Chinese Shadow Puppetry with an Interactive Interface Using the Kinect Sensor

Hui Zhang[1,2], Yuhao Song[1], Zhuo Chen[1], Ji Cai[1], and Ke Lu[1]

[1] Dept. of Computer Science, United International College, 28, Jinfeng Road, Tangjiawan, Zhuhai, Guangdong, China
[2] Shenzhen Key Lab of Intelligent Media and Speech, PKU-HKUST Shenzhen Hong Kong Institution, Shenzhen, China

Abstract. This paper addresses the problem of using body gestures to control the Chinese shadow puppets with the Microsoft Kinect sensor. By analyzing the motion of the actors in the Chinese famous drama, Wusong Fights the Tiger, we propose a general framework for controlling two shadow puppets, a human model and an animal model. A performer can conduct simple actions such as turning the head, stretching the arms or kicking the legs. However, it is more difficult for a normal performer to simulate more complicated movements, for example, back flips and splits. Therefore we define some special postures to represent these difficult movements. Besides, in order to be compatible with the Chinese drama style, we use water color to paint the background scenery and the foreground characters. We show some preliminary results which demonstrate the effectiveness of this work.

Keywords: Kinect, shadow puppetry, gesture.

1 Introduction

The Chinese shadow play, or shadow puppetry, is an ancient form of storytelling in which sticks and flat puppets are manipulated behind an illuminated background to create moving pictures [1]. It is a popular means of entertainment being regarded as a predecessor of animation and movies. The shadow puppet is a cut-out figure operated by the puppeteer between a light source and a translucent screen. By moving both the puppets and the light source, various effects can be achieved. A talented puppeteer can make the figures perform all kinds of actions, such as walk, dance, fight, nod and laugh. However, it is not that easy for a normal user to control the movements of the puppets. A more friendly interface for them is to manipulate directly using their body gestures [2] so that the puppets can mimic the performers' motion. Although action and gesture recognition [3][4][5][6][7] is an extensively studied research field, controlling the movements of puppets is still a challenging problem because of the diversity of the movements of the puppets and the mapping between performers and puppets.

Recently, a lot of works have been carried out on digital puppetry [8]. The first live animated computer graphics puppetry is probably the Waldos [9], which is

A. Fusiello et al. (Eds.): ECCV 2012 Ws/Demos, Part I, LNCS 7583, pp. 352–361, 2012.

manipulated through a mechanical arm controlled by a puppeteer. Puppeteers could also use multimodal systems, with data gloves, joysticks, motion capture and midi devices, to control the digital puppets [10]. Shin proposed to transfer motion capture data to animated characters using the notion of dynamic importance of an end effecter [11].

However, all of these mentioned works are based on 3D characters. The solution of animation performed by two dimensional puppets appears only recently. In [12], Hsu et al. introduced a motion planning technique which automatically generates the animation of 2D puppets. Tan et al. presented a method for interactive animation of 2D shadow play puppets by real-time visual simulating the shadow using texture mapping, blending techniques, and lighting and blurring effects [13]. Inspired by traditional Chinese shadow puppetry, the project ShadowStory [14] allows the users to design their own puppets with a tablet PC and animate them with orientation sensors. In [15], a 2D shape deformation of the triangulated cartoon is driven by its skeleton and the animation can be obtained by retargeting the skeleton joints to the shape.

With the emergence of Microsoft Kinect, the Xbox 360 video game console Kinect, people start to use it to control the puppet. Kinect is composed of a depth sensor, a color camera sensor and a four-element microphone array [16]. Through a natural user interface using gestures and spoken commands, it enables users to control and interact with the computer without touching a game controller. Kinect has the capability to track the skeleton image of one or two people moving within the Kinect field of view for gesture-driven applications.

