Fire Safety Journal 50 (2012) 12–24

Contents lists available at SciVerse ScienceDirect

Fire Safety Journal

journal homepage: www.elsevier.com/locate/firesaf

结合火灾动力学数据的基于虚拟现实的火灾训练模拟器

A virtual reality based fire training simulator integrated with fire dynamics data

Moohyun Cha [a], Soonhung Han [b,*], Jaikyung Lee [a], Byungil Choi [c]

[a] Mechanical Systems Safety Research Division, Korea Institute of Machinery & Materials (KIMM) 171 Jang-Dong, Yuseong-Gu, Daejon 305-343, Korea
[b] Division of Ocean Systems Engineering, Korea Advanced Institute of Science and Technology (KAIST) 335 Gwahangno, Yuseong-Gu, Daejeon 305-701, Korea
[c] Environmental and Energy Systems Research Division, Korea Institute of Machinery & Materials (KIMM) 171 Jang-Dong, Yuseong-Gu, Daejeon 305-343, Korea

ARTICLE INFO

Article history:
Received 7 April 2011
Received in revised form
6 January 2012
Accepted 24 January 2012
Available online 23 February 2012

Keywords:
Virtual reality
Training simulator
CFD (Computational Fluid Dynamics)
Fire simulation

一系列的数据转换技术和实时处理框架开发一个消防训练模拟器的基础上精确的CFD模拟,能计算各种无形的物理量如有毒气体和热量以及可见的烟和火焰等因素

ABSTRACT

VR (virtual reality)-based fire training simulators provide the general public or inexperienced firefighters or commanders with wide-ranging second-hand experience so that they can make prompt decisions and safe and organized responses in actual fire situations. In order to effectively achieve this training goal, it is crucial to reliably express fire dynamics as realistic graphics. In the field of engineering, computational fluid dynamics (CFD) is widely used to precisely predict the behaviors of fluid phenomena. The resultant data, however, have structures and capacities that are not readily applied to real-time virtual reality systems. This study proposes a series of data conversion techniques and a real-time processing framework to develop a fire training simulator on the basis of a precise CFD simulation that is capable of calculating various invisible physical quantities such as toxic gases and heat as well as visible factors such as smoke and flame. By exploiting safety level-based visualization mapping, this study also proposes a new method to intuitively experience dangerous fire environments and perform training and evaluation. Lastly, this study implements a simulator that can undertake simple firefighting activities such as evacuation and rescue in fire situations at road tunnels; the functions and real-time performance of the simulator have been experimentally measured to verify the applicability of the proposed framework.

1. Introduction

Fire situations, especially those in closed spaces used by a large number of people, entail significant constraints in firefighting and rescue activities due to the rapid growth of fire and difficulties related to ventilation and evacuation route provision. To minimize casualties and property damage under such dangerous situations, it is essential that firefighters experience various fire situations and undergo training to improve firefighting responses. Full-scale fire training where actual fire situations are reproduced, however, involves enormous social/economic costs, as buildings or roads should be shut down or new structures built for training purposes. Furthermore, toxic gas poisoning or structure collapse during the training process may potentially lead to casualties.

The recent, groundbreaking development of virtual reality (VR) technologies has given rise to computer–human interaction technology that allows actual users to participate in a virtual world reproduced by computers. This highlights the need for VR-based training simulators enabling safe, convenient, and planned repetitive training. Training simulators, in general, should have basic functions such as scenario generation and control, virtual reality content representing actual situations, immersive virtual reality interface devices, and the capability of evaluating the training process and results. In particular, realistic virtual reality content should be expressed on a real-time basis under the given training scenarios depending on the locations and viewpoints of users' choice. The content is offered to trainees via immersive interface devices such as a HMD (Head-Mounted Display), large screen, and tracker to maximize the sense of presence and interactivity; trainee responses and mission fulfillment data are recorded and analyzed for overall training evaluation.

In the case of fire training simulators that mimic fire situations and provide virtual experiences and training, the dynamic

Abbreviations: ADMS, Advanced Disaster Management Simulator; ASCII, American Standard Code for Information Interchange; CFD, Computational Fluid Dynamics; CPU, Central Processing Unit; CSV, Comma-Separated Values; DXF, Drawing Exchange Format; FCO, Fire Company Officer; FCTVE, Firefighter Command Training Virtual Environment; FDS, Fire Dynamics Simulator; FPS, Frames Per Second; GB, Giga Byte; GPU, Graphics Processing Unit; GUI, Graphical User Interface; HMD, Head-Mounted Display; HSV, Hue, Saturation, and Value; LOD, Level Of Detail; MW, Mega Watt; OGRE, Object-Oriented Graphics Rendering Engine; SOP, Standard Operating Procedure; SSD, Solid-State Drive; VE, Virtual Environment; VR, Virtual Reality

* Corresponding author. Tel.: +82 42 350 3040; fax: +82 42 850 1510.
E-mail addresses: mhcha@kimm.re.kr (M. Cha), shhan@kaist.ac.kr (S. Han), jkleece@kimm.re.kr (J. Lee), cbisey@kimm.re.kr (B. Choi).

0379-7112/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.firesaf.2012.01.004

behavior of smoke and heat serves as valuable data in fire responses such as evacuation, rescue, and firefighting. An important function of these simulators is to express such data as realistic graphics to provide trainees with second-hand experience on dangerous situations that they can generally not experience directly so that they will not feel intimidated or confused in real fire situations. While the aforementioned four functions are crucial for fire training simulators, the simulation of precise fire dynamics corresponding to actual fire situations and the realistic representation of these phenomena into real-time graphics are especially important.

Simulation technologies for fluid flows (e.g. spread of gas by fire) can be broadly divided into: (i) real-time simulation technology that offers computation speed of more than dozens of times per second and is directly applicable to real-time simulators; and (ii) non-real-time simulation technology that requires significant computation time and thus cannot be directly applied to simulators.

The representative real-time simulation technology is particle system technology, which expresses fluid flows into movements of fine particles [1–3]. The method predicts the trajectories of particles considering basic Newton's laws of motion (e.g. amount of release, speed of release, gravity field, and life cycle); it is widely used in general games to express flames or smoke, as the amount of calculation is relatively small and real-time calculation is possible [2]. Accurate prediction, however, is impossible here, as pressure differences by structure or heat, which play a crucial role in fire-driven fluids, are not taken into account; fire simulators may only be utilized for limited uses such as the recognition of fire occurrence.

