2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)
Dec 20-22, 2013, Shenyang, China

Virtual Experiment System for Electrician Training based on Kinect and Unity3D

Hongjian Liao
Laboratory Center
Guangzhou University
Guangzhou, China
LiaoHongJian@gzhu.edu.cn

Zhe Qu
Information Institute of Library
Guangzhou University
Guangzhou, China
soluloo@126.com

***Abstract*—In order to establish low-cost and strongly-immersive desktop virtual experiment system, a solution based on Kinect and Unity3D engine technology was herein proposed, with a view to applying Kinect gesture recognition and triggering more spontaneous human-computer interactions in three-dimensional virtual environment. A kind of algorithm tailored to the detection of concave-convex points of fingers is put forward to identify various gestures and interaction semantics. In the context of Unity3D, Finite-State Machine (FSM) programming was applied in intelligent management for experimental logic tasks. A "Virtual Experiment System for Electrician Training" was designed and put into practice by these methods. The applications of "Lighting Circuit" module prove that these methods can be satisfyingly helpful to complete virtual experimental tasks and improve user experience. Compared with traditional WIMP interaction, Kinect somatosensory interaction is combined with Unity3D so that three-dimensional virtual system with strong immersion can be established.**

Keywords—virtual experiment system; electrician training; Kinect; Unity 3D; gesture interaction

I. Introduction

Virtual Experiment System (VES) creates a three-dimensional virtual experiment world which achieves real-time demonstration of changes in physical objects and their interactions for all users by means of simulation. So users can not only take advantage of visual sense, auditory sense, tactile sense and a variety of other channels to conduct natural interaction with virtual world, but also can directly probe into roles and changes of simulation objects in their environments and enjoy the immersive feeling [1].

Virtual experiment can go beyond the traditional simple object teaching methods and make microscopic experiment, dangerous experiment and costly experiment become possible, which is an important auxiliary means for experimental teaching in schools. Virtual reality technology is characterized by immersion, interaction, imagination, reality, multi-sensory and other features [2]. In order to give these advantages into full play, a three-dimensional (3D) virtual experiment system should overshadow the existing WIMP interaction in terms of interactive tools, and utilize the more advanced natural human-computer interactive technology. Moreover, a powerful and flexible engine is a must for the management over virtual scenes and experimental logic tasks.

Kinect is innovative game somatosensory equipment developed by Microsoft. With Kinect, game players can launch human-computer interactions without any handheld devices [3]. Kinect mainly adopts infrared camera, image recognition, image processing and other technologies. Its powerful functions and strong user demands help it expand from game industry to PC application community. Kinect applications based on PC desktop are gradually popular. Unity3D is a multi-platform game development tool at ready for creation, which is a fully-integrated dedicated game engine. It boasts powerful virtual scene editing function, rendering function and powerful logic management function based on Java Script/C#/Boo and other computer languages [4].

In view of "expensive equipment, poor platform portability" and other shortcomings of dedicated virtual experiment system [5], this paper proposed to apply Kinect and Unity3D engine to set up inexpensive and strongly-immersive desktop virtual experiment system. Taking Lighting Circuits (a subproject of "Virtual Experiment System for Electrician Training") as an example, two key issues are herein analyzed, i.e., "Gesture recognition and interaction by Kinect" and "Experiment logic management by Unity3D finite-state machine programming". Related algorithms and steps for system prototype are also proposed.

II. Design of Lighting Circuit Virtual Experiment System

A. *Purpose of Virtual Circuit Virtual Experiment*

Lighting circuit experiment is a basic experiment for physics, electronics and other science and engineering disciplines in colleges and universities, which mainly enable students to understand the principles of lighting circuits, familiarize with each component, master the correct wiring methods and achieve troubleshooting of lighting circuit faults through test pencil and universal meter. Lighting Circuit Virtual Experiment System is mainly developed in order to solve insoluble problems of real experiment as follows.