People use different ways to control puppets. Gobeille and Watson developed a prototype of directly mapping the movement of one hand to the motion of the puppetry bird mouth [17]. Walther controlled a 3D puppet's motion only by the movements of two hands [18]. Instead of only one puppeteer controlling one puppet, Boyle proposed a collaborative interface of two puppeteers using their body movements [19]. However, all these state-of-the-art works only provide preliminary results for controlling puppets. Lots of puppet movements could not be easily mapped to the gestures of an actor, such as splits and somersaults, which are quite normal in Chinese Kungfu Shadow Play. Therefore we study the movie in which the puppeteer manipulates the shadow puppets and then recognize the hand and body movements of the actors by making use of Kinect. Furthermore, in addition to the body movement, the shadow puppet would have some subtle face expressions, such as moving its lower jaw and eyebrow when it speaks, moving eyeballs when it thinks, etc. Therefore we propose to control the simple movements of the shadow puppets directly by users' gestures and trigger the animation of complicated puppet movements by pre-defined user postures or by voice. Besides, in order to be compatible with the Chinese drama style, we use water color to paint the whole scenery.

The remainder of the paper is organized as follows. Section 2 describes details of the puppeteer gestures and how to control the puppets. Section 3 introduces implementation details of the proposed technique. Section 4 presents the experimental results, followed by discussions and conclusions in Section 5.

2 Gesture Control

Kinect scans the user's skeleton for further interactions with the computers. Therefore, it is easy to directly map the human gesture to the puppet gesture. However, this is only true for simple human movements, such as moving legs, jumping. Those complicated actions, e.g., back flips and splits, could not be performed by the normal user.

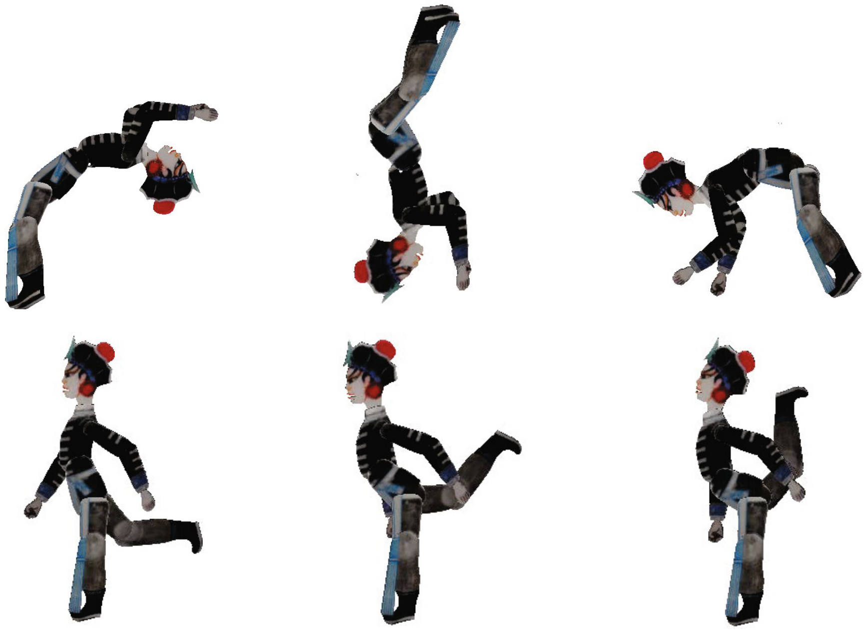


Fig. 1. The backfilp (the first row) and the standing splits (the second row) gesture of the puppet Wusong

Furthermore, the shadow puppet is of two dimensions whose movement is in the screen plane, while the skeleton and its motion are three dimensional. Considering the difficulty of transforming 3D behaviors of the skeleton into the movement and rotation of 2D puppets, we prefer using the data-driven method instead of the mechanism method (i.e. the physical model). For example, the behavior of forearms is transformed into rotation angles by Kinect. Those series of angles are read time by time and mapped into the user-controlled character in program. Therefore for the definition of the action movement, our solution is to use event listener to trigger the character movement, such as walk, splits and rolling.

According to the characteristic of motion of the Wusong and the Tiger puppets, we found that they require different means of representation. We classified them as a human model and an animal model, and introduced different ways for controlling their puppets in below.

