Simulation System for Teleoperated Mobile Robots

Tomáš Kot, Václav Krys, and Petr Novak

VŠB - Technical University of Ostrava, Department or Robotics, Ostrava, Czech Republic
{tomas.kot,vaclav.krys,petr.novak}@vsb.cz

Abstract. The paper presents a mobile robot simulator designed primarily for teleoperated mobile robots, to aid in the prototyping phase of development or as a training tool for existing robots. The simulator uses virtual reality with realistic 3D graphics to simulate complex behaviour of a mobile robot in various conditions and with different properties. The system can be used to discover or verify driving and manipulation abilities of a mobile robot defined by its kinematics structure and dimensions, and also to simulate navigation using the camera subsystem containing one or more cameras.

Keywords: mobile robot, simulation, teleoperation, navigation, virtual reality.

1 Introduction

The area of service robotics has rapidly developed lately especially in terms of the scope of applications. Mobile service robots are relatively often used not only for operations associated with safety engineering such as firefighting, chemical accidents, terrorist attacks, disposing of explosives, surveying of hazardous and cramped areas, searching for earthquake victims, but also in the commercial and science sectors, e.g. checking and maintenance of piping, exploration of planets, etc. Also military applications cannot be neglected.

Vast majority of the above applications use mobile robots remotely controlled by a human operator. Compared to autonomous robots, this is still the cheaper and above all more reliable solution. Algorithms of robot artificial intelligence and sensory systems still have not reached the level when it would be possible to use a robot for tasks when human lives are at stake.

Owing to the immense range of possible applications, the mobile robotics field is subject to continuous development and innovation. The development process of a new mobile robot requires a lot of time and funds, and therefore it is convenient to use virtual prototyping – in an ideal case the resulting first physical prototype developed would meet all requirements and no additional modifications would be required. As for the mechanical construction of a robot, the existing CAD/CAM systems may be used. However, these systems usually do not allow complex simulation of behaviour of the whole mobile robot in the conditions similar to reality. So a situation may occur when a physical prototype would be fully operational in terms of its mechanical aspects but due to an unsuitable concept, the mobile robot would be difficult to control

J. Hodicky (Ed.): MESAS 2014, LNCS 8906, pp. 164–172, 2014.

in the required conditions, the camera subsystem would not provide the operator with sufficiently clear images, and suchlike.

The possibility to carry out extensive testing on a virtual prototype of a whole robot in a simulated real environment may make the whole development process much faster, much more efficient, and especially cheaper. The virtual model may be further used even when the robot is physically manufactured, fully tuned, and used in practice – the virtual model may be used for training of operators without the necessity to use the actual robot.

2 Basic Structure and Properties of the Simulator

There is a great number of simulation systems available, for example [1-7]. However, most of them focus on testing of various algorithms of autonomous behaviour, and therefore they offer advanced possibilities of programming of robot control systems. However, creation of a programme code is also necessary even for basic tasks, unless one of the ready-made mobile robots based on actual commercial robots is used. It is quite difficult to install and commission the systems, which is given particularly by their composition of various more or less independent modules and libraries. The primary view of the 3D simulated scene is from top, but there usually is a possibility to define virtual robot cameras and use them to monitor the scene. The graphic output is average, created using basic principles and simple shadowing is usually the most advanced effect.

RoboSim simulation system described in this article tries to excel compared to the competition of the already existing systems particularly in the following areas:

- Short time needed to start working with the system, application with ease of use even for designers and other professionals not skilled in programming.
- Possibility of quick verification of the concept and kinematics of the mobile robots plus easy modifications.
- Advanced simulation of virtual cameras, superior virtual presentation for more realistic feel for operators when controlling virtual robots.
- Specialization in verification of the ease of operator control and navigation in an unknown or complex environment.

