

R&D of Industrial Robot Simulation System Based on Unity 3D

JIAO Heyan

Tianjin Key Laboratory of Information Sensing & Intelligent Control, Tianjin University of Technology and Education, Tianjin, China

Abstract: For the students who have opened various robot-related courses, there are dangers of misoperation during the operation of the physical robot and the single form of the current teaching software, lack of interactive modules, and insufficient image. Based on the Unity 3D platform, 3D max software is used to complete 3D scene modeling. C# is used to realize the virtual operation teaching of industrial robot equipment through interface interaction technology, providing users with a good human-computer interaction environment and learning robots in advance. Knowledge, in order to improve the efficiency of industrial robot teaching work and the quality of education and training.

Keywords: Industrial robots, Unity 3D, virtual simulation, Human interaction

1. Introduction

Virtual reality technology is a hot topic at present, and one of the key development areas in many countries and enterprises [1]. In addition to the application in virtual manufacturing, 3D somatosensory film, interior design and other fields, virtual reality technology has also brought new education concept and education method to the education industry. Especially in the industrial robot profession education, industrial robot is an important part of the current development of intelligent manufacturing and intelligent equipment. Many enterprises or universities hope to quickly train high-quality skilled talents in the field of robotics [2].

With the advent of the golden age of mobile Internet terminal, mobile cross-platform offline programming system will become an inevitable trend. Due to its high cost performance, simple development process, cross-platform and strong compatibility and other characteristics, more and more are being used as the development platform in virtual reality projects. This paper is based on the Unity 3d engine to create an industrial robot virtual operating system featuring authenticity and professional interactivity. The design mode of the system is discussed, and the key technologies including robot kinematics, collision detection, subtitle prompt, highlighting, UGUI and 3d scene browsing are realized.

2. Page Size and Layout

Unity is one of the most professional, stable, efficient and supported platforms on the market today. Therefore, this paper chooses Unity 3D platform for offline programming application software development of cross-platform robots [3]. The simulation software is divided into two parts: industrial robot design and UI design. The specific structure is shown in figure 1. The teaching device software part contains the functions of manual mode, automatic mode and program storage, program modification, and robot return to zero, etc. The UGUI of Unity3D is used for design. The robot scene mainly has the functions of perspective switch, scene roaming and software description. The simulation software can be released to various platforms such as Windows, ios or Android

for easy use.

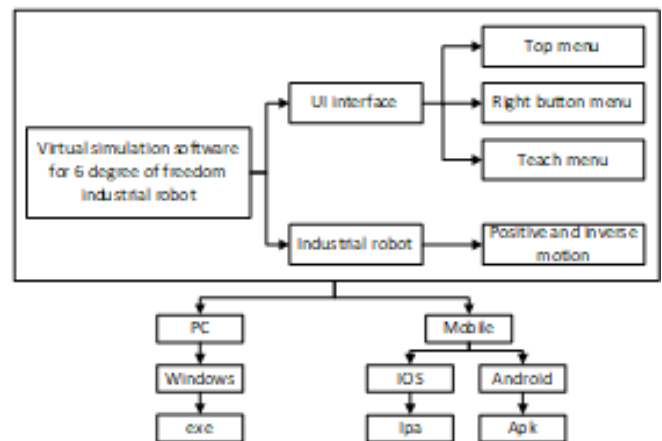


Figure 1: System structure diagram

3. Robot teaching simulation

In order to establish the robot teaching scene, we need to model the robot ontology and work piece in 3D max, and build the robot scene in Unity 3D. In order to improve the brightness and reality of the scene, a directional light is added to simulate the solar light source in the scene, and the effect of the scene is improved by adjusting the direction, color and shadows of the light source. Secondly, the kinematics of the robot is solved based on the analysis of the physical parameters of the robot. Planning is carried out to analyze the interpolation method of spatial straight lines and circular arcs. The UI interface for teaching scenarios is established.

3.1 Establishment of robot model and scene

The RB08 robot is a handling robot with a maximum load of 8kg. It is a multi-functional industrial robot with its small size and flexible movement. The RB08 robot has 6 degrees of freedom, of which 3 degrees of freedom describe the end position of the robot, and 3 degrees of freedom are used to describe the end pose of the robot [4]. The ontology model of the robot is shown in Figure 2.

