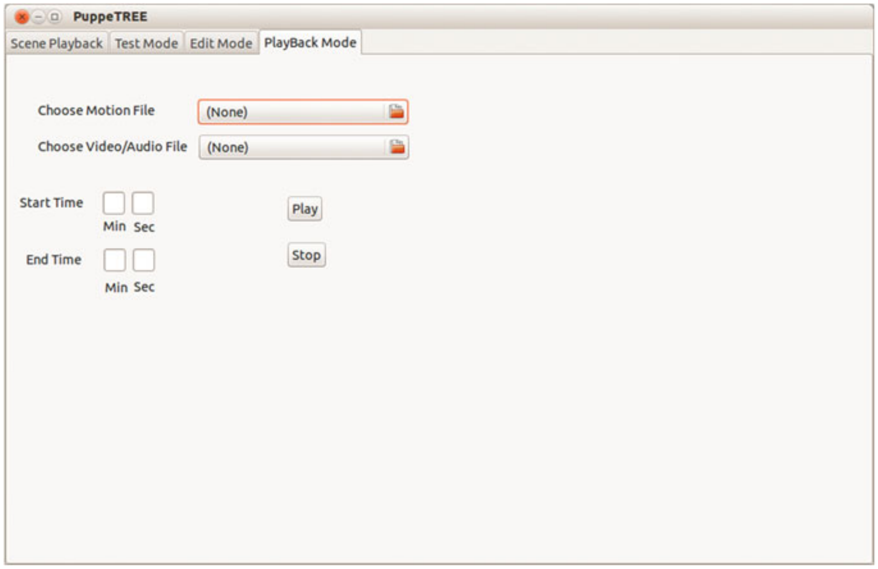


Fig. 8 Playback mode

5.1.4 Real-Time Mode

In this mode, four puppeteers can manipulate four different puppets in real time simultaneously using Kinect. An audio/video could be played on the screen behind the puppet. This enables the user to interact directly with the audience and respond to them. This is a novel approach to interactive puppetry.

5.2 Synchronization of Motion Data with Audio/Video

In order to synchronize the motion to the audio/video, it was required to transmit one frame of motion data, which consists of data for all axes, within one frame of the video. We have designed the system for 30 FPS video stream where, duration of one frame is 33 ms.

A new media player was designed for this purpose. The media player synced a motion frame with that of an audio/video frame. For this purpose, a file containing the motion data was created. When the video codec set up the next frame on the video buffer, the corresponding motion data for that frame was extracted from the motion data file and placed on the data buffer connected to the WIFI protocol. This would then transmit the data to the controller on the manipulator just as the video frame would come up on the screen. The audio was synced to this video frame just like any other media player.

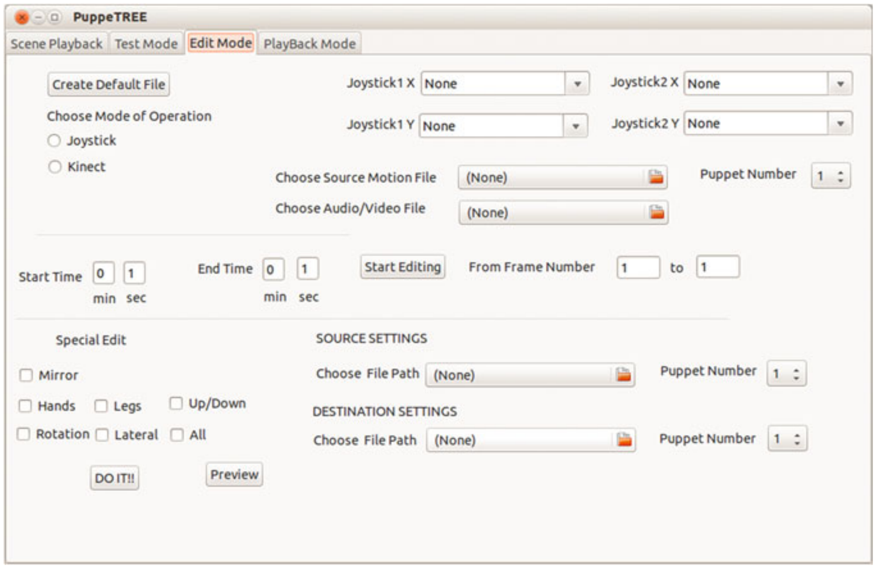


Fig. 9 Edit mode

The setting up of the video and audio frame took an overhead of 15 ms on the software. The WIFI module had a round trip ping time of around 6 ms for sending one frame of motion data for 5 puppets. This still provides for an extra buffer time of around 8 ms which was reserved for re-transmissions, health monitoring of manipulators and future use.

Provisions were also made to fast forward and rewind the audio/video/motion data. This way the play could be reset to start from any position with respect to the video. This enabled the user to fine tune a particular move by quickly replaying the same sequence on the press of a button. A resolution of 30 ms was achieved for editing the motion file.

6 System Integration

The Robo puppet system architecture is shown in the block diagram in Fig. 10. It consists of a master controller, a Linux machine which interacts with several slave puppet manipulators. The communication happens through Wi-Fi. Each puppet manipulator consists of electromechanical actuators with ARM processor based controller. Each slave has a unique ID through which it will communicate with the master. The entire communication between master and slaves takes care of real-time synchronization of motion and audio/video streams. The master controller also gets inputs from Kinect, joysticks and keyboard which are used for programming the puppets.