Meanwhile, some researchers have applied real-time physical simulations to three-dimensional fluids using GPU (Graphics Processing Unit), the performance of which has recently been enhanced dramatically [4–6]. To predict three-dimensional fluid motions, the Navier–Stokes equation is used within discrete grids to perform a numerical simulation, but this requires enormous amounts of calculation. For this reason, these researchers have simplified fluid models to inviscid and incompressible models to stably simulate the Navier–Stokes equation in real-time; they have also omitted the heat transfer process and employed a parallel computation-optimized GPU to obtain real-time performance [6]. Researchers have focused on the reality of visual appearance rather than on precise engineering, and they have thus far covered local fluid effects in relatively small calculation space. As a result, their findings are inappropriate for the expression of massive gas flows where the training space is relatively large in size and the speed of the fire is altered by heat or ventilation facilities.

Meanwhile, non-real-time fluid simulation technology, as explained above, precisely predicts fluid behaviors using the Navier–Stokes equation, the governing equation of fluid dynamics. The equation takes into consideration the conservation of momentum, mass, and energy for a viscous and heat-conducting fluid at the same time, and thus wide-ranging physical quantities related to fluid behaviors can be reliably calculated. The technology has been developed by engineering researchers in the field of computational fluid dynamics (CFD) over a long period of time and has been used in the design/analysis of various mechanical systems.

The most representative non-real-time fire simulation model is the field model-based fire dynamics simulator (FDS), which can analyze fire-driven fluid flow on a three-dimensional basis [7–8]. Developed by the National Institute of Standards and Technology (NIST), FDS quantitatively predicts almost all fire-related physical quantities such as toxic gas and temperature as well as fire development and smoke propagation over time. Having a margin of error of 5–20% from actual experimental values, it is widely applied to various fields of research such as fire accident analysis, architecture design, and disaster prevention planning [9–11]. Such CFD-based simulation requires considerable computation time and the results are large-volume data amounting to dozens of GB, and consequently additional technology for data processing is essential to apply this to real-time simulator systems.

To visually analyze these CFD numerical simulation results, either specialized post-processors loaded onto each CFD solver or general-purpose scientific visualization tools such as VTK [12], Paraview [13], OpenDX [14], and Ensight [15] are used in the field of engineering. They provide meaningful information for engineers via Vector Field, Iso-Surface, Streamline, and other various visualization methods, but they generally do not offer realistic visualization environments—as in the case of virtual reality training simulators—for the general public to experience physical phenomena intuitively. Furthermore, it is extremely difficult to photorealistically visualize the components of the virtual world (e.g. structures and characters) as well as physical phenomena, reflect training scenarios, and comprehensively integrate VR interface devices with them.

This study precisely predicts fire growth and fluid flow with a non-real-time CFD simulation and applies the results to real-time systems to design and develop a virtual reality based fire training simulator. For this purpose, the most widely-known FDS was employed to simulate wide-ranging fire-related physical quantities offline. We also propose a real-time simulator framework that include a series of data handling processes such as data structure conversion, space partitioning/search, memory management, three-dimensional volume rendering, and training evaluation for the secured large-volume data. This study also implemented a simulator that can undertake simple firefighting activities such as evacuation and rescue in fire situations at road tunnels; the functions and real-time performance of the simulator were experimentally measured to verify the applicability of the proposed framework.

One major contribution of this study is that it proposes a framework for interfacing reliable three-dimensional CFD simulation results into real-time virtual reality systems so that they can be utilized in the development of similar, engineering-based training simulators. The expression of precisely predicted fire dynamics into realistic graphics can be used to ensure emergency-response capabilities, and relevant knowledge can be obtained intuitively on the basis of visual experience. This study proposes a safety level-based visualization mapping method to visualize risk factors such as toxic gas and temperature distribution that are invisible but are closely correlated with the safety of trainees. In addition, the pre-calculated physical quantities may be used not only for visualization purposes but also for measuring safety achievement levels in the space where trainees pass through. On this basis, this study proposes a trace-based training evaluation method and applies it to the simulator system.

2. Related works

This section summarizes the objectives and functions of the virtual reality visualization system supporting fire training. The fire simulation and graphic expression methods, in particular, are compared with those proposed in this study.

Many research results related with virtual reality-based fire training environments have been reported. Most studies have emphasized the benefits of this approach over conventional training methods, including improvement of firefighting performance without risking lives or property. Tate et al.[16] reported promising results of virtual environment (VE) as a tool for shipboard firefighting training and mission rehearsal. Serious game-based training methods and simulators have also been proposed to increase personal fire safety skills while performing evacuation scenarios [17], to provide a self-learning method to search for and rescue victims [18] and to give children realistic experiences in high-risk safety training [19]. They used game technology, the so-called serious-game, not only for entertainment or self-motivating experience but also for contributing to the achievement of a defined training purpose. These researches showed that virtual reality techniques can play an important role in fire safety training; however, simulation techniques that generate realistic fire or smoke behavior in a virtual world were not used or addressed.

Fire simulation was first applied to virtual reality environments in the Walkthru-CFAST system developed by Richard et al. [20]. Prior to the development of CFD-based FDS, the zone model CFAST (consolidated model of fire and smoke transport) was widely used to predict the impacts of various fire types on temperature, gas concentration, and smoke layer height in multiple-roomed architectures [21]. As only four differential equations are solved in one compartment, the calculation speed is fast, but geometries other than squares or complex, three-dimensional fluid phenomena could not be expressed. The Walkthru-CFAST system applied this CFAST fire simulation program to existing virtual reality platforms to derive experience in fire hazard situations in various architectures (e.g. buildings). The authors proposed an interface between physical quantity-based scientific visualization (e.g. smoke visualization and temperature visualization modes) and virtual reality systems, but the computing environment and graphic technologies employed were somewhat antiquated; the employed fire prediction method is not being widely used in simulator systems presently due to the limitations of the CFAST zone model.