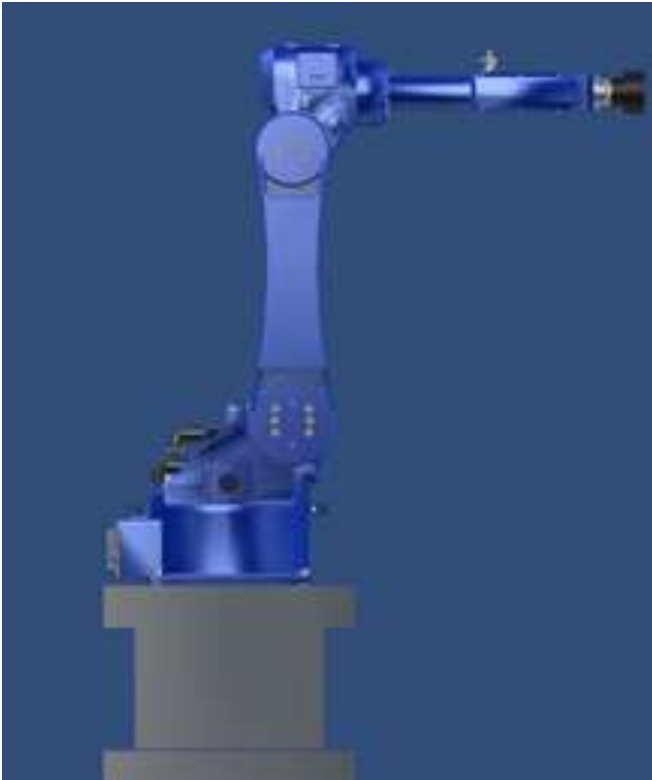


Figure 2: Robot body and 3D model

In SolidWorks, each joint of the robot is modeled, then 3ds Max is imported to modify the coordinates of each joint, and the father-son relationship between the joints is set up, so that the six joints can be controlled passively. The robot connecting rod coordinate system is shown in Figure 3.

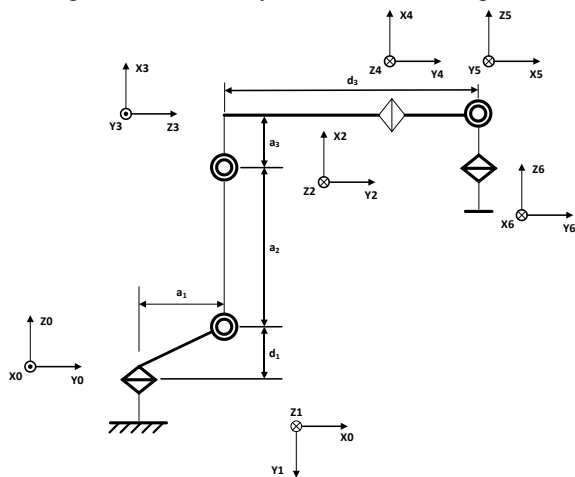


Figure 3: Robot Link Coordinate System

Using the D-H matrix method [5], each link parameter and joint variable are set according to the specific parameters of the robot as shown in Table 1.

Table 1: RB08 robot D-H parameters

Joint	$\theta/^\circ$	$\alpha/^\circ$	d/mm	a/mm
1	θ_1	-90	450	170
2	θ_2	0	0	560
3	θ_3	-90	0	155
4	θ_4	90	630	0
5	θ_5	90	0	0
6	θ_6	0	0	0

3.2 Robot kinematics analysis

Robot kinematics describes the kinematics relationship between the joints of the robot and the rigid body of the robot. Among them, there are many methods of kinematics analysis, among which homogeneous transformation is a more intuitive and convenient way.

