

The Application of Virtual Reality Technology in Driving Training in Exceptional Scenes

Zhou Likun

Continuing Education School
Shanghai Jiao tong University
Shanghai ,China

Abstract—With the increasing number of vehicles in China, the traffic condition is getting worse and worse. To drive safely in exceptional scenes is very significant to life and property. Using a technique with computer simulation, virtual reality and hardware such as sensors and computer interface, to simulate the interactions between driver and vehicle in such scenes is very interesting. This paper introduces the concept of this technique and the details of software and hardware components to make of this training system.

Keywords- Virtual Reality, Driving Training, Exceptional Scenes

I. OVERVIEW

In recent years, driving training systems are well developed in traditional scopes such as subject II (parking examination) and subject III (driving examination). These systems are focus on normal traffic scenes such as subway in a fine day. These systems can really make new drivers master the basic driving techniques more easier. By doing a certain operation again and again such as when and where the steering wheel, the brake should be operated, newbies can pass these examinations very easily. But it does not mean that they have already been versatile veterans and mastered all necessary skills to deal with the exceptional scences such as muddy or icy roads in strom or hailstone days.

Recently, Ministry of Transport of the PRC proposed another subject to examine the drivers in such exceptional scenes, for instance, driving on muddy, icy, steep country roads or spiral hill roads in a rainy, stormy, foggy or snowy day, etc. It's very unpratical or impossible to train the drivers in real exceptional scenes as it's very dangerous and these scenes are very occassional. But with the new technologies in computer VR and hardware we can build a system to simulate such a virtual environment to train the drivers without life danger and with high training quality.

II. SYSTEM CONCEPT AND CONSTRUCTION

INTRODUCTION

The system consists of two parts:

- Hardware: a vehicle without wheels, new sensors and factory sensors connected by CAN-Bus and embeded into this vehicle, interfacing circuit such as OA, MUX,

A/D or D/A conversion circuit, DSP, force feedback in VR, display screen, multichannel stereo, etc..

- Software: signal system, VR system, driver behavior expert system (DBES).
 - Signal system collects/sends signal from/to sensors or force feedback devices to simulate the interaction between driver and vehicle or VR environment.
 - VR system simulates the exceptional training scenes and the interaction between driver or vehicle and all objects such as ice and snow, fog, roads, hills, etc. in VR environment.
 - DBES collects the driver's behaviors in such an exceptional scene through the signals collected from sensors, force feedback devices, etc. Firstly, with self-learning, it will know if a certain behavior is effective and will build up a behavior decision database. Later, with this database, it can judge the driver's behaviors and give out the examination points.

The system should be configured beforehand such as selecting a vehicle (car, bus or truck) and a specific scene. Different vehicles and scenes will impact the vehicle's velocities, turn radius, force coefficient, and signal parameters of sensors, etc. and eventually, the response of the system. The system construction is given in Figure 1.

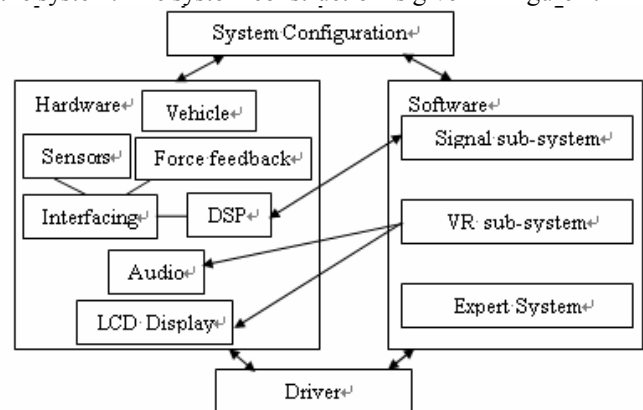


Figure 1. System Construction

III. HARDWARE COMPONENTS

The infrastructure of the system is a cab from a real vehicle (a bus, a truck, or a car), with other electronical devices such as sensors, force feedback devices, a signal

process microcontroller, preamplifiers, filters, MUX, timer circuit, sample/hold, D/A and A/D conversion circuit in a PCB. This PCB will do all signal processing which comes from or sends to sensors/force feedback devices. All sensors are interconnected with CAN-Bus. They collect all needed parameters about the vehicle such as velocity, ON/OFF of all lights, positions of brake, gears, accelerator pedal, angle of steering wheel, etc.. The hardware components in this PCB are given in Figure 2[1].

