*2011 International Conference on Advanced Materials Engineering*
*IPCSIT vol.15 (2011) © (2011) IACSIT Press, Singapore*

# 3D Real-time Display for A Typical Construction Machinery

Wen Gong [1+], Zhong-Jian Yu [1], Tian-Liang Lin [2]

[1] Shanghai Institute of Special Equipment Inspection and Technical Research

[2] College of Mechanical Engineering and Automation, Huaqiao University

**Abstract.** The hardware-in-the-loop simulation research on hydraulic excavator presents the performance of the entire machine, when the character of some components varies. The simulation has important significance to the controlling of the entire machine and manipulability research. The 3D real-time simulation platform, based on DirectX SDK and VC++, is developed to realize the 3-D real-time kinematics simulation of the hydraulic excavator. The result shows the well-designed platform is a good solution for the 3D real-time simulation visualization, data analysis of off-line simulation and remote monitoring.

**Keywords:** kinematics, models, simulation, DirectX

## 1. Introduction

During the process of the excavator's real-time simulation, the operator needs the information that presents the position and pose of the excavator in the real-time simulation system, comparing that of the target position and pose, to determine the direction and feed amount of the joystick. Considering that, it is important for the real-time simulation platform to be able to display the status of the whole process in real-time. Generally, a real-time monitoring system can display the state of the object in the form of plane curves or data texts. However, under this environment, the operator has to do some comprehensive treatment work, when he is operating the virtual excavator, which increases the working strength and would be burden for the operator. Using the 3D visualization technology, we can get the system's state data and use it to drive the 3D excavator to move in real-time. In this way, the operator can get the useful state data of the virtual excavator directly, make the information more abundant, improve man-machine interaction environment and simulation efficiency. This paper develops an 3D excavator real-time display platform, which is driven by the data from the serial port, under the environment of VC++ and DirectX SDK. The 3D display platform is applied on the real-time simulation of excavator.

## 2. Kinematics Analysis of Excavator

In order to display the position and pose of the virtual excavator correctly, kinematics analysis is necessary. As the excavator is a complicated multi-degree of freedom manipulator, when one part of the system moves, the status of the others changes. The following work is to simplify the excavator's structure and build the kinematics model.

An excavator is comprised of base, cab, boom, arm and bucket, Fig. 1. The crawler of the base drives the machine to move on the ground; the cab rotates on the base and the boom, arm and bucket is driven by the cylinder respectively. Considering the machine as a multi-joint manipulator, which consists of five links and five joints.

Firstly, an absolute three-dimensional coordinate system is established, as in Fig. 2. The coordinate system is attached to the base. Other three local coordinate systems are attached to the cab, boom, arm and

[+] Corresponding author. Tel.: +86-21-32584764; fax: +86-21-32584764.
*E-mail address*: gwsunny@zju.edu.cn.

bucket respectively. Following the Denavit-Hartenberg method, the position and pose of excavator can be described by the homogeneous transfer matrix [1,2]. Assumption that the joint angles are represented by $\gamma$, $\alpha$, $\beta$ and $\theta$.

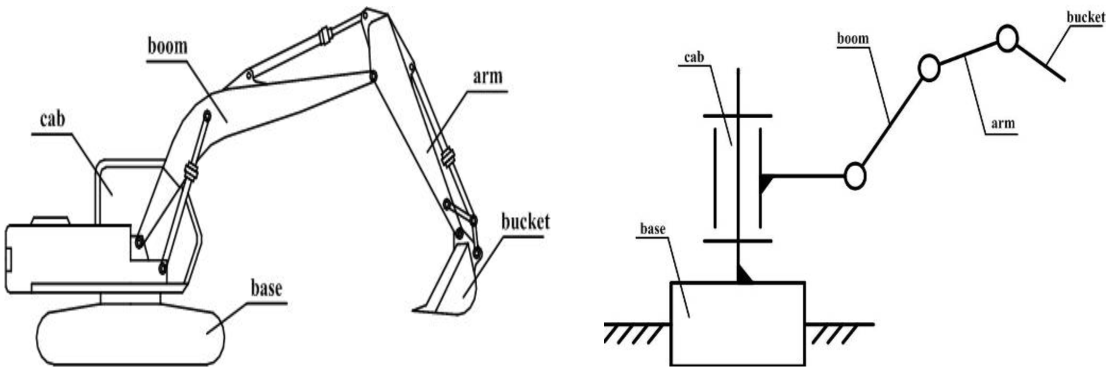


Fig. 1: Structural simplification

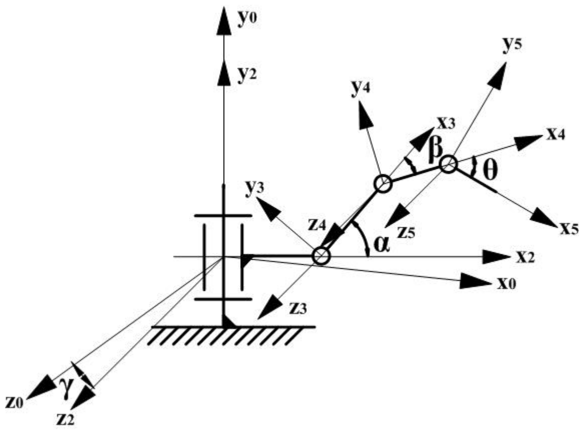


Fig. 2: Kinematics coordinate system

The coordinate transformation matrix between the base and cab is:

$$ {}^{0}_{2}\boldsymbol{T} = \begin{bmatrix} cos\gamma & 0 & sin\gamma & 0 \\ 0 & 1 & 0 & 0 \\ -sin\gamma & 0 & cos\gamma & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

The coordinate transformation matrix between the cab and boom is:

$$ {}^{2}_{3}\boldsymbol{T} = \begin{bmatrix} cos\alpha & -sin\alpha & 0 & d_2 \\ sin\alpha & cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

where $d_2$ is the length of the cab.