Firstly, Absolute safety experiments are achieved. In the assembly of lighting circuits under real experiment, if the differences of live wire, neutral wire and ground wire are

This research was sponsored by Guangzhou Municipal Universities Research Project named "Research and Application of 3D Virtual Experiment System based on Kinect Natural Human-Computer Interaction"(NO.2012A095)".

978-1-4799-2565-0/13/$31.00 ©2013 IEEE

ignored and confused, this may lead to electric shocks and cause life-threatening dangers.

Secondly, in classroom teaching for real experiment, “Troubleshooting” is an important part. In front of dozens of students or test consoles, a teacher is almost impossible to create a variety of random and diversified “fault circuits” for every student to conduct troubleshooting. As long as a proper algorithm is designed, the virtual experiment system can easily make “Troubleshooting” come true.

B. Design of Interaction

Main objects of Three-dimensional Lighting Circuit Virtual Experiment System (i.e., interactive objects) are as shown in Fig.1, which fall into two types as follows: Experimental tools, such as wire stripper, test pencil, universal meter and needle-nose pliers; Components and parts, such as terminal board, dual-control switch, rectifier and fluorescent tube.

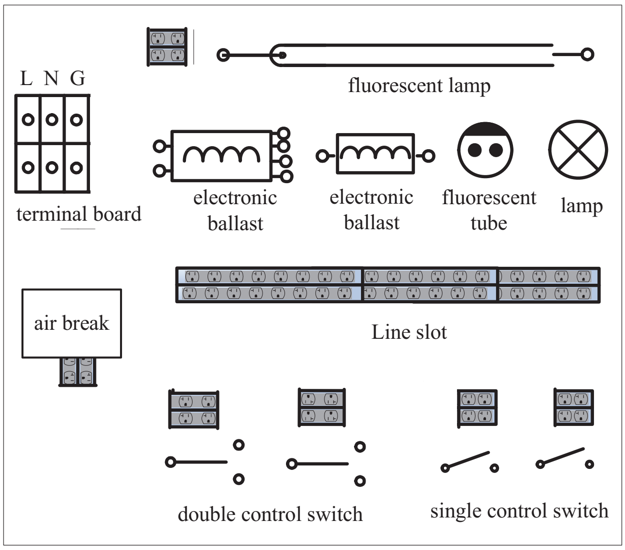


Fig. 1. Lighting circuit virtual experiment scenes and interactive objects

The functional modules of Lighting Circuit Virtual Experiment System are defined in Table 1, among which “Manual Wiring” and “Troubleshooting” are focuses of this system. The required hierarchy for interaction is specified at Interactive Level.

TABLE I. TABLE 1 DEFINITION FOR MAIN FUNCTIONS OF LIGHTING

Code	Function Name	Interactive Level
101	Experimental Demonstration	none
102	Manual Wiring	high
103	Troubleshooting	middle
104	Components & Parts Demonstration	middle
105	Error Analysis	low

Through the definition and analysis for virtual experiment objects and functions, it is proposed to achieve virtual experiments by means of “gamification” so that learners can fulfill experimental tasks by means of games under the relaxed and immersive status [6]. In this process, users’ trial and error and thorough explorations should be allowed. In terms of human-computer interaction, Kinect gesture interaction can be adopted to improve system immersion. Experiment scenes can be established with Unity3D engine so as to create highly-simulated experiment scenarios. With use of finite-state machine programming method, complex experimental logic management can be implemented.

III. KINECT GESTURE RECOGNITION ALGORITHM

A. Kinect Gesture Recognition Process

Currently the open source community of OpenNI provides four types of hand gesture for Kinect [7], but it’s too simple to meet the demand of gesture interaction completely in virtual experiment system. So this research extended the gesture recognition based on OpenNI to complete interaction tasks in virtual experiment efficiently. The work progress of gesture recognition is shown in Fig.2.