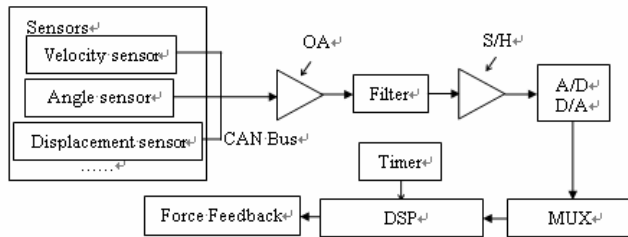


Figure 2. Hardware Components of the System

A. Sensour

Sensors can be used to detect displacement, temperature, strain, force and light, and so on. Sensor is a type of transducer which can transfer the physical quantity to electrical signal[1]. There are a great variety of sensors used in vehicle. In our virtual training system, we mainly use the following types of sensors, shown in Figure 3.

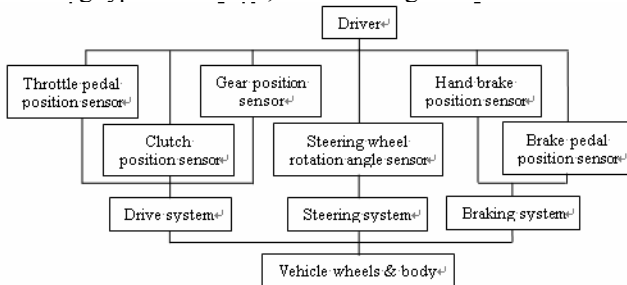


Figure 3. Sensors Used in the System

B. CAN-Bus

CAN (Controller Area Network) is sponsored by BOSCH and now is widely accepted as an industry standard for interconnection of vehicle-mounted sensors. It is a serial communications network that supported distributed and real-time access effectively. Normally, DeviceNet protocol and CANopen protocol based on CAL (CAN Application Layer) are used in a CAN network to support the direct access of equipment parameters and process data communication in real time. All kinds of information from sensors mounted all over the vehicle connected to a CAN can be transported to CAN-Bus very fastly, effectively and accurately without interferenced.

C. Discrete Signal Processor (DSP)

Sensors collect all signals from a vehicle, such as angle signal of steering wheel, position signal of gear, position signal of hand brake and braking pedal, position signal of clutch, position signal of accelerator, velocity of the vehicle, signal of turn lights, headlights, taillights and fog lights.

Some of these signals are ON/OFF signals, while the others are continuous signals. But all these signals are analog signals. They will be amplified, filtered, sampled, and at last be converted into digital signal for further processing by DSP (Discrete Signal Processor). And only after that, these digital signals can be accepted by IO interface of computer and then the software system. The software system will analyze these signals to interact with VR system. On the other hand, the interaction between vehicle and VR system (for example, ups and downs roads, winding roads) are translated into digital signals. These signals will be sent to DSP through the computer IO interface for further processing and hereafter to be converted into analog signals in the reverse data path and at last control the devices mounted on the vehicle, such as force feedback devices.

D. Force Feedback System

Force feedback in this system is to simulate the real force from steering wheel, accelerator pedal, brake pedal and clutch pedal, gear handlebar etc. in real driving and will adjust the amount based on the VR scenes such as muddy or dry road, turn radius, etc[2]. All these forces are parameterized first in a configuration file and therefore VR sub-system is parameterized accordingly.

IV. SOFTWARE COMPONENTS

The software consists of the following parts: VR basic component library, script generation engine, script parser, random parameters seeds generator, fractal and particle system, render engine, DBES expert system. The scene contains some regular objects from the VR basic componet library. And Fractal and Particle system generates irregular rain, snow, hailstone and fog, 3D terrains as other graphic systems based on the script generated by script generatin engine. The Render engine renders the scene and outputs it to display. The DBES sub-system will self-learn the drivers' behaviors to construct its rules database and in turn to judge if the driver's behavior is effective or not. The structure of software is shown in Figure 4.