Simulation requirements may be divided into two large groups:

- **Movement physics** (driving properties, handling in various terrains, stability, ability to avoid various obstacles, turning radius and manoeuvrability in general, speed, power, etc.).
- **Navigation** (ability of the operator to control the robot using only the feedback from the cameras and sensory subsystem, verification of the required quantity, placement and the type of cameras, benefits of stereovision, etc.).

The application is based on our own application core and is programmed in Visual C++. The Direct3D application programming interface (API) is used for graphics [8] and the Havok engine for rigid objects physics simulation [9].

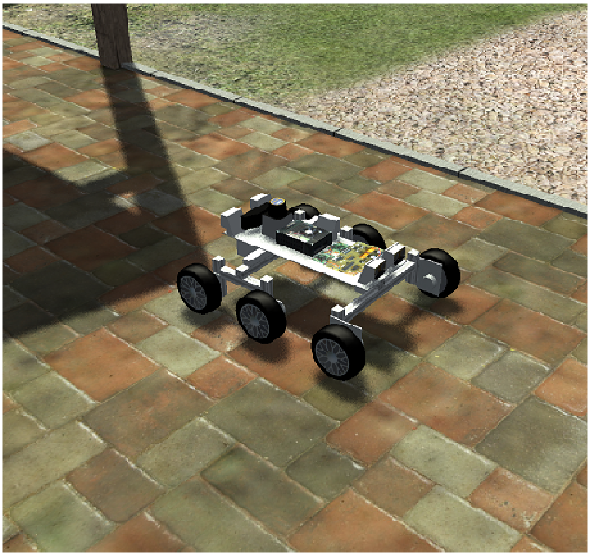

Fig. 1. Six-wheeled mobile robot displayed in the simulator

The rendered virtual scene consists of a greater number of separate entities (these are usually separate bodies), where each of them has its visual and physical properties [10,11]. Various relations and dependencies may be defined between the entities. Rendering of entities is optimized for maximum speed and therefore suitable filtering is applied according to visibility and the rendered objects are also ordered for more efficient calling of Direct3D functions. Hardware acceleration of rendering is a matter of course. Therefore, a graphic chip supporting at least Shader Model 3.0 (DirectX 9.0c) is required to run the application.

3 Kinematics and Dynamics of Movement

Comprehensive simulation of Newton physics of a mobile robot is achieved through proper use of the above-mentioned Havok engine. The mobile robot is divided into individual parts (entities) with suitably adjusted physical properties. The enti-ties are then connected by appropriate physical links and action parts (e.g. driven wheels) are complemented with physical actions (drives). When the mobile robot defined this way is placed in a virtual environment also containing entities with a physical component (at least a floor or ground), complex simulation of robot behaviour is secured.

The current version of the simulator uses only rigid body dynamics from the Havok library, and therefore it is impossible to simulate e.g. flexible wheel tyres. However, simulation of springs for sprung wheel suspension is possible. Each rigid body (entity) has particularly the following properties:

- weight, centre of gravity position, inertia moments,
- shape for detection of collisions,

- restitution (degree of energy loss upon collision),
- friction,
- linear and angular damping (degree of energy loss when moving without effect of external forces).

Body shape is used for detection of collision with other objects and calculation of correct reaction to these collisions. Calculation of collisions is an operation whose complexity rises significantly with the complexity of body shapes. Therefore, it is desirable to define the collision shape of the body as simple as possible while preserving sufficiently accurate behaviour. Therefore, in most cases a different shape is used for visual representation of an object (rendered triangular mesh with many details) and for collisions (Fig. 2). Examples of some shapes ordered starting with the least demanding: sphere, capsule, box, cylinder, general convex body, general concave body.

Fig. 2. Comparison of detailed visual representation and simplified collision

Restriction of movement of two bodies against each other may be achieved by links. Havok allows definition of new arbitrarily complex links but it also contains ready-made links of which the following are the most suitable for undercarriages, handling superstructures or other parts of mobile robots: *hinge* – axial rotation, *limited hinge* – restricted rotation, *wheel* – rotation round one axis, optional rotation round another axis (steering) and optional translation along a third axis including a spring and dampening (wheel suspension) and *prismatic* (translation).