2.1 Actions of Human Models

As the structure of the Wusong puppet is similar to the user, it is easy to directly map the Kinect skeleton to its model, parts by parts [2]. We propose to use the above mentioned data-driven method to implement those complicated gestures. However, since the range of Kinect detection is limited, a displacement might not be captured and we have to define the puppet walking by making use of pre-defined gestures.

– *Back flips.* When the user hands stretch to up-backward, the animation of back flips will be triggered. The top row of figure 1 shows the screenshots of the puppet Wusong's back flips.
– *Standing splits.* When the user rising the calf of one leg backward with a $\geq 90^o$ angle to the thigh, the back flips animation will be triggered. The bottom row of figure 1 shows the screenshots of the puppet Wusong's standing splits.
– *Walking.* When the user's left foot steps to left and has a certain distance to the right foot, the walking animation of the Wusong model would be triggered. Figure 2 shows the screenshots of the puppet Wusong's walking animation.

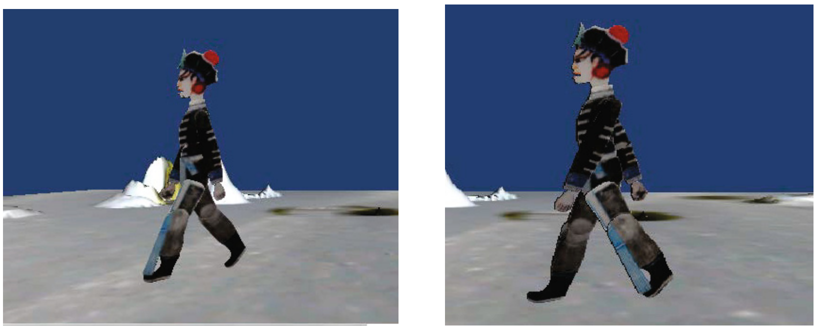


Fig. 2. The puppet Wusong walks

Additionally, for subtle movements such as the opening of the mouse, we use Kinect audio control. The performer's voice drives the mouse movement of the puppet Wusong (see figure 3 for details).

2.2 Actions of Animal Models

Let's use the Tiger puppet as an example of an animal model. Since animal models and human models are very different, we cannot directly map the Kinect scanned skeleton to the corresponding Tiger puppet. Especially in the shadow play, Wusong Fights the Tiger, most of the action of the Tiger has relatively large range of movements, such as somersaults, rolling. These actions are difficult to mimic by human. Therefore we have proposed three methods to solve the problems.

(a) (b)

Fig. 3. The mouse movement of the puppet Wusong is triggered by the voice of the performer. (a) The puppet Wusong's mouth is closed. (b) The puppet Wusong's mouth is opened.

The first solution is to define a number of gestures to achieve these movements of the Tiger. The advantage of this approach is that the animation can be very real to mimic the tiger's action, especially for some subtle movements. The figure 4 shows three images in the animation of the Tiger's idle state.

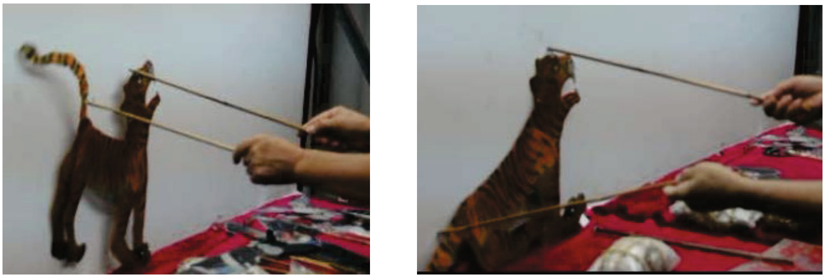
(a) (b) (c)

Fig. 4. The animation of the Tiger's idle state

However, the drawback of this method is obvious. These gestures can only be applicable for the Tiger in this shadow play, and the more the number of gestures, the longer the cycle of producing the animation.