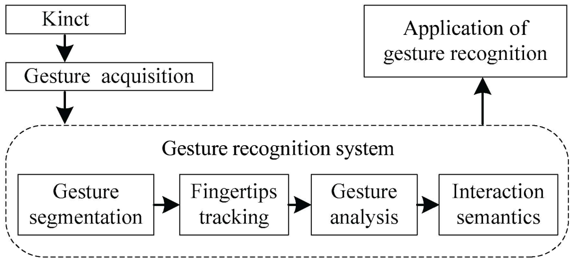


Fig. 2. Gesture recognition process by Kinect

Firstly, it need to obtain the data flow of depth image from Kinect, and then cut apart the gesture according to position of depth from hand, and get the quasi-binary image finally by this step. Secondly, it cut apart the fingertip from gesture and tracks target deformation and movement effectively. Thirdly, it generates mode parameters based on gesture features by which gesture classified and semantics defined on demand. Finally, it drives gesture by defined semantics to complete virtual experiment interaction.

B. Fingertip and Convexity Detection Algorithm

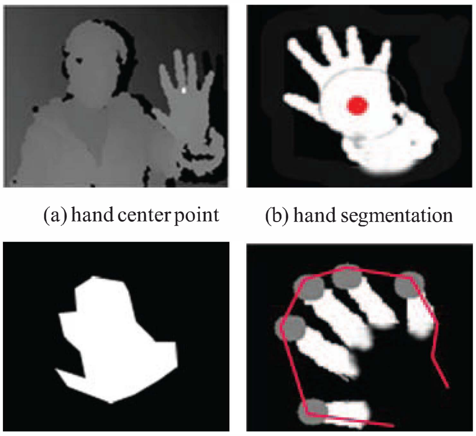
This paper proposed the convexity detection algorithm for gesture recognition based on Sklansky’s 3-coins algorithm [8]. This approach attempts to locate the convex points and concave points of the hand shape. Then according to the number of convex and concave points detected, a gesture and its semantics is recognized. And the key steps are as follows.

Step1: Track the center position of palm by NITE middleware of OpenNI based on hand depth image. It is shown as Fig.3 (a).

Step2: Make threshold process with hand according to the depth value of palm’s center point, and then get the depth image of palm. And finally get the outer contour of hand by linearization processing with area of palm and background in depth image. The result is shown in Fig.3 (b).

Step3: Approximating the hand contour with a polygon. This step aims to reduce the unwanted convexity point number by approximating the hand contour with a polygon that has fewer vertices. The drawing of hand polygon is shown in Fig.3(c).

Step4: Calculate the convex and concave points of the approximation polygon. This work was done by OpenCV function <cvConvexityDefects> which is based on the Sklansky's graph algorithm. Using this function we got the convex hull and points set as shown in Fig.3 (d).



(a) hand center point (b) hand segmentation

(c) Palm region (d) convexity detection

Fig. 3. Gesture recognition process based fingertips analysis

C. *Gesture Analysis and Semantics Description*

In virtual experiments, the operation mainly includes such gestures as selecting, picking up, moving, zoom, rotation, release. And the description of gestures and corresponding meanings are as follows.

- Selecting: When the center point of palm is in the bounding box of a object, and residence time is above 3 seconds, the object will be selected.
- Picking up: As the hand from open into a fist on the basis of selecting, the object will be picked up.
- Rotation: When an object is picked up and all the point of fingertips rotating in one direction, the object will be rotated.
- Zoom: When a object is picked up and the numbers of fingertips between three and five, and the distance between all fingertip points and center point of palm is reduce or increase, the object will be enlarged or reduced.
- Move: When the hand waving in a certain direction, the picked up object will be moved along with hand.
- Release: When the numbers of fingertips points is zero, That is, the hand is in open state, the object will be released.

IV. FINITE-STATE MACHINE PROGRAMMING FOR UNITY3D

A. *Principle of Finite-State Machine*

Finite-State Machine (FSM) refers to a mathematical model indicating finite-number states, as well as transition, action and other behaviors between these states [9]. FSM decomposes the behavior of an object into easily-processed "blocks" or states. For example, light switch is a simple FSM. It sees two states, i.e., "ON" and "OFF". The operating principle of FSM is as shown in Fig.4. After the occurrence of an event according to the current state, the execution action is determined and the serial number of next state is set.

