Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 75 (2015) 107 – 112

Procedia Computer Science

2015 International Conference on Virtual and Augmented Reality in Education

Virtual Reality Application for Simulation and Off-line Programming of the Mitsubishi Movemaster RV-M1 Robot Integrated with the Oculus Rift to Improve Students Training.

Raúl Crespo[a], René García[a], Samuel Quiroz[a]*

[a]Tecnológico de Monterrey, Calle del Puente #222 Col. Ejidos de Huipulco Tlalpan 14380, D. F., México

Abstract

The overall purpose of this project is to test the impact and potential benefits of Virtual Reality technology by improving the current training methods for the operation of the Mitsubishi Movemaster RV-M1 Robot. The final application aims to achieve this goal by increasing the user's interactivity with the robot's features inside a virtual environment developed using the game engine Unity and the Oculus Rift headset for the virtual visualization. By using this application, on site operation time is decreased mainly by allowing off-line programming of the equipment. It also allows universities without or with limited industrial machinery to provide their students a way to learn and practice on industrial automation and robotics simulation topics without inconvenience. The application is designed to decrease the student's learning curve by displaying a complete virtual environment where the tridimensional model of the robotic arm can be visualized and programmed according to the real model's parameters and specifications. Joint type moving sequences are compiled into a file which afterwards can be transferred to the robot for real testing and execution. The system integrates a set of joysticks that allows the user to program each of the robot's joints as well as display several features of it in the virtual environment such as animations, images and text information that simplify the instructions shown in a printed manual. In order to find an optimized, collision-free movement sequence between two points an A* shortest path algorithm is implemented using some of the built-in tools and plugins available in Unity and its Asset store. As a result of this last feature, the application can create a comparison between the user's input sequence and a computer generated sequence based on the A* algorithm.

Peer-review under responsibility of organizing committee of the 2015 International Conference on Virtual and Augmented Reality in Education (VARE 2015)

Keywords: Virtual Reality; Robotics; Off-Line Programming; Unity; Oculus Rift; A* algorithm

* Corresponding author. *E-mail address:* samuel.quiroz.lima@gmail.com

Peer-review under responsibility of organizing committee of the 2015 International Conference on Virtual and Augmented Reality in Education (VARE 2015)
doi:10.1016/j.procs.2015.12.226

1. Introduction

In a time full of great technological achievements and reaching everyday higher levels of connectivity all around the world, innovation is mandatory. Since virtual reality (VR) is consider to enhance the user experience, its use as an innovation feature is required on applications being produced for the awakening entertainment, games and educational markets. Currently, educational methods lack of an efficient methodology to support new instructional approaches, the development of cognitive skills, and the development of attitudes on students. College facilities, such as laboratories, design and robotics workstations imply expensive investments on equipment and maintenance. Furthermore, it is common that these facilities' capacity become out of range because the amount of students exceed the number of available instruments, tools and workstations. This paper discuss the possibility to develop and implement a virtual reality application for robotics, which could be considered as a helpful tool to improve students training by enabling concurrent learning and interaction with the robot and workspace features.

Nomenclature

VR	Virtual Reality
DOF	Degree of Freedom

2. Background

The development of VR applications is a current trending phenomenon and according to Pantelidis [1] its use in education can be considered as one of the natural evolutions of computer-assisted instruction (CAI) or computer-based training (CBT). Pantelidis [1] asserts that at every level of education, VR has the potential to make a difference mainly by motivating, encouraging and exciting students. Based on Pantelidis [2] some important reasons to use VR in education are:

- Virtual reality provides new forms and methods of visualization and an alternate method for accurate presentation of materials and features, allowing different levels of close-up and angle examination of an object and examination of areas and events unavailable by other means.
- Virtual reality requires interaction and encourages active participation rather than passivity. Also distributed collaboration can be achieved.
- Virtual reality allows the student to proceed through an experience during a broad time period not fixed by a regular class schedule, at their own pace. VR offers the possibility for learning to be tailored to learner's characteristics and needs.

Virtual reality is not appropriate for every instructional objective. Pantelidis [3] makes the following suggestions on when to use and when not to use virtual reality in education. Use or consider using virtual reality when:

- A simulation could be used.
- Teaching or training using the real thing is dangerous, impossible, inconvenient, or difficult.
- A model of an environment will teach or train as well as the real thing.
- The experience of creating a simulated environment or model is important to the learning objective.
- Information visualization is needed, using graphic symbols, so it can be more easily understood.

3. Scope and Objectives

This research project aims to become an innovative educational proposal, since its main objective is to enhance the current robotics teachings by increasing the empowerment of the students through the use of virtual reality development and visualization technologies, such as Unity 3D and the Oculus Rift.