Behaviour of objects in a certain way is affected by *actions*. Analogous to links, new actions also may be easily defined according to actual needs. Rotational and translation motors in particular belong to the prearranged actions. A new link was created for the needs of RoboSim, simulating behaviour of a DC motor including characteristic dependence of torque on speed.

4 Simulation of Camera and Sensoric Subsystems

With regard to the primary designation of the simulator for testing of operator controlled robots particularly when using a feedback from a camera subsystem, great attention was paid to correct simulation of cameras.

The simulation system allows fitting of the robot with an arbitrary number of virtual cameras, which may be used for monitoring of the robot surroundings. These cameras may be visually rendered as a 3D model (Fig. 3).

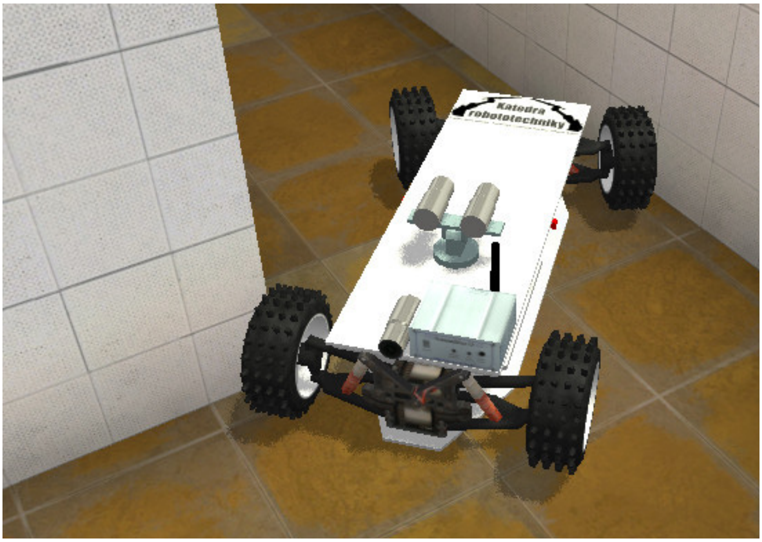

Fig. 3. Virtual mobile robot with 2 stereovision cameras and one rear camera

Each camera needs to have specified the basic optical properties – viewing angle, resolution in pixels, or even for example parameters of barrel distortion of the image.

The actual optics of cameras usually does not have sufficient depth of field to cover large distance range of viewed objects, and therefore focusing is implemented. The objects outside the focused depth of field then appear to be blurry. This effect may substantially affect operator's ability to navigate in a complex environment using only the cameras and therefore RoboSim includes simulation of depth of field (Fig. 4). When rendering a scene, this visual effect is achieved using a special *pixel shader*. It is not an exact physical simulation of this optical phenomenon but only its rough and simplified approximation.

When using *stereovision* it is necessary to specify a pair of cameras with suitable mutual position and identical optical parameters (Figure 3). Stereovision may then be viewed in the following ways:

- *Anaglyph* – encoding images for individual eyes to different colours and subsequent filtration using simple passive spectacles with coloured glasses through which the monitor is viewed [12].
- *Head-mounted display (HMD)* – various types of active displaying devices placed directly on user's head [13]. Two general methods of displaying are supported – interlacing and image alternating.
- *Oculus Rift* – new HMD with exceptional properties and very low price. This device requires a specific way of rendering when the images for both eyes are placed

next to each other and software barrel distortion is applied to them to eliminate the opposite distortion created subsequently in the device optics [14].