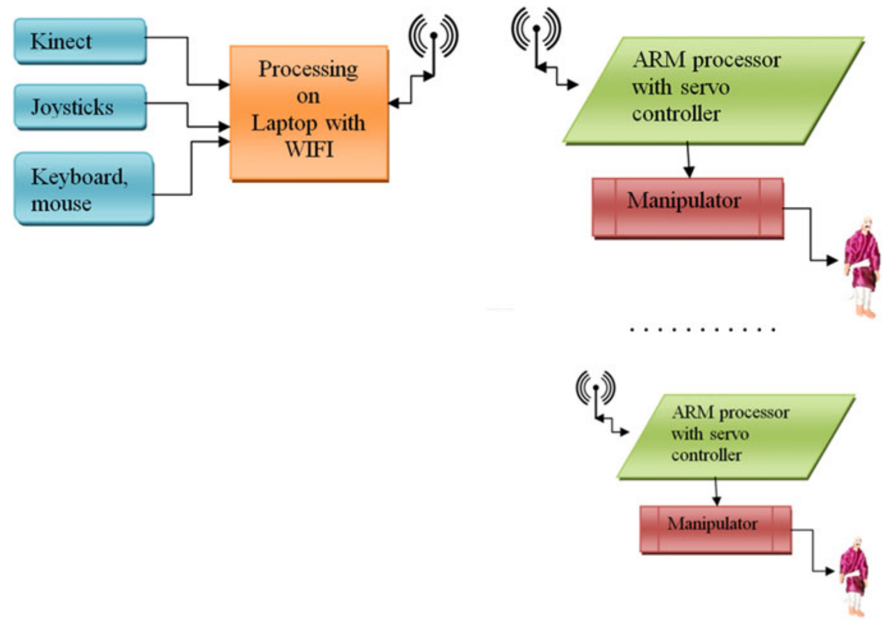


Fig. 10 System Integration Flowchart

7 Conclusion

This work presents a technology which automates Marionette Theater. Papier-mâché puppets were constructed. A robot manipulator was designed based on the requirements for effective marionette manipulations. An automated puppet theatre was built with provision to play 5 puppets simultaneously with programmable stage management. A user friendly GUI was created to capture, edit, record and playback motion data in sync with audio/video file. Some of the highlights of this technology are:

- The puppet show can be dubbed into any language making it transcend language barrier. The audio track is made up of key frames, which are synchronized with the video and the motion streams. A provision has been made to substitute the existing audio track with a new audio sound track of different language. This enables the puppeteer to present the show in any language.
- It does not need highly skilled people to choreograph the puppets.
  As mentioned in Sect. 5.1.2, the system has been designed to program the puppet in three different modes—Keyboard, body motion, pre-defined motion libraries. In conventional puppetry, puppeteer will never get to see his own actions through the marionettes, whereas in the proposed system, the puppeteer gets direct visual feedback while programming through any of the above modes. This enables even a novice puppeteer to choreograph the puppets reasonably well. Certainly an experienced puppeteer can make his shows more effective.

- Provisions of cut and paste and mirroring will help a person to program the entire show single handedly.
  A character in a show may have a distinct style of walking which can be programmed once using any of the above modes. The motion thus created, can be directly used anywhere within the show without requiring programming. Mirroring the motion sequence will add to the aesthetics of dance sequences or other general choreography. This feature provides for ease of programming and creation of entire show single handedly.
- Puppet show gives consistent performance every time.
  An effective control system and mechanical design ensures that, once choreographed, the system will perform consistently in each show as against a conventional marionette show where the outcome is dependent on the puppeteers experience and human error. This enables us to archive entire puppet shows which can be recreated anytime at will.
- Traditional marionettes can be directly used without modifications.
  Unlike other puppet systems like the Animatronics where the actuators and the electronics are integrated inside the puppet, the present system has been designed consciously to work with conventional marionettes which are built by traditional techniques. Therefore, this technology can be used by traditional puppeteers without any technical knowledge of robotics or electronics.

References

1. Currell, D.: Puppets and Puppet Theatre. Crowood Press, Wiltshire, UK (1999)
2. Moeslund, T., Granum, E.: A survey of computer vision-based human motion capture. Comput. Vis. Image Underst. **81**, 231–268 (2001)
3. Nguyen, K.D., Chen, I.M., Yeo, S.H., Duh, B.L.: Motion control of a robotic puppet through a hybrid motion capture device. In: IEEE Conference on Automation Science and Engineering Scottsdale, pp. 754–758 (2007)
4. Yamane, K., Hodgins, J.K., Brown, H.B.: Controlling a marionette with human motion capture data. In: IEEE International Conference on Robotics and Automation, pp. 3834–3841 (2003)
5. Chen, I.M., Xing, S., Yeo, S.H.: Robotic marionette system: from mechatronic design to manipulation. IEEE International Conference on Robotics and Biomimetics, pp. 228–233 (2005)
6. Hu, J.S., Wang, J.J., Sun, G.C.: Self-balancing control and manipulation of a glove puppet robot on a two-wheel mobile platform. In: International Conference on Intelligent Robots and Systems, pp. 424–425 (2009)
7. Setyawan, L., Kesaulja, L.M., Darma, J., Kuantama, E.: Hand-puppet Robot Control. IEEE (2011)