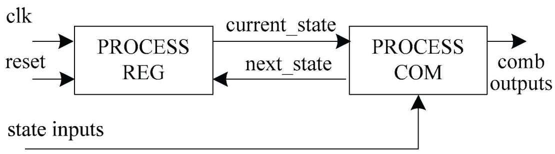


Fig. 4. Operating Principle of Finite-State Machine

B. *Instance of FSM in Circuit Fault Diagnosis*

For example, in fault diagnosis for dual-control switch of lighting circuits, failure judgment for dual-control switch needs a series of judgments for other states. And these states change under the different conditions. The logical relationship of some states is as shown in Fig.5.

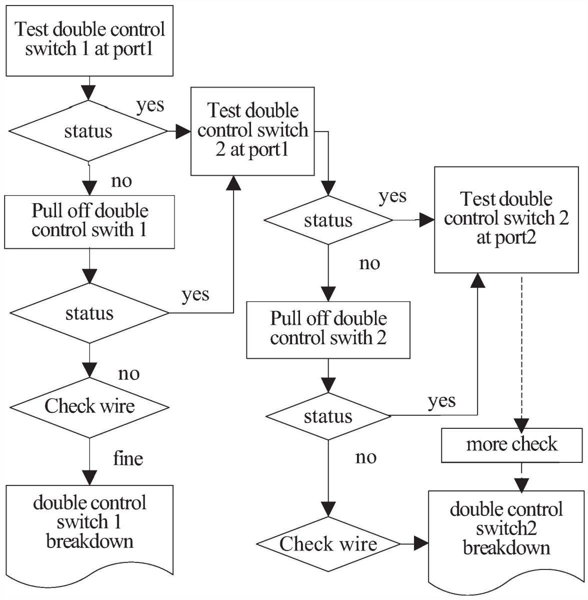


Fig. 5. Fragment Logic of Fault Diagnosis in Lighting Circuits

Abstract description can be made for events and state transition from Fig.5. Specifically speaking: Under State s0, if Event e0 occurs, Action a0 should be executed, and the state should remain unchanged. If Event e1 occurs, Action a1 should be executed, and the state should shift to State s1. If Event e2 occurs, Action a2 should be executed, and the state should shift to State s2. And the Execution process can be shown with the following codes:

```
void e0_event_function(int * nxt_state)
{
   int cur_state;
   cur_state =nxt_state;
   switch(cur_state){
        case s0:
        case s2:
        nxt_state = s0;
   }
}
```

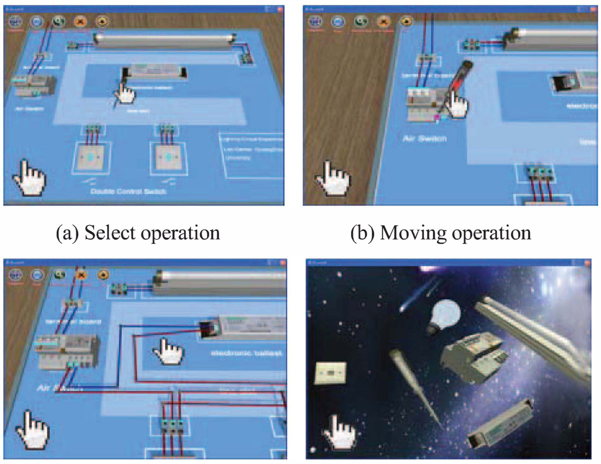
If Event1 occurs, the following function should be executed:

```
void e1_event_function(int * nxt_state)
 {
    int cur_state;
    cur_state = nxt_state;
    switch(cur_state){
         case s0:
    nxt_state = s1;
     }
}
```

Similarly, if event 2 occurs, the similar function should be executed.