The second possible solution is to bind the tiger model with two control points. These two control points are the points which the puppeteer manipulates when they play the puppet Tiger. Figure 5.(a) shows the location of the control points. As the control points move, various parts of the Tiger will move due to the inertia and the traction. Kinect only gets the skeleton position of the two hands in order to manipulate the shadow play and it will be easy for players. However, relying solely on the two control points cannot completely achieve all the tiger actions, such as the Tiger squatting on the ground. The Tiger's legs will bend after colliding with the underground (see figure 5.(b) for detail).

The third solution that we currently conceive is the best one. It is similar to the human model mapping in the first approach. The difference is that we map the joints on left and right part of the human skeleton to those on the Tiger's front and rear. Figure 6 illustrates the joints on the human skeleton captured by Kinect and their correspondences on the tiger model.



(a) (b)

Fig. 5. (a) The puppeteer uses two sticks to control the movement of the Tiger, through two control points. (b) The Tiger squatting on the ground. The Tiger's legs will bend after colliding with the underground.

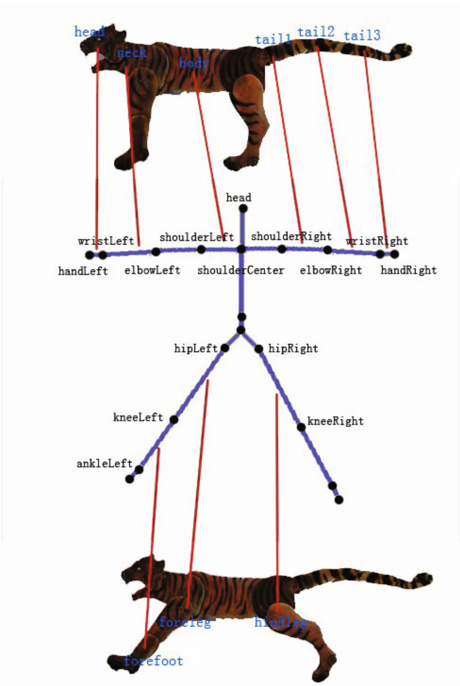


Fig. 6. The joints on the human skeleton scanned by Kinect are mapped to their corresponding joints on the tiger model

2.3 Gesture Mapping

To summarize, there are some important gestures to be represented in the Chinese Kungfu drama. We shows the gesture mapping list for the Tiger and Wusong puppet in table 1.

Table 1. Gesture mapping list for (a) the Tiger and (b) Wusong. The postures captured will trigger the corresponding puppet actions.

Actions of the puppet	Postures to be recognized
To turn around	Nod.
To move left	Moving the leg one step to right.
To move right	Moving the leg one step to left.
To roll forward	Preparatory action of rolling forward.
To roll backward	Preparatory action of rolling backward.
Legs in the air	Surrender

(a)

Actions of the puppet	Postures to be recognized
To walk left	Moving the leg one step to right.
To walk right	Moving the leg one step to left.
To roll forward	Preparatory action of rolling forward.
To roll backward	Preparatory action of rolling backward.
Backward somersault	Arms stretching up-backward.
Front somersault	Arms stretching up-forward.
Splits	Splitting the two legs with an angle $\geq 60^o$.
Standing splits	Rising the calf of one leg backward, with an angle $\geq 90^o$ to the thigh.
Single-Handspring	Putting two hands on one foot while keeping the legs straight.

(b)

3 Implementation

In our system we use the official Microsoft Windows SDK for posture detection instead of the other cracked SDK such as the very popular OpenNI. This is due to the reason that the Windows SDK provides the best recognition accuracy and furthermore, the Windows SDK supports audio driven interaction. Therefore different from OpenNI, we can use it for controlling the subtle face expression of the shadow puppets. By integrating SDK and the sensor Kinect into Unity3D, our method can control the 2D shadow puppet through a multimodal interface with depth, rgd and voice inputs.