So the coordinate transformation matrix between the boom and cab is: ${}^{0}_{3}T={}^{0}_{2}T\,{}^{2}_{3}T$ .

The coordinate transformation matrix between the boom and arm is:

$$ {}^{3}_{4}\boldsymbol{T} = \begin{bmatrix} cos\beta & -sin\beta & 0 & d_3 \\ sin\beta & cos\beta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3} $$

where $d_3$ is the length of the boom.

So the coordinate transformation matrix between the arm and cab is: ${}^{0}_{4}T={}^{0}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T={}^{0}_{3}T\,{}^{3}_{4}T$ .

The coordinate transformation matrix between the arm and bucket is:

$$
{}^4_5\boldsymbol{T} = \begin{bmatrix} cos\theta & -sin\theta & 0 & d_4 \\ sin\theta & cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}
$$

where is the $d_4$ length of the arm.

So the coordinate transformation matrix between the bucket and cab is: ${}^0_5T={}^0_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T={}^0_4T\,{}^4_5T$ .

# 3. Excavator's 3D Solid Modeling

Excavator is a machine of complex body parts, and therefore this paper uses a professional three-dimensional modeling software to create the various parts of the excavator model. In this paper, 3ds Max is used to build the model of the excavator. 3ds Max provides users with a high-level command of three-dimensional modeling for a wide variety of modeling complex shapes [3]. More importantly, it provides powerful rendering functionality and the ability to edit material to ensure that the model has enough realistic three-dimensional effect, which can meet the platform's requirements on the fine of the model.

In the 3ds Max environment, in accordance with previous kinematics analysis, the establishment of the three-dimensional solid model of the base, cab, boom, arm and bucket is taken respectively. In order to facilitate the following procedures for the reading of model data, model file is saved, a DirectX can be identified, as an X format. .X format document preserves the required information for DirectX Mesh model mesh object and a list of materials and other information, so that using 3ds Max software to obtain the model data.

DirectX SDK is a set of APIs which provides a set of multimedia development tools for Windows system applications and hardware associated itself with a strong three-dimensional graphics display ability [4,5]. The framework of three-dimensional rendering procedure is designed and the rendering process flow chart (Fig. 3) is as follows.

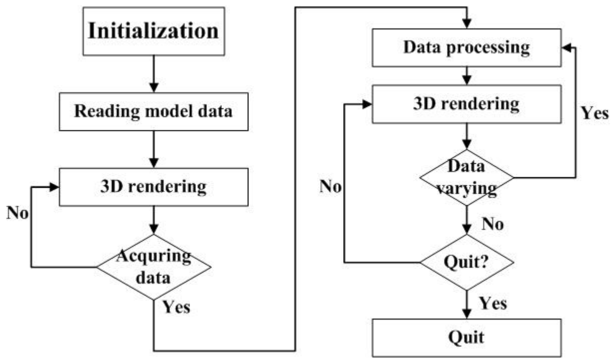


Fig. 3: 3D rendering flow chart

The result of the 3D real-time platform is shown in Fig. 4.

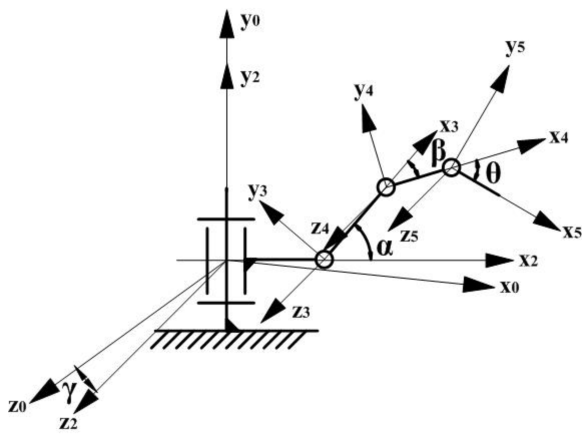


Fig. 4: Process of the simulation

## 4. Summary

Under the Windows system environment, this paper uses DirectX SDK and VC++ to develop an 3D real-time display platform for excavator's hardware-in-the-loop simulation. The platform uses 3ds Max software to create three-dimensional model, therefore, when the object changes, the quick change in 3ds Max software can create three-dimensional object model and reduce the consuming time of obtaining three-dimensional model data.

Experiments have proved that the platform can be a good solution for the 3D real-time display. During the process of the excavator's real-time simulation, the platform works well. The model runs smoothly and the data sending and receiving stably and reliably.

## 5. Acknowledgements

The authors gratefully acknowledge the support from hoisting machinery virtual test evaluation and safety assessment technology research project.

## 6. References

[1] Y. L. Xiong. *Fundamentals of Robot Techniques*. Huazhong University of Science and Technology Press, 2004.

[2] F. F. Zhou, X. P. Fan, and Z. Ye. 3-D Virtual Robotic Model Generated by D-H Parameters. *Journal of System Simulation*. 2006, 18 (4): 947-950.

[3] Murdock and L. Kelly. *3ds Max 9 Bible*. Posts & Telecom Press, 2008.

[4] J. Sanchez and M. Canton. *DirectX 3D Graphics Programming Bible*. Beijing Publishing House of Electronics Industry Press, 2000.

[5] J. J. Shang. *Detailed Direct3D Game Development Technology*. Posts & Telecom Press, 2006.