Fig. 4. Depth of field effect

Simulation of sensors is only simple in the current version of *RoboSim*. There are three types of sensors available – *linear* (measuring linear distance to one point of an obstacle, simulation of IR and suchlike sensors), *conical* (measuring distances in a cone and returning the smallest value, simulation of ultrasound sensors etc.) and *planar* (measuring distances in a fan pattern on a plane and providing a set of measured values, simulation of laser scanners). All sensors are sending imaginary mathematical beams and detect intersections with object in the scene. In order to prevent unrealistic absolute accuracy, it is possible to add random noise.

5 Realistic Testing Environments

Besides the possibility to define completely new environments, RoboSim also of-fers three predefined scenes covering various conditions of possible applications of robots – a family house (complex interior with narrow spaces and a large number of obstacles, Fig. 5), an outdoor scene (open space, uneven terrain with various slop-ing, Fig. 1, 4) and a special testing laboratory (various exactly defined sizes of corridors, door openings, obstacles, stairs, inclined planes, ramps, etc., Fig. 2, 3, 6).

In order to provide the most authentic feel of the view available from the mobile robot cameras, the testing environments are created with emphasis placed on the visual quality and details using some advanced methods to achieve realistic 3D graphics. Common graphics applied in CAD systems and in most of the existing simulation systems uses only simple dynamic lighting and possibly textures. The resulting uniform colour surfaces and repeating textures do not provide the operator with sufficiently authentic image when testing navigation using the camera subsystem and the result may be biased due to this fact – it is much more difficult for the operator to find his/her way in the virtual environment than in reality.

Fig. 5. One of the rooms in the family house testing environment

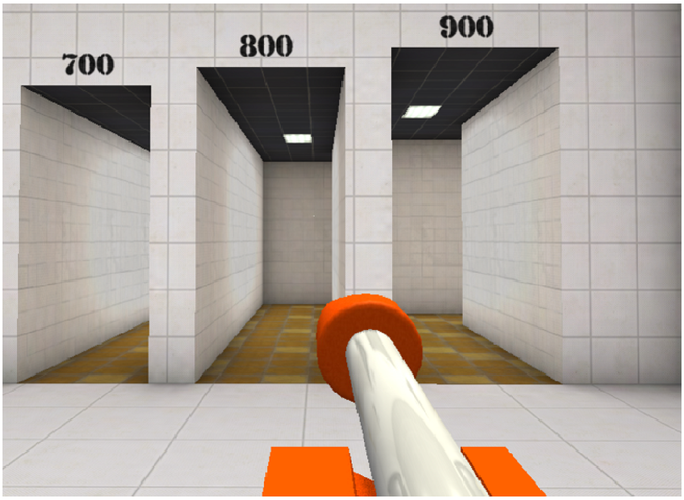

Fig. 6. Demonstration of the advanced lightning model used in the simulator

The advanced techniques used in RoboSim include for example:

- *Shadows* – very important for understanding the mutual spatial relations between objects. The used technique is shadow mapping including an optional degree of smoothing (Fig. 1).
- *Lightmapping* – replacement of real-time dynamic lighting by much more accurate pre-calculated model simulating even reflected light (walls on Fig. 6).
- *Environment mapping* – simplified simulation of highly reflective materials (robot arm on Fig. 6).
- *Normal and specular mapping* – simulation of uneven and diverse surface using a suitable lighting calculation without the necessity to create a very complex 3D model (Fig. 1).
- *Multitexturing, texture layers* – alleviating of undesirable effect of repetition of texture patterns, adding of imperfection, impurities, etc.

6 Conclusion

To compare simulation with reality, virtual models of several actual robots were created and their behaviour was tested (Fig. 7). Although the simulation of dynamics in real time implemented by Havok system is quite realistic, it is not fully physically accurate, which is given by the necessary simplifications in order to allow sufficient calculation speed.