Based on a house fire scenario, Julien et al. developed a firefighter command training virtual environment (FCTVE) system so that trainees can instruct avatars to follow predefined commands for firefighting and human rescue activities and the avatars can fulfill such missions under virtual realistic fire situations [22]. The study provided decision-making training for fire company officers (FCO) who have extensive experience with and expectations of fire behaviors, and thus realistic simulation of fire and smoke was very important. For this purpose, a FDS-based numerical simulation was carried out and volume rendering undertaken on fire dynamics data to express virtual flames and smoke. This can be viewed as the very first case of applying three-dimensional CFD results to training simulators. However, data processing technologies such as space partitioning were not employed in order to guarantee real-time performance of handling massive data; and numerical physical quantities were used to express only the visible spreading of smoke.

Based on an underground station fire scenario, Ren et al. developed a virtual reality system to provide experience in fire situations and perform evacuation training [23]. As in previous studies, a FDS-based fire simulation was carried out to predict fire situations reliably, and the results were saved in a database. The real-time system retrieved physical quantities such as smoke concentration and heat from the simulation database, but the data were not used directly for volume rendering purposes but rather employed only as parameters (e.g. fire source location, number of particles, speed of release, and life) for modeling the particle system of a commercial VR engine. This helped secure real-time performance—a major advantage of the particle system. However, the benefits of the CFD simulation could not be maintained, as the precisely simulated data were mapped into simple physics models. For such particle systems, it would be more meaningful to utilize particle physical quantities among FDS result data.

Developed as an FDS-specialized post-processor, SmokeView offers a visualization function for engineers to interpret all types of massive amounts of data analyzed by FDS and also realistically expresses 3D smoke. It also provides a simple navigation function [9,24–25]. The main types of visualization in SmokeView are particles, slices, isosurfaces, and realistic smoke. Particles visualize the direction and speed of soot particles, slices visualize the distribution of physical quantities in a 2D plane, isosurfaces visualize the same level of values as 3D surfaces, and realistic smoke visualizes the soot density distribution of smoke in a 3D space using volume rendering. With the exception of volume rendering of 3D smoke, it is difficult for individuals lacking sufficient engineering knowledge to grasp or experience fire behavior with these visualization methods in a virtual reality environment. The main goal of the system is to analyze engineering data; SmokeView is thus not easily applicable to the development of actual simulator systems, because it lacks essential basic functions for the VR simulator (e.g. realistic graphics for 3D geometric models and avatar control), as in the case of the aforementioned scientific visualization tools.

Meanwhile, ADMS (Advanced Disaster Management Simulator), a widely known commercial solution, is a virtual reality training system developed by etc (Environmental Tectonics Cooperation) to provide training on how to respond to disasters/accidents such as terrorism and fire [26]. The National Fire Service Academy of Korea has also adopted this system to train its students how to allocate firefighting resources (e.g. equipment and personnel) efficiently and how communicate by the unit of teams. The commercial software is designed to help firefighting commanders set up firefighting plans, make decisions, and learn the command system—rather than providing fire experience for firefighters—and thus has a simple particle system for fire simulation and graphic expression to ensure the visual recognition of fire accidents.

Unlike in previous works, as the users of the proposed simulator system this study selected members of the general public that wish to experience fire situations directly, firefighters that seek to undertake simple fire training, and firefighting commanders aiming to identify fire situations and set up firefighting plans. The goal of this study is to visualize reliable fire dynamics data for these training participants as realistically and intuitively as possible and thereby provide a virtual environment where the objectives of the training can be best achieved. For this purpose, an open-source graphic library is used to design and implement a virtual reality simulator enabling realistic rendering. FDS results are used to intuitively visualize the concentrations of toxic gases and oxygen in the air as well as smoke; space partitioning and other effective data processing methods are also proposed to visualize large-volume physical quantity data on a real-time basis.

3. System design

3.1. System architecture

This study first designed a fire training simulator system that can perform fire training using virtual reality. The VR engine generally interfaces the three-dimensional virtual reality model with the logics to undertake given tasks to control the virtual world. Participants of the virtual world interact with the VR engine and fulfill certain objectives using the interface module. This was mapped into the fire training system to structure the entire system, as shown in Fig. 1, and come up with the following four components:

1. Fire Training and Evaluation Logic: The logic is comprised of: fire accident scenarios including the origin of fire, fire type,

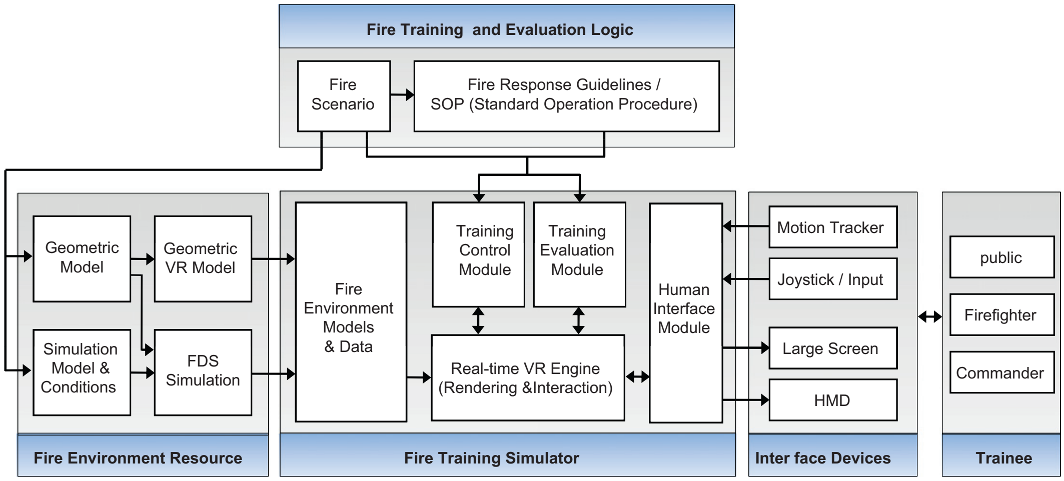

Fig. 1. System architecture of the fire training simulator.

environmental situation of the fire site, and resources and safety facilities available; guidelines for ensuring the right responses and commands for the purpose regarding the given fire situation; and standard operating procedures (SOPs) for firefighting activities.