C. *System Prototype*

This system employs Unity3D as a three-dimensional space management and display platform. With Kinect Development SDK 1.8, OpenNI, Middleware NITE 2.2 program development kit and OpenCV image processing function library, thanks to development tool of Microsoft Visual Studio 2010, definition and expansion of Kinect gesture are designed and completed in Windows7 environment. Interactive control for gestures in the virtual experiment is successfully achieved. Gesture interactions in the virtual experiment are as shown in Fig.6.



(a) Select operation (b) Moving operation

(c) Zooming operation (d) Rotation operation

Fig. 6. gesture interaction in virtual experiment

The process that different gestures alternately work together to complete virtual assembly is as shown in Figure 6. Firstly, scenes and models created in 3D Max should be exported in the format of "dts", which should be subsequently imported into Unity3D engine. Secondly, users should trigger gesture of "RaiseHand" by raising their hands. If such a gesture is successfully triggered, it is necessary to find the position of hands and mark with gesture icon in the scenes.

Gesture of "Select" is used to select the required parts, while gesture of "Move", gesture of "Rotate" and gesture of "Scale" are respectively used for movement, rotation and scaling of experiment objects, as shown in Fig.6(a) / (b) / (c) / (d).

V. CONCLUSIONS

Under the premise of function analysis for Virtual Experiment System for Electrician Training, Kinect is hereby adopted as a means of human-computer interaction. A kind of algorithm tailored to the detection of concave-convex points of fingers is put forward to identify various gestures and interaction semantics. The practice has proved that this method can meet the interactive needs in the virtual experiment process. On this basis, in the context of Unity3D, Finite-State Machine (FSM) programming is applied in intelligent management for experimental logic tasks. Thanks to these methods, "Virtual Experiment System for Electrician Training" is designed and put into practice. The applications of "Lighting Circuit" module prove that this method can be satisfyingly helpful to complete virtual experimental tasks and improve user experience. Compared with traditional WIMP interaction, Kinect somatosensory interaction is combined with Unity3D so that three-dimensional virtual system with strong immersion can be established. The perfect integration of Kinect and Unity3D also plays an important role in pushing forward the applications of virtual reality technology.

REFERENCES

[1] Li Xin, Chen Mingming. "A study for educational applications of virtual experiment system", Proc., Education Technology and Computer (ICETC), Jun. 2010, pp.296-298, doi:10.1109/ICETC.2010.5529677.

[2] Shuang Li. "Research on application of virtual reality technology in teaching and training", Proc., Computer and Electrical Engineering, Dec. 2009, pp.77-80, doi:10.1109/ICCEE.2009.22.

[3] Webb J, Ashley J. Beginning Kinect Programming With the Microsoft Kinect SDK, California :Apress, 2012, pp.4-6.

[4] Jingming Xie, "Research on key technologies base Unity3D game engine", Proc. Computer Science & Education (ICCSE), July 2012, pp. 695-699. doi:10.1109/ICCSE.2012.6295169.

[5] Abdul-Kader,H.M, "E-Learning Systems in Virtual Environment", Proc., Information & Communications Technology, Dec. 2008, pp.71-76, doi:10.1109/ITICT.2008.4806644.

[6] Takaoka, R., mokawa, M., Okamoto,T., "A framework of educational co ntrol in game-based learning environment", Proc., Advanced Learning T echnologies (ICALT), July 2011, pp.32-36, doi:10.1109/ICALT.2011.18.

[7] OpenNI. OpenNI User Guide, 3rd ed., 2011, pp.3-6. unpublished.

[8] Klette, G. "A recursive algorithm for calculating the relative convex hull", Proc. Image and Vision Computing New Zealand (IVCNZ), May. 2012, pp. 1-7, doi: 10.1109/IVCNZ.2010.6148857.

[9] Yang, M., Lai, J., Hongying Xu. "Finite state machine synthesis based on relay-based algorithm", Digital System Design (DSD), Sept. 2012, pp.405-410, doi:10.1109/DSD.2012.60.