Formula (1) is the transformation matrix of connecting rod. The specific link transformation matrix of each joint can be obtained by substituting the link parameters in table 1 into formula (1).

$${}^{i-1}T_i = T_{rot}(z_{i-1}, \theta_i) T_{tran}(z_{i-1}, d_i) T_{tran}(x_{i-1}, a_i) T_{rot}(x_{i-1}, \alpha_i) = \begin{pmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\sin \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1)$$

According to the homogeneous transformation matrix between adjacent joint coordinate systems, the transformation matrix 0T_n of the last link attachment coordinate system of the n-degree-of-freedom series robot relative to the base coordinate system can be obtained as shown in formula (2):

$${}^0T_n = {}^0T_1 {}^1T_2 \cdots {}^{n-2}T_{n-1} {}^{n-1}T_n \quad (2)$$

The position of any point on the robot end link $np=[px,py,pz]$ in the base coordinates can be described as:

$${}^0P = {}^0T_1 {}^1T_2 \cdots {}^{n-2}T_{n-1} {}^{n-1}T_n {}^n P \quad (3)$$

The link transformation matrix of the six joints is sequentially multiplied to obtain a transformation matrix 0T_6 of the robot end relative to the base coordinate system.

$${}^0T_6 = \begin{pmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4)$$

Each element is in order:

$$\begin{aligned} n_x &= c_1[c_{23}(c_4c_5c_6 + s_4c_6) - s_{23}s_5c_6] + s_1(s_4c_5c_6 - c_4s_6) \\ n_y &= s_1[c_{23}(c_4c_5c_6 + s_4s_6) - s_{23}s_5c_6] - c_1(s_4c_5c_6 - c_4s_6) \\ n_z &= -s_{23}c_4c_5s_6 - c_{23}s_5c_6 - s_{23}s_4s_6 \\ o_x &= c_1[c_{23}(s_4c_6 - c_4c_5s_6) + s_{23}s_5s_6] - s_1(s_4c_5s_6 + c_4c_6) \\ o_y &= s_1[c_{23}(s_4c_6 - c_4c_5s_6) + s_{23}s_5s_6] + c_1(s_4c_5s_6 + c_4c_6) \\ o_z &= s_{23}c_4c_5s_6 + c_{23}s_5c_6 - s_{23}s_4s_6 \\ a_x &= c_1(c_{23}c_4s_5 + s_{23}c_5) + s_1s_4s_5 \\ a_y &= s_1(c_{23}c_4s_5 + s_{23}c_5) - c_1s_4s_5 \\ a_z &= -c_1s_5s_{23} + c_{23}c_5 \\ p_x &= c_1(a_3c_{23} + a_2c_2 + a_1 - d_4s_{23}) \\ p_y &= s_1(a_3c_{23} + a_2c_2 + a_1 - d_4s_{23}) \\ p_z &= -d_4c_{23} - a_3s_{23} - a_2s_2 + d_1 \end{aligned} \quad (5)$$

In the formula: $s_i = \sin(\theta_i)$, $c_i = \cos(\theta_i)$, $i=1,2,\dots,6$.

In Unity3d, script programming in C# and JavaScript language is supported. Considering the object-oriented features and powerful function library of C# language, C# is used as the development language in this paper. The key code to realize the forward solution of kinematics is as follows,

```
public Matrix4x4 WKinematics(float[] angles){
    .....
    Matrix4x4 W = T01 * T02 * T03 * T04 * T05 * T06;
    return W;
}
```

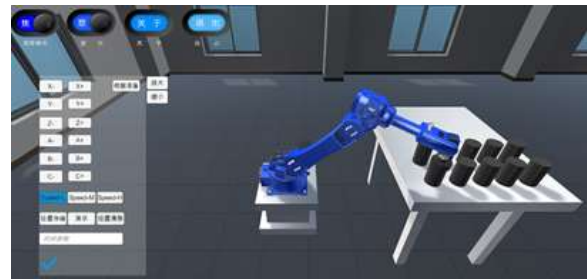
The inverse kinematics solution of the robot is based on the desired pose of the end-effector of the robot, namely the tool coordinate system, to find the rotation Angle of each joint of the robot [6]. The inverse kinematics solutions of robots are divided into analytical methods and numerical methods. Because the numerical methods are not stable in solving the inverse kinematics of robots and slow in solving the inverse kinematics of robots, the analytical methods are usually used to solve the inverse kinematics of robots, but their existence is related to the structure of robots. The sufficient condition is that if the 3 adjacent joint axes of a 6-dof robot always intersect at a point, the inverse kinematics of the robot must have an analytical solution. The RB08 robot used in this paper also has similar features. Usually, the 6-dof robot has no analytic solution, but only in some special cases. Pieper studies that when three adjacent axes intersect at one point, there must be an analytic solution. Obviously, the origin of the link coordinate system of RB08 is all intersected at a point, so the analytical solution can be obtained.