2. Fire Environment Resources: The resources include the basic, static VR geometric model needed to build a virtual world and three-dimensional fire dynamics data calculated with CFD. Geometric models such as structures, facilities, tools, and vehicles in the accident site are produced on the basis of fire scenarios; a FDS simulation is carried out, using the fire types and conditions obtained from the scenarios and geometric models for CFD grid generation, and the results are utilized as "Fire Environment Models and Data" for the simulator.
3. Interface Devices between Human and Computer: The VR system involving human users requires interactive interface devices to maximize the sense of presence and immersion. The intentions of the participants are obtained from input devices such as a motion tracker and joystick; an immersive large screen or HMD is used to provide the visualized graphic content.
4. Fire Training Simulator: The simulator is basically run by the "Real-time VR engine", while the virtual environment generated from "Fire Environment Models and Data" interacts with users through the "Human Interface Module". The training and evaluation module implemented from "Fire Training and Evaluation Logic" controls the VR engine, records the training process and oversees the entire training process.

3.2. Data processing logic

The simulator system that this study develops is aimed mainly at realizing fire phenomena through the VR engine—with the real-time VR system as the basis—using precisely simulated fire dynamics data. Data calculated through CFD, in general, have a large-volume of physical quantities distributed in a three-dimensional discrete grid space, and FDS includes the gas concentration, heat release rate, and temperature distribution within the space. These physical quantity data should be expressed into intuitive graphics so that the general user can easily experience physical phenomena; the most effective technique here is volume rendering, which renders the optical properties of the given volume on a three-dimensional basis. General CFD data, however, have structures that cannot easily be used directly in real-time volume rendering. Furthermore, these time-variant large-volume data are comprised of hundreds of thousands of cells. For this reason, data structure conversion and processing technology are needed to render the data on a real-time basis. In addition, most of the casualties from fire are attributable to gas poisoning and oxygen deficit rather than to severe burns, and the visual experiences of these situations are very important in enhancing safety responses. In relation to this, lethal physical quantities other than smoke, however, generally do not have optical properties, and thus visualization technology is needed for these invisible data.

As for large-volume data processing and real-time visualization, extensive research has been made in the field of scientific visualization, which involves the study of efficient volume rendering technologies for large-volume, three-dimensionally distributed physical quantities (e.g. medical measurement data and CFD simulation results). Among these technologies, multiple resolution-based level of detail (LOD) selection hierarchically divides the given volume space to yield varying precision levels (i.e. resolution levels) and selects precision levels offered to the users in accordance with adequate criteria [27–28]. In other words, the technology improves real-time performance through acceptable data simplification. To implement this simplification method, Octree space partitioning, which expresses three-dimensional multiple resolution and is directly applicable to volume rendering [29–31], is generally applied to space tree data structures. Various optimization methods have also been introduced, including compression and frequency adaptive LOD selection [32]. To verify the functional applicability of the simulator, this study pre-processes CFD data using basic Octree space partitioning and applies LOD selection by viewpoint distance and data distribution to implement the real-time volume rendering function. Memory management technology involving partial data loading is also employed to improve the overall system performance.

For invisible physical quantities such as toxic gases, this study proposes a safety-based transfer function that maps visual attributes into levels of hazardousness toward the human body. The transfer function, used in the computer graphics area, extracts optical properties such as color and transparency from the physical quantities of the three-dimensional volume and converts them into basic data for volume rendering [33]. The main goal of visualizing the physical quantities of fire in training and evaluation is to experience/evaluate how fatal the impacts of the given volume are on trainees—mapping the visual attributes of the transfer function with human fatality levels thus holds greater significance than mapping the mere scalar values of physical quantities.

Generated through CFD, data on the distribution of physical quantities for space can be used not only in volume rendering for visualization but also in evaluations related to the safety achievement of trainees. In other words, current physical quantities surrounding the trainees are measured, and data on hazards exposed to the trainees throughout the entire training process are accumulated as basic data for the safety achievement evaluation.

As illustrated in Fig. 2, the proposed data processing logic is divided into: "Numerical Fire Simulation" for obtaining FDS output data based on the given scenarios; "Pre-Processing" for application to real-time virtual reality systems; and "Real-Time Processing" for undertaking real-time volume rendering and safety evaluation. The detailed sub-processes are specified in the following subsections.

3.3. Numerical fire simulation

In order to undertake a FDS-based numerical fire simulation, simulation models should be determined on the basis of given fire scenarios and grids generated with the geometric models for the calculation space. Simulation models include: a hydrodynamic model for turbulent fluids; a combustion model for defining the physical quantities and combustion characteristics of combustibles; and a radiation model. For each case, the model that best represents the fire characteristics should be chosen. In grid generation, fire growth or fluid flow can be predicted roughly and grid size set differently according to the importance of the simulation space. Next, obstructions and vents considered in fire fluids are defined, and materials and boundary conditions for each case are set and assembled to generate FDS inputs and carry out a simulation.

FDS output files are comprised of binary data such as Thermocouple, Sprinkler, Heat Release Rate, Gas Mass, Particle, and Plot3D. Among these data, the Plot3D output is an engineering standard format developed by NASA and is mainly used in the plotting of general CFD simulation data [34]; in FDS, it can be converted into text format (i.e. ASCII). Plot3D data produced in FDS contain three-dimensional positional information and static physical quantities corresponding to each grid cell at the given time-step and thus have the best data structure for volume rendering. Table 1 exemplifies Plot3D outputs, where Soot Density is used to visualize smoke propagation while other physical quantities may be used to visualize or evaluate gas mixture flows that can be fatal to trainees via mechanisms such as burning or gas poisoning.

3.4. Pre-processing

In the pre-processing process for applying FDS results to virtual reality systems, data are converted to better suit volume rendering and space partitioning data structures applied to obtain real-time performance. First, FDS grids, as explained in the previous section, are generally comprised of non-uniform grids, and thus they should be collectively converted into text-format ASCII data and resampled at uniform grid size for volume rendering purposes. Also, the FDS analysis coordinate axis and the VR engine coordinate axis may differ from each other and thus be transformed. Octree space partitioning is then applied to the simulation space for each time-step to generate hierarchical data structures. Lastly, Octree data for each hierarchy level are interconnected by time-step, and the applicable memory data are directly saved in the file system in binary stream format. This study defines this as the Octree database, and the entire conversion process is illustrated in Fig. 3.