The resulting platform will enhance the students' virtual and educational experience, since they will be immerse in the virtual workspace. As a positive consequence concurrent interaction and collaborative learning will be encouraged.

In addition to these features, students' off-line programming performed in the virtual workspace will be validated afterwards by compiling the simulation code and executing it on the real world robot.

4. System Architecture

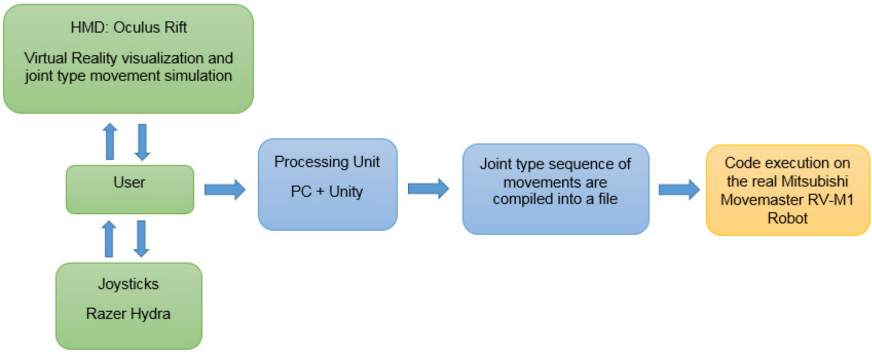

Fig. 1. System Architecture

Hardware

1. Oculus Rift
2. Razer Hydra Joysticks
3. Mitsubishi Movemaster RV-M1 Robot

Software

1. Oculus SDK
2. Sixense SDK
3. Denford Robot Communications
4. Unity 3D
5. Sketch-Up and Blender

   Unity Assets:
   a) Takohi Inverse Kinematics
   b) APEX Game Tools

5. Methodology

5.1. Process Analysis

The first step in order to develop the VR application was to analyze the process for virtualization, in this case the simulation of the robot manipulator, identifying the parts, each degree of freedom (DOF), and the tools involved (gripper), and measure them to create the 3D model replica of each one of them. This step also involved the analysis of the teach pendant programming mode in order to measure the motion limits of each DOF.

5.2. 3D Modeling

The 3D modeling was done using the design workbench Sketch-Up and Blender, since the 3D model was based on a model obtained from Sketch-Up Warehouse and modified to meet the Mitsubishi Movemaster RV-M1 Robot measures and to be able to export part by part to Blender with .dae extension file. Once in Blender, meshes of each part of the robot were joined together and parts were individually exported to Unity with .fbx file extension.

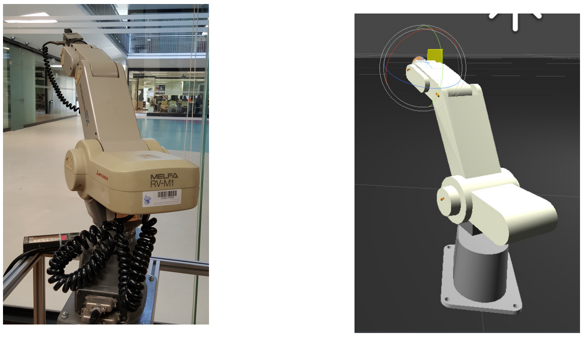

Fig. 2. (a) Real world Mitsubishi Movemaster RV-M1; (b) 3D model integrated in Unity

5.3. Integration and programming in Unity 3D

Once in Unity, each part of the robot was assigned to a different GameObject and scale was adjusted individually to a scale factor of 0.3. Finally the robot was constructed by embedding each GameObject in a hierarchical way according to the hierarchical order of the DOF's links in the real world robot.

The virtual model of the robot was programmed using built-in functions of Unity, mainly the Transform function, which allows to store and manipulate the position, rotation and scale of the GameObject in a scene. Each GameObject movement is done having a point of origin as a home reference, which is where the robot is initialized when the program starts.

By programming the degree and direction of rotation of each degree of freedom of the robot, according to the manufacturer specifications, many things are accomplished:

First, rotation angles information can be displayed to a GUI in order to assist the user when a high precision task is required to grab an object or if user is assigned a task to configure the degrees of rotation necessary to move an object form one place to another.