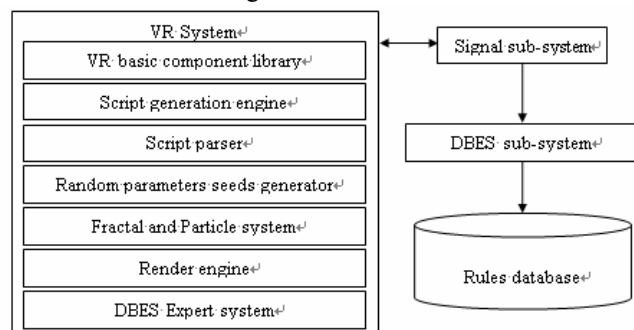


Figure 4. Software Components of the System

A. VR Basic Component Library

In this system, a VR basic component library populates some common regular models such as roads (straight, winding, ramp, bridge, etc.), buildings, traffic signs, other vehicles and pedestrians to construct the basic scene very

easily. This library can be expanded by creating new components or imported from other modelling softwares.

B. *Script Generation Engine*

When the system is configured to a special vehicle and a special scene, the system configuration module will invoke the Script Generation Engine to generate a random scene in this special type automatically according to special syntax rules in Script Parser. This script can be created or modified manually as long as the syntax rules are totally obeyed.

C. *Script Parser*

Scripts generated by the Script Generation Engine or by users are all synthetic language. To improve the performance of Render Engine, the scripts should be interpreted or compiled or parsed into a certain formatted binary file that contains many kinds of data structures which describe the whole scenes in an effective way and meaningful to Render Engine.

D. *Parameters Seeds Generator*

This module is to randomize the scene generated by parsed scripts. After randomizing, driver will interact different scenes even the configurations are same. The purpose of this is to avoid the driver always taking the same operations in a rigid scene and the training is meaningless afterwards. Furthermore, for getting more realistic effects, some objects generated by fractal and particle system are needed to be randomized too. Therefore this module will provide the random seeds to fractal and particle system.

E. *Fractal and Particle System*

Many natural objects, such as leaves, grass, hills, coastlines, etc., have irregular appearances and are very hard to modeled. But all these objects have a very prominent character, self-similarity and it's the professional scope of fractal system modeling. Fractal algorithm is only based on initial condition and proceeding morph algorithm, and perhaps the randomizing when morph.

On the other hand, some objects such as rain, snow, water-drop, flame, etc., have totally random distribution of parameters, for example, shape, direction of motion, velocity and life cycle. In particle system, the actual graphic attributes of the object is irrelevant and the essential is particles' random Brownian movement. In particle system, graphical attributes (shape, size, color, transparence, ...) and motion attributes of an object are all randomized. Random algorithm can find its usage here one more time.

Both fractal and particle systems have an inherent character which is very useful if computer graphic is concerned: objects created by these two algorithms are very easy to be LODed (Level Of Detail). LOD is a very useful technique to reduce details of a scene and hence improve the render performance greatly by modeling an object in different levels of detail and only render the low level detail when this object is not in the focus of the scene or in a long distance from the viewpoint.

F. *Render Engine*

After all objects are created and randomized, the whole scene defined by the parser is ready to be rendered by the render engine. The render engine is similar to render engines in other modeling softwares except it has a front-end to pre-process the customized binary data structures which defines all objects in a scene. It has no need to be discussed very detail here. At last, this scene is output smoothly on LCD display.

G. *DBES*

The DBES expert system works in two phases: firstly it self-learns drivers' behavior if it's effective to construct its rules database; secondly, it analyses the results of driver's behavior, and judges if these behaviors are effective or not in these given scenes, and if a behavior is effective but not in the rule database, the DBES will add it into the database too. As it's same with other expert systems, it's discussed here only for completeness.

V. SUMMARY

This paper introduces the concepts of a driving training system for exceptional scenes and the software and hardware construction with a new usage of VR system. This system combines software technologies such as virtual reality and hardware technologies such as sensors and computer interfaces, simulates some complex road conditions and exceptional scenes and the interactions between driver and VR scenes.

REFERENCES

- [1] Stephen E. Derenzo. Practical Interfacing in the Laboratory: Using a PC for instrumentation, data analysis, and control, China Machine Press, Apr., 2006, PP: 1-3.
- [2] JING Xu, CHAI Shan. Research and Development of Vehicle Virtual Driving System Based on Virtual Reality Technology, Shandong University Of Technology, Master's Thesis, Apr., 2007, PP: 73.