The key code to realize the inverse kinematics solution is as follows,

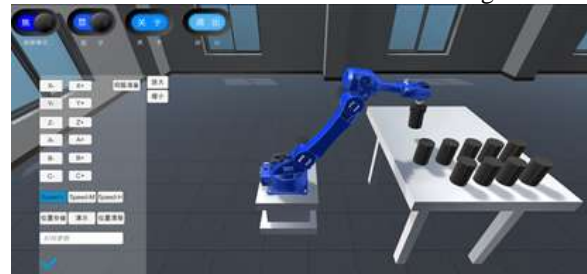
```
public float[] InvKinematics(Matrix4x4 G)
{
    // Wrist posture
    Matrix4x4 W = G * this.ETOT.inverse *
    this.WTOE.inverse;
    // Wrist position when the tool coordinate system is the
    target coordinate system
    // Joint Angle result
    float[] dstAngles = WInvKinematics(W);
    return dstAngles;
}
```

4. System Effect Display

The industrial robot virtual teaching system based on Unity 3D includes the system effect and some functional modules as shown in fig.7 and fig.8. The top menu includes: view switch, show and hide of the repository, use of the software, and exit buttons. The change of perspective includes focus mode and real person mode. Focus mode is divided into main view, top view and left view. The live-action mode is where the first person roams the warehouse. The indicator menus includes rotation and linkage of each axis, position and pose storage, deletion, speed adjustment, and enlargement and reduction of the indicator. Here is a demonstration of the robot's ability to move objects.



7 Effect and some function block diagram



8 Effect and some function block diagram

5. Conclusion

In this paper, the virtual operating system of industrial robot based on Unity 3D is studied in detail. Taking RB08 robot as the research object, ontology model and scene model are constructed by analyzing its mechanism. The physical parameters of the robot are studied and the forward and inverse kinematics of the robot is solved by using the d-h parameter method. The trajectory planning and corresponding attitude interpolation are carried out for space straight line and space arc. Using virtual reality technology, the construction of robot teaching scene was completed in Unity3D. Due to the strong cross-platform capability of Unity 3D, users can conduct theoretical study and virtual operation practice on mobile phone, computer platform and any time and place to improve learning efficiency. The research of this system not only opens up a new direction for the teaching of industrial robot equipment but also provides some reference value and application value for the development of other virtual reality projects.

References

- [1] J. Zhou Zhong, Zhou Yi, Xiao Jiang Jian. Summary of virtual reality enhancement technology. Chinese science.2015, 45 (2): 15 7 - 18 0.
- [2] J. You Wei. Design of Virtual Teaching System for Robot Body Disassembly Based on Unity3D. Mechanical Research & Application, 2018, 31 (04):230-234.
- [3] M. Unity Technologies. Unity4. X, from introduction to proficiency. Beijing: China railway publishing house, 2013.
- [4] J. zhuo yanwa, bai xiaocan, Chen yongming. Three regular curve interpolation algorithms for robots. Equipment manufacturing technology, 2009 (11):27-29.
- [5] J. DENAVIT J, HARTENBERG R S. A kinematic notation for lower - pair mechanisms -based on matrices. Trans. of the Asme. Journal of Applied Mechanics, 1955,22:215-221

- [6] A. wang xiaoli. Study on instructional programming methods of MOTOMAN industrial robot. Automation professional committee of China coal association., 2008:3
- [7] J. Chen shan. Development of robot teaching box system. Manufacturing automation, 2005(02):75-78.
- [8] J. zhuo yanwa, bai xiaocan, Chen yongming. Three regular curve interpolation algorithms for robots. Equipment manufacturing technology,2009(11):27-29

Author Profile



Jiao Heyan, Tianjin Key Laboratory of Information Sensing & Intelligent Control, Tianjin University of Technology and Education, Tianjin , China