A core feature of this app version is a function programmed in Unity to measure the differential angle of each DOF. This function is used to create the instructions program that will be downloaded to the real world robot. This file is generated programmatically from the Unity script and it will instruct the robot which degree of freedom, angle and direction to move. The purpose of generating a second file is to keep positions. Positions coordinates are generated having as a reference the robot's gripper. The positions format includes the gripper status (open or closed) and its rotation angle

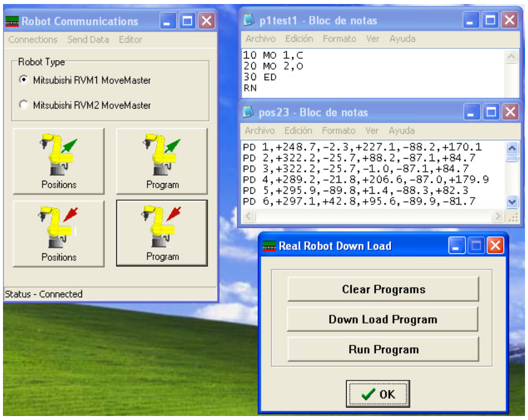

Fig. 3. Structure of both type of programs (instructions and positions), created programmatically using scripts in Unity, based on the robot's user manual. Both type of programs are downloaded to the robot using Denford Robot Communications software.

6. Results

6.1. Visualization of the VR Application

The application is programmed using a videogame engine, Unity, using the built-in function Transform and several methods embedded to work with local Euler angles. The corresponding scripts are coded using C#. The Takohi Inverse Kinematic asset is used to program the correct joint displacement in the virtual world. At this stage, the robot joints angles are displayed in a GUI. In order to program the robot to follow the most optimal movement sequence in a 2D space, the APEX Game Tools asset was used. The implementation of the collision-free motion planning algorithm produced successful results in two dimensions, where the robot's end effector moves from one point to another on the x/y plane [4]. As a result, collision-free paths and optimal movement sequence are achieved.

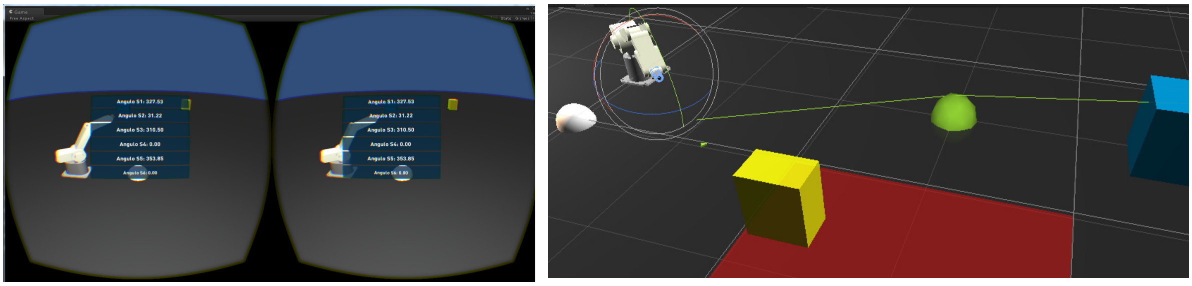

Fig. 4. (a) Virtual visualization using the Oculus Rift; (b) Pathfinding implementation between two points

In the final application the user interacts with the Oculus Rift, the virtual version of the Mitsubishi Movemaster RV-M1 Robot and the Razer Hydra Joysticks in two possible modes: teach pendant mode or automatic motion planning mode. The results from testing the application with a group of engineering students showed that students with no simulator relied on "trial and error" executions. While students using the simulator completed successfully the test in a shorter period of time. Also, doubts related with the robot's joints and spatial parameters were solved in a faster way.

7. Conclusions and Future Work

Using Virtual Reality elements for controlling robots, along with the automatic creation of programs that are used to reproduce the movements made in the virtual environment, have proved satisfactory results by minimizing robot programming time and by easing the routines' implementation. The resulting application encourages immersive interaction and enhance learning.

The further stage of research will be concentrated on the consolidation of trajectory generating and simulation phase with the program execution stage in such a way, that the determination of collision-free path can be improved.

8. Online license

References

1. Pantelidis VS. Reasons to Use Virtual Reality in Education and Training Courses and a Model to Determine When to Use Virtual Reality, Themes in science and technology education, *Klidarithmos Computer Books* 2009; **2**(2-1): 59-70.
2. Pantelidis VS. (1995). Reasons to use virtual reality in education. *VR in the Schools* 1995; **1**(1): 9. Retrieved from http://vr.coe.ecu.edu/vrits/1-1pante.htm

3. Pantelidis VS. Suggestions on when to use and when not to use virtual reality in education. *VR in the Schools* 1996; **2**(1):18. Retrieved from http://vr.coe.ecu.edu/vrits/2-1Pante.htm
4. Latombe JC. Robot motion planning, Kluwer Academic Publishers, London, 1993.