The basic simplification directly concerning wheeled mobile robots is the replacement of simulation of soft rubber of wheels (or even air filled tyres) with ideally rigid

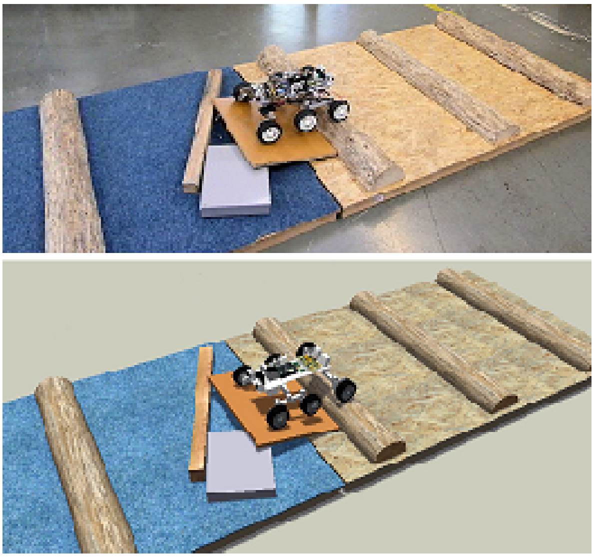

Fig. 7. Mobile robot on an obstacle course in reality and in the simulator

objects. Neither is it possible to simulate accurately the tyre tread and so the possible shape contact of the wheel with an obstacle is replaced by mere friction of a cylinder against an obstacle. Despite that, the behaviour of robots in the simulation is depicted relatively accurately in particular in terms of manoeuvrability and ease of control.

The strongest point of *RoboSim* – simulation of camera subsystem – brought several interesting findings even during the development of the simulator. It was successfully used for optimization of camera position on a developed mobile robot and this SW was also used for testing of stereovision and particularly its usability for navigation.

The application is currently being extended by a more convenient and fully graph-ical editor of mobile robots and testing scenes and adding of simulation of tracked chassis would be also desirable. There is also a plan for the possibility to control the virtual robot in the simulator using the operator's station for the actual robot, which instead of the real robot would establish connection with the PC with the running simulation.

Acknowledgment. This article has been elaborated in the framework of the project Opportunity for young researchers, reg. no. CZ.1.07/2.3.00/30.0016, supported by Operational Pro-gramme Education for Competitiveness and co-financed by the European Social Fund and the state budget of the Czech Republic.

References

1. Microsoft MSDN. Microsoft Robotics Developer Studio, http://msdn.microsoft.com/en-us/library/bb905420.aspx
2. The Player Project. Gazebo, http://playerstage.sourceforge.net/wiki/Gazebo
3. CyberRobotics Webots 6, http://www.cyberbotics.com/
4. Universität Leipzig. Research Network for Self-Organization or Robot Behavior, http://robot.informatik.uni-leipzig.de/
5. Virtual Robot Experimentation Platform, http://www.v-rep.eu/
6. OpenSim, http://opensimulator.sourceforge.net/
7. Simbad Project Home, http://simbad.sourceforge.net/
8. Microsoft. Direct3D 9 Graphics <Literal>http://msdn.microsoft.com/ cs-cz/library/bb219837(en-us,VS.85).aspx</Literal>
9. Havok, http://www.havok.com/
10. Eberly, D.H.: 3D Game Engine Design: A Practical Approach to Real-time Computer Graphics, 561p. Morgan Kaufmann Publishers (2000) ISBN 978-1558605930
11. Gregory, J.: Game Engine Architecture, p. 860. AK Peters, Ltd., Massachusetts (2009) ISBN 978-1-56881-413-1
12. Wikipedia. Anaglyph image, http://en.wikipedia.org/wiki/Anaglyph_image
13. Wikipedia. Head-mounted display, http://en.wikipedia.org/wiki/Head-mounted_display
14. Oculus VR. Oculus Rift, http://www.oculusvr.com/rift/
15. Kot, T., Mostýn, V., Novák, P.: Application of Virtual Reality for Verification of Characteristics of Mobile Robots. In: Proceedings of the ICMT 09 - International Conference on Military Technologies, pp. s.517–s.524. University of Defence, Brno (2009) ISBN 978-80-7231-649-6