3.5. Real-time processing

3.5.1. Large memory management

The Octree database generated in the pre-processing process comprises large files over dozens of GB in size, which exceeds the memory range that can be controlled by applications under general operating systems. Therefore, buffering management technology that can minimize input/output loads for the hard disks and enable regular memory inputs/outputs is required. This study uses memory mapped files that map disk files themselves into the memory address space to ensure physical copying is performed only once. Also, memory copying is carried out under a multi-core CPU through an independent process to minimize overall system loads. As described in Fig. 4, ring buffers are provided on the application heap memory, and n-second data blocks from the Octree database are copied into the buffer n seconds ahead every n seconds. This task is synchronized through the main process using events.

3.5.2. LOD selection and interpolation

Data for each buffered Octree level loaded onto the memory undergo consecutive Octree node traversal; LOD selection is carried out on the basis of predefined selection criteria. The view

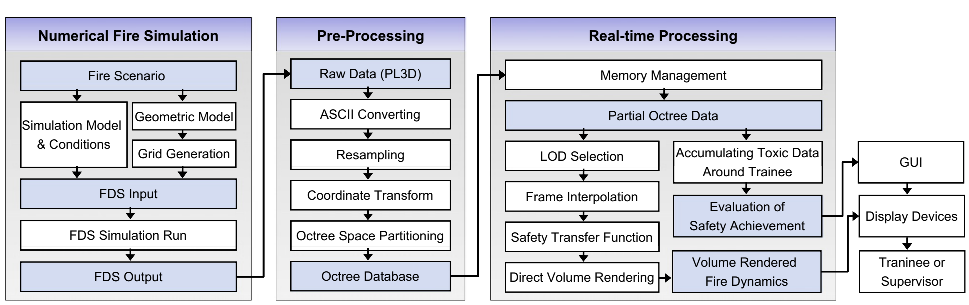

Fig. 2. The data processing logic.

Table 1
Data specifications of Plot3D output file.

Category	Quantity	Description	Units
Grid information	X/Y/Z	Coordinates of the cell corners	m
Gas phase	Soot density	Smoke particulate concentration	mg/m^3
	Temperature	Gas temperature	°C
	HRRPUV	Heat release rate per unit volume	kW/m^3
	Carbon monoxide	CO volume fraction	ppm
	Carbon dioxide	CO_2 volume fraction	ppm
	Oxygen	O_2 volume fraction	mol/mol

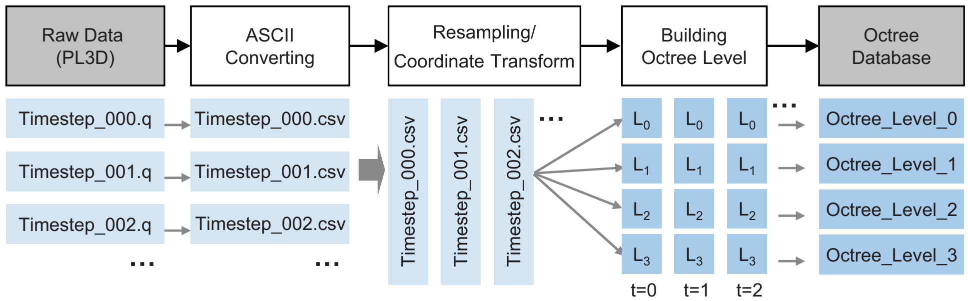

Fig. 3. Pre-processing procedure of FDS output file.

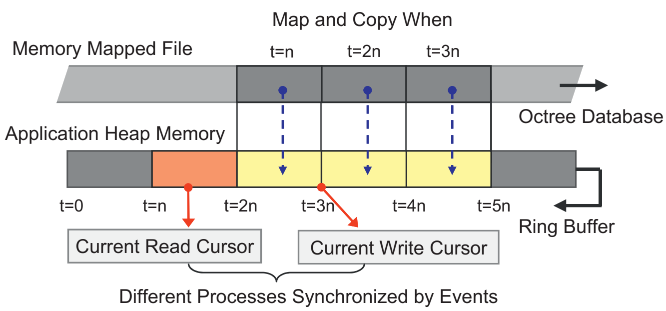

Fig. 4. Memory management for buffered Octree database.

dependent method, which selects the precision level by distance from the user viewpoint, is generally used in LOD selection. CFD data may contain physical quantity spaces whose values are minimal and thus negligible; hence a data dependent method may also be considered, where the minimum data values requiring processing are set and the precision level is selected only when the values are exceeded. Alternatively, a hybrid method combining the two approaches may also be utilized. The selected data for the current and subsequent time-steps are discontinuous data and should thus be interpolated into continuous data needed for the rendering process, which operates at over 30 frames per second. Fig. 5 represents LOD selection and linear frame interpolation; Table 2 exemplifies LOD selection criteria. Parameters implementing each of the LOD selection criteria can enhance hardware adjustability as they are designed to be adjustable on a real-time basis depending on simulator hardware performance and precision level of users' choice.

3.5.3. *Transfer function and volume rendering*

The design of the transfer function plays a very important role in volume rendering, especially when visualizing invisible physical quantities. Smoke propagation can be easily rendered by assigning the normalized soot density value as an opaqueness value. However, the temperature value and volume fraction value of toxic gases are not easily mapped into the visual attribute, because they are invisible. In this work, we propose a safety based transfer function design. For example, the maximum permissible concentration of carbon monoxide in the air is regulated as 50 ppm in South Korea, and anoxia can occur when the oxygen concentration in the air drops to less than 16%. First, we defined the minimum value for each physical quantity that is harmful for human beings. Quantities under this value can be excluded from the rendering process. We then calculated the normalized value from 0 to 1, based on the above minimum value and maximum value during the overall simulation time. Finally, we associated this result value with a color, using the hue parameter of the HSV color model. In addition, opacity was assigned in proportion to the result value. As a result, very noxious gases can be shown as opaque and red-colored volumes, while less harmful gases are shown as transparent and blue-colored volumes.

Optical physical quantities obtained from the transfer function are expressed as graphics images through volume rendering, which is realized via specific technologies such as Ray-Casting, Texture-Slicing, Splatting, and Shear-Warping [33]. For ease of implementation, this study employed Splatting technology, which expresses voxels (volumetric pixels) into small kernel textures and projects/synthesizes and visualizes them on the two-dimensional plane. Gaussian texture that has a normal distribution in the given area is generally used as the kernel texture. Meanwhile, the Splatting method may lead to flickering of volume textures in close proximity from the viewpoint due to the near clipping plane, and consequently we reduced the opaqueness of volume data close to the near clipping plane linearly by distance to visualize the volume data naturally during navigation within the volume.