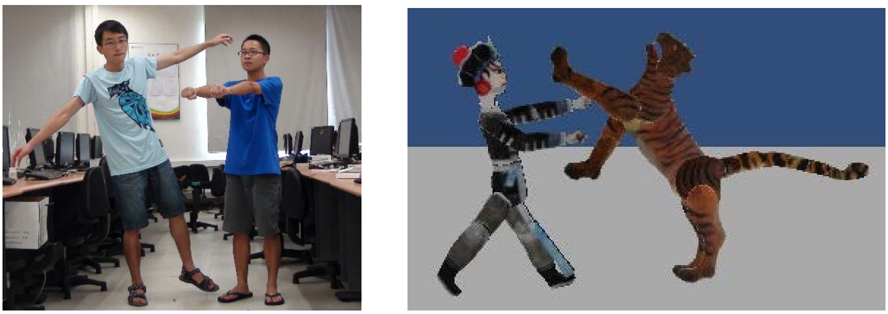
We designed the 3D scenery by Autodest 3DMax and put the puppets model into it. To generate a more traditional Chinese style in the screen, we render the final result with the method of watercolor painting in [20]. By making use of Shading language, we implemented several steps of processing such as blurring, edge strokes, sharpening. Some extra elements were also added into the scene, for example the waterfall and the wave effect, generated with the particle system in the Unity engine. By controlling the emission, the size and the energy of the particles, effects such as the water splash and the waterfall are imitated veritably. Additionally the texture of the water-drop is refined for a higher reality.

4 Experimental Results

In our experiment, Wusong and the tiger of the famous Chinese novel "Outlaws of the Marsh" are the two puppets to be controlled. We took separate pictures of their body parts as the textures and create their models by assembling them respectively in 3D Max. Figure 7 shows the separate parts of the tiger and figure 4 shows the Tiger puppet.

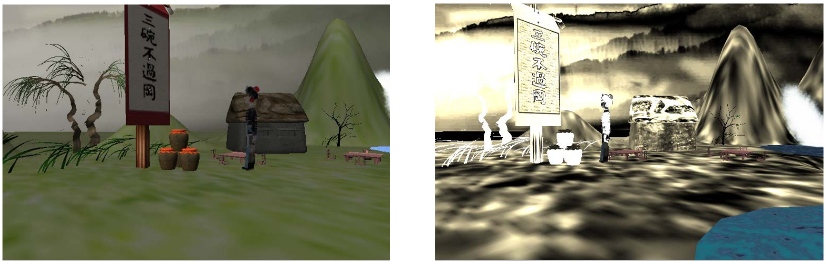
Fig. 7. The separate parts of the tiger

We detected Wusong and the Tiger with a single Kinect at the same time. However, since a human and an animal has different structures of the skeleton, we have to use different mapping strategies for controlling them. The Wusong puppet was mapped with a normal human skeleton and the Tiger puppet was mapped with the skeleton shown in figure 6.(b). Figure 8.(a) shows that two players are posing in front of Kinect and figure 8.(b) shows that their gestures are mapped to the Wusong and the Tiger puppets, respectively.



(a) (b)

Fig. 8. (a) Two players are captured by Kinect. (b) The players' gestures are mapped to the Wusong and the Tiger puppets, respectively.



(a) (b)

Fig. 9. (a) One of the scenes in the shadow display. (b) The water colored version of the scene.

Figure 9.(a) shows the 3D scenery we designed, in which there stands the puppet Wusong. After several steps of processing, such as blurring, edge strokes, sharpening, we added a background with the rice paper. Figure 9.(b) shows the final result after our rendering.

5 Conclusions and Future Works

This paper presents using body gestures to control the Chinese shadow puppets with the Microsoft Kinect sensor. By analyzing the motion of the actors in the Chinese famous drama, Wusong Fights the Tiger, we propose a general framework for controlling two shadow puppets, a human model and an animal model. A performer can conduct simple actions such as turning the head, stretching the arms or kicking the legs. However, it is more difficult for a normal performer to simulate more complicated movements, for example, back flips and splits. Therefore we define some special postures to represent these difficult movements. Besides, in order to be compatible with the Chinese drama style, we use water color to paint the background scenery and the foreground characters. Some preliminary results demonstrate the effectiveness of this work.