3.5.4. *Measuring safety achievement*

The buffered Octree data used in the rendering process contain physical quantity data on the whole simulation space at the given time-step. Therefore, we can access the current physical quantities around the trainee, and accumulate them along his or her moving path. This can provide very useful information when it is necessary to evaluate the training process according to safety achievement or scored results. For example, if a trainee evacuates the scene of a fire accident while the value of accumulated carbon monoxide exceeds the critical limit, the evacuation can be regarded as a failure. As another example, if a trainee stays in an area with a temperature of 500 °C for only one second, the result is also a failure. Generated on the basis of fatality and exposure dose concerning hazardousness, safety achievement data are delivered to trainees or training supervisors via GUI and utilized as evaluation information for the entire training process.

4. Implementation

4.1. *Implementation environment*

Table 3 summarizes the H/W and S/W implementation environment utilized in the development of the simulator system proposed in this study. The open-source 3D graphics engine OGRE [35], which is capable of detailed volume rendering control and supports the latest real-time rendering technologies, was selected as the graphics library to enable the integration of data processing of CFD data with the high quality VR graphics.

4.2. *Fire scenario*

For the simulator developed in this study, a vehicular accident occurring in Jukryeong Tunnel, the longest road tunnel in South Korea with a total length of 4.6 km, was selected for simulation.

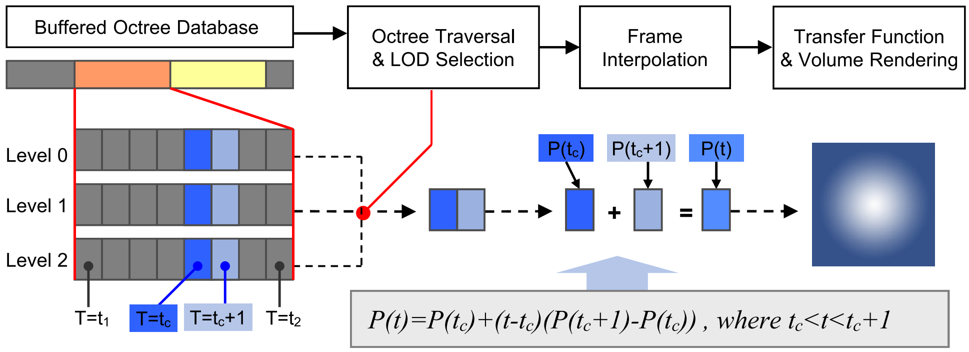

Fig. 5. Real-time procedure of LOD selection and interpolation.

Table 2
Examples of LOD selection criterion and parameters.

LOD Type	Selection criterion	Parameter
None	N/A (No selection needed)	N/A (Fixed Octree level)
View Dependent	Distance between cells and viewpoint	Distance map for each Octree level (1)
Data Dependent	Distribution of physical quantities	Minimum value to be processed (2)
Hybrid Method	Synthesis of view and data method	(1)+(2)

Table 3
Specifications of implementing environment.

	Category	Specification
H/W	CPU	Xeon 2.66 GHz
	RAM	4GB (667 MHz)
	VGA	Quadro FX5600
	HDD	750GB (SATA2)
S/W	OS	WindowsXP
	Fire simulation	FDS 4.0
	FDS input editor	PyroSim 2006
	Graphics API	DirectX 9.0C
	Graphics library	OGRE3D 1.49
	3D Modeling	3DSMAX 2008
	Compiler	VisualC++ 2005

The fire scenario was designed with consideration of ventilation situations caused by longitudinal jet fans and the possibility of escape via a connecting shelter (sub-tunnel). Also, to ensure efficiency of training and system performance, a partial and simplified structural model of the tunnel, as illustrated in Fig. 6, was created by considering the area that an ordinary individual or a firefighter can cover on foot during escape or rescue. The overall training scenario was designed using five categories—i.e. the size of the fire, air flow speed caused by jet fan facilities, training type, control method, and training starting point (Table 4)—that can be combined in various ways to generate diverse training programs.

4.3. FDS simulation and data processing

First, we ran FDS simulations using the above scenario. For the (x: $-500\sim500$, y:$0\sim7.5$, z: $-3\sim3$) range, non-uniform grids were created at $0.5\sim2$ cell-space according to the distance from the fire source, which was placed at (0,0,1) so as to coincide with the hypothetical location of the vehicular accident (unit: meter). Although the range of the actual tunnel model is ($-300\sim300$, $0\sim7.5$, $-3\sim3$), a wider range was set to facilitate later relocation of the fire source along the direction of road traffic. Grids were created by converting the 3D geometric model into DXF format to allow inputting in PyroSim [36], a pre-processor for FDS. The fire size was set at 6 MW, representing an ordinary passenger car on fire, and 18 MW, corresponding to a fully-consumed bus or truck. Flow speed could be set at 0 m/s or 2.5 m/s to simulate longitudinal ventilation using jet fans, resulting in a total of four different fire situations. Each time-step of the simulation was defined as one second, and 1800 pieces of Plot3D data (30 minutes long) containing the physical quantities listed in Table 1 were obtained for each type of fire.

After the FDS simulation, we converted these files into CSV text files using a customized FDS2ASCII utility and then re-sampled the files as a uniform cell-space of 0.75 m. We then transformed cell data from Z to Y for the alignment of different coordinate axes. A five-level hierarchical structure was then created by applying Octree-based space partitioning to each time-step data. Finally, the specific levels of Octree data, which have 1800 time-step pieces, were connected sequentially and stored in the file system as binary streams, which comprise an Octree database. Table 5 shows the detailed specifications for the raw data and the pre-processed Octree database.

4.4. Main features

Fig. 7 is a screen-shot showing the user interfaces for the simulator prototype developed in this study; the various functions of each interface are explained in Table 6. Simple missions assigned to a trainee involve approaching the fire source to identify the accident, finding a fire hydrant, re-approaching the fire source to conduct firefighting activities, and finding nearby victims for rescue and evacuation. Using a joystick button push, the trainee enters the completion status of each mission into the simulator system. Once training is complete, quantitative training results—including completion time by mission, safety achievement, etc.—are outputted to the simulator to enable a simple evaluation of the training result. Fig. 8 shows screen-shots for part of a training program based on a specific scenario.

5. Experiment and results

5.1. Functional evaluation

The key function of the training simulator proposed in this study is to reproduce virtual fire environments and fluid flows

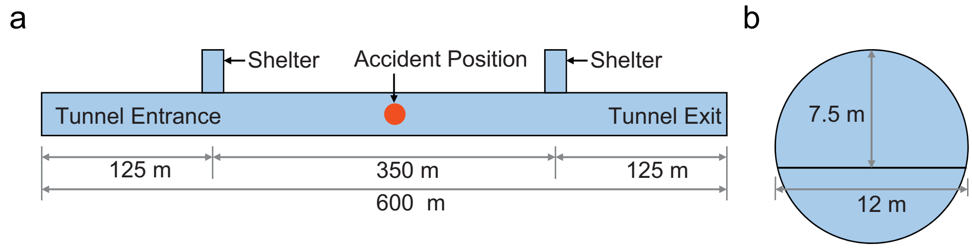

Fig. 6. A simplified structural model of the road-tunnel.

Table 4
Scenario configuration for the fire training.

Category	Scenario
Fire size	6 MW 18 MW
Air flow	No Ventilation 25 m/s
Navigation	Scene overview Fire experience Firefighter training
Avatar control	Predefined path Manual
Starting point	Tunnel entrance/exit First/second shelter

Table 5
Specifications of the raw data and the pre-processed data.

Data type	Cell offset (m)	Cell count	File size (KB)	File count	Binary stream size (MB)
Raw data	Non-uniform	288,000	41,057	1800	72,170
Octree Database					
Level 4	12.0	50	2	1,800	113
Level 3	6.0	400	16	1800	113
Level 2	3.0	3200	125	1800	225
Level 1	1.5	25,600	1000	1800	1800
Level 0	0.75	204,800	8000	1800	14,175
Sum	–	234,050	9143	1800	16,425

based on reliable CFD calculation results and utilize the data in fire training and evaluation. This section evaluates various related functions to verify the applicability of the proposed system design and data processing framework. Table 7 presents the results of a comparative analysis between the implemented simulator system and relevant approaches discussed in Section 2.

5.1.1. CFD data integration

Fig. 9(d) shows the process of volume rendering—using Octree-based space partitioning—on the database generated through data pre-processing; Fig. 9(c) illustrates how gas mixtures are ventilated by the jet pan. The users can either observe such various physical quantities of fire from distance or experience them directly within the volume. With the realistic virtual reality visualization system as the basis, all physical quantities are visualized through volume rendering. This ensures better image quality overall and offers more intuitive interfaces for ordinary users than in the case of scientific visualization tool-based physical quantity visualization for engineering purposes.

5.1.2. Invisible volume visualization

Fig. 9(e), (f), and (g) presents visualization results on oxygen concentration, carbon monoxide concentration, and temperature distribution in the air, respectively. In all cases, color and transparency are mapped on the basis of values that may have impacts on the human body, so the trainers can intuitively identify dangerous and safe areas on three-dimensional settings. As the bird's-eye view helps identify the spread of smoke, toxic gas and heat comprehensively over time, training commanders can come up with optimal responses by scenario and train themselves together with firefighters.

5.1.3. Evaluation of safety achievement

As Fig. 10 shows, we can monitor the current physical quantities around the user by a 2D slice view during the training process. Also, after training is completed, the accumulated physical quantities are calculated and displayed in the result dialog box along with the mission completion time. This illustrates how the CFD data can also be used as a tool for evaluating the fire training process by measuring the safety achievement of the trainee.

5.2. Performance evaluation

The training simulator proposed in this study is a system that is basically run in real-time, so it is essential to evaluate real-time performance using the per-second frame rate of the visualization system. Simulating a situation approximately five minutes after the fire breakout, when the upper portion of the tunnel is almost completely filled with smoke, the number of cells to be rendered, rendering frames per second (FPS), and degree of realism were measured for each LOD method from points of view inside and outside the tunnel, and the average values were calculated. As shown in Table 8 results, real-time performance can be partially ensured depending on LOD selection. The view dependent method is efficient in the inside view, where the direct fire experiences of the general public or firefighters are required and the physical quantities around the users are crucial; the hybrid method combining realism and fps works better for the outside view, which requires control of overall fire propagation by fire-fighting commanders.

5.3. User experience with immersive devices

To effectively achieve the objectives of training, it is necessary not only to realistically render the visual content, but also to adopt an immersive display and interaction devices that increase the degree of perception and interactivity on the visual content [37–39]. In this study, a head mounted display (HMD) that provides eye-separated 3D stereoscopic visualization and a magnetic motion-tracking device that tracks the orientation of the user's head were used to enhance the trainee's immersive experience. In addition, to facilitate the instructor's capacity to monitor and control the training, a large titled screen system implemented by a curved 10-channel multi-vision system was adopted. As shown in Fig. 11, the multi-vision system visualized the supervisory content on the left portion (2 rows $\times$ 2 columns),

Fig. 7. GUI of the simulator system.

Table 6
Key functions of simulator system.

ID	Interface Name	Function
1	Simulation playback	Start/Stop/Pause of fire simulation
2	Data processing status	Monitoring of data processing by time-line
3	3D smoke	Volume rendering of smoke data
4	Trainee avatar	3D avatar moving by user input
5	LOD parameter	Adjusting of LOD type and parameters
6	Mission status	Information of mission list and achievement result
7	Gas property graph	Histogram of physical quantities around the trainee
8	Gas property slice view	Slice view of physical quantities around the trainee

and the training content, identical with the HMD view, on the right portion (2 rows $\times$ 3 columns). Through a number of user tests involving real firefighters, it was possible to conduct a simple VR based fire training course.

6. Discussion

A proper data processing and visualization method is required in order to incorporate CFD results—widely used in engineering analyses—into virtual reality systems; this study has thus proposed an overall system design and an efficient CFD data processing framework for a training simulator. Section 5 has shown that a realistic fire visualization function can be run in combination with precise CFD data. Furthermore, invisible risks such as toxic gases and heat, in addition to smoke, which is directly correlated with psychological fear in fire situations, can also be visualized and used for training purposes. A wide variety of hazards experienced in the training process can also be quantitatively assessed on the basis of the spatial distribution of various physical quantities calculated through CFD. This will help predict the safest evacuation method and route of avoidance in the event of similar fire situations and thereby ensure the safety of firefighting activities.