However, we had only introduced controlling the human and animal puppets in this paper. We still need to map the human skeleton to other categories of models, such as birds, plants, or arbitrary shaped puppets. Another possible work is to directly capture the puppeteer's movements of manipulating the puppets and use these gestures to control the puppet movements.

Acknowledgments. The work described in this article was partially supported by the National Natural Science Foundation of China (Project No. 61005038) and an internal funding from United International College.

References

1. Chen, F.P.L.: Visions for the Masses; Chinese Shadow Plays from Shaanxi and Shanxi. Cornell University (2004)

2. Leite, L., Orvalho, V.: Shape your body: Control a virtual silhouette using body motion. In: Proceedings of ACM SIGCHI Conference on Human Factors in Computing Systems (2012)
3. Zhang, Z.R.M.Y.: Robust hand gesture recognition with kinect sensor. In: Proceedings of the 19th ACM International Conference on Multimedia, pp. 759–760 (2011)
4. banovic Volkan Isler Linnda R. Caporeal Jeff Trinkle, E.M.M.: Shadowplay: A generative model for nonverbal human-robot interaction. In: Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction, pp. 117–124 (2009)
5. Shao, L., Ji, L., Liu, Y., Zhang, J.: Human action segmentation and recognition via motion and shape analysis. Pattern Recognition Letters 33, 438–445 (2012)
6. Shao, L., Gao, R., Liu, Y., Zhang, H.: Transform based spatio-temporal descriptors for human action recognition. Neurocomputing 74, 962–973 (2011)
7. Wu, D., Shao, L.: Silhouette analysis based action recognition via exploiting human poses. IEEE Transactions on Circuits and Systems for Video Technology (2012)
8. Sturman, D.: Computer puppetry. IEEE Computer Graphics and Applications 18, 38–45 (1998)
9. Graham, W.: Course Notes: 3D Character Animation by Computer. In: Course Notes: 3D Character Animation by Computer, ACM SIGGRAPH, pp. 6–79 (1989)
10. Tardif, H.: Character animation in real time. In: Panel Proceedings: Applications of Virtual Reality I, ACM SIGGRAPH (1991)
11. Shin, H., Lee, J., Shin, S.Y., Gleicher, M.: Computer puppetry: An importance-based approach. ACM Trans. Graph., 67–94 (2001)
12. Hsu, W.S., Ye, T.: Planning character motions for shadow play animations. In: Proc. CASA, pp. 184–190 (2005)
13. Tan, K.L., Talib, A.Z., Osman, M.A.: Real-time simulation and interactive animation of shadow play puppets using opengl. International Journal of IJCIE 4, 1–8 (2010)
14. Lu, F., Tian, F., Jian, Y., Ca, X., Lu, W., Li, G., Zhang, X., Dai, G., Wang, H.: Shadowstory: creative and collaborative digital storytelling inspired by cultural heritage. In: Proc. of the ACM Conference on Human Factors in Computing Systems, pp. 1919–1928 (2011)
15. Pan, J., Zhang, J.J.: Sketch-Based Skeleton-Driven 2D Animation and Motion Capture. In: Pan, Z., Cheok, A.D., Müller, W. (eds.) Transactions on Edutainment VI. LNCS, vol. 6758, pp. 164–181. Springer, Heidelberg (2011)
16. Kean, S., Hall, J.C., Perry, P.: Meet the Kinect: An Introduction to Programming Natural User Interfaces. Springer-Verlag New York Inc. (2011)
17. Emily Gobeille, T.W.: Interactive puppet prototype with xbox kinect (2011)
18. Walther, J.: Kinect digital puppeteering with openkinect (2010)
19. Boyle, C., Foster, A.: We be monsters: A collaborative kinect puppet (2011)
20. Lei, S.I.E., Chang, C.-F.: Real-Time Rendering of Watercolor Effects for Virtual Environments. In: Aizawa, K., Nakamura, Y., Satoh, S. (eds.) PCM 2004. LNCS, vol. 3333, pp. 474–481. Springer, Heidelberg (2004)