Although the application of CFD to the fire training process may not be unprecedented, this study holds great significance in which the merits of engineering analysis results are fully exploited in order to further broaden the applications of virtual reality simulators.

However, we should note several limitations and issues that could arise when developing similar simulators, as well as possible future research directions.

As shown in Section 5.2, the real-time performance of our system was achieved with respect to our training scenario. The frame rate is the main indicator for real-time performance of a visualization system; however, it often can vary depending on the number and size of visible objects. These include structures, vehicles, and human subjects in the scene as well as kernel cells for fire volume rendering. This is mostly related to GPU performance, including speed and memory bandwidth, because static objects are stored in the GPU and reused at every frame. Therefore, a

b

c

Fig. 8. Fire training examples using the simulator system. (a) Scenario setup dialog, (b) Mission : Approaching the fire source and (b) Mission : Approaching the fire source.

if we extend our training scenario, both the complexity of the geometric models and the number of kernel cells must be adjusted with consideration of GPU limitation. In the present case, we attempted to adjust the number of kernel cells using LOD selection in order to overcome the GPU bottleneck; however this accompanies CPU calculation when retrieving and selecting the Octree data structure. Meanwhile, because our system generates temporal behavior of each kernel cell using the simulation data stored in the disk, the data transfer rate from disk to GPU, which is affected by the disk, memory, CPU, and system bus, is also an important factor. We observed that data reading operation from the disk is one of the main bottlenecks here, and thus a buffered memory management method was employed. This performance can be improved using recent hardware such as SSD or a 64bit-based large system memory. Against this backdrop, if an extended simulation scenario requires a more geometrically complex model only, the GPU limitation will be an important factor. Furthermore, if it requires larger or more detailed fire phenomena, and thus more kernel cells and more data transfer will be needed, the overall system performance including GPU, CPU, and bus speed should be considered simultaneously.

Another limitation is related to user interactivity. Our study is based on a data-driven method, where the fire simulation is computed off-line and the trainee cannot influence the fire simulation. In other word, users can experience pre-simulated fire phenomena caused by heat transfer or ventilation facilities from different viewing angles in real-time, but cannot interact with fire or smoke directly by means of user inputs such as extinguishing the fire. This is an important limitation for a training application. This issue can be partially resolved by utilizing a hybrid method, which synthesizes global motion calculated off-line using a CFD solver and local details calculated online using a simplified fire dynamics model. Consequently, we could achieve local interactivity while preserving global accuracy of fire phenomena.

As noted in Section 5.3, we conducted a user test and obtained some feedback from several firefighters working at a nearby fire station (Daejeon-Pukbu Fire Station in South Korea). They commented that although our training simulator faithfully visualizes fire phenomena and provides an opportunity to experience lifelike emergency situations including physical quantities, the scenario and training objectives were focused on very fundamental firefighting activities. Because real fire situations can be very diverse, they suggested that more diverse or compositive scenarios are necessary and game playing elements such as solving mazes can also be considered. Nevertheless, they commented that inexperienced firefighters can quickly improve their ability to respond to serious fire situations in a realistic manner through our system, and this should accordingly be the aim of the training activity. A haptic interface using touch or heat was another suggestion, because firefighters search for routes mostly relying on tactile perception via the hand in the dark and smoke-filled space. In addition, a team-based muti-user training environment and communication training between firefighters and a firefighting commander are also suggested.

The detailed implementation technologies suggested in this study, especially space partitioning and volume rendering techniques, are not the latest technologies and have been extensively studied in the field of scientific visualization, as described in Section 3.2; more recently, several studies have been reported where GPU instead of CPU is used to accelerate real-time performance or numerical computation using a parallel processing mechanism. Therefore, further research is needed on the implementation of more efficient and better-performance visual simulation techniques, such as GPU-based space partitioning and volume rendering, or CUDA-based real-time fire dynamics simulation.

In addition, it has thus far been possible only to experience graphically visualized fire behaviors and toxicity exposure as part of training; in the future there will be need for research on VR simulators with multi-sensory interfaces encompassing tactile, auditory, and olfactory sensation, as reinforced by the firefighter's feedback. The general public may feel nervous due to their surroundings and lose their will to evacuate even in the initial stages of fire where no direct damage is done by heat and smoke. Therefore, a multi-sensory-based virtual experience system could play a crucial role in developing and learning a wide variety of preemptive measures and actions needed to further delay the phase of physiological loss of will for evacuation (i.e. incapacitation). Future research should thus include multi-sensory interfaces and related integration techniques that could improve the training effects.

This study has extracted the safety level for a single physical quantity and mapped it into visual properties. However, if the

Table 7
Technical comparison of previous works with our work.

Related works	Objective	Fire simulation and visualization techniques	Data processing techniques	Types of visualized physical quantities	Usage of physical quantities
Julien et al. [22]	Training for experienced firefighters to make decisions and deliver commands on house fires	Volume rendering of FDS data	RLE (run length encoding) compression to reduce data file size	3D smoke and flame	Recognition of fire and smoke propagation
Ren et al. [23]	Training for the general public to experience and evacuate during underground station fires	Particle system with reference of FDS data	Mapping FDS data to a simple physics model (particle system)	3D smoke and flame	Recognition of fire and smoke propagation
SmokeView [24]	Engineering-centric, FDS-only post-processor	Volume and Polygon rendering of FDS data	Simplification by partially transparent parallel planes	Every fire dynamics data simulated from FDS	Understanding and insight of fire dynamics
ADMS [26]	Training for firefighting commanders to set up firefighting plans, learn command systems and communicate	Particle system without physics simulation data	Unknown	3D smoke and flame	N/A
This work	Training for the general public, firefighters and firefighting commanders to experience road tunnel fires and evaluate safety levels	Volume rendering of FDS data	Octree-based space partitioning, multi-resolution LOD selection, memory management	3D smoke and toxic gases (CO, CO_2, O_2, heat and temperature)	Recognition of fire and smoke propagation, experience of harmful gases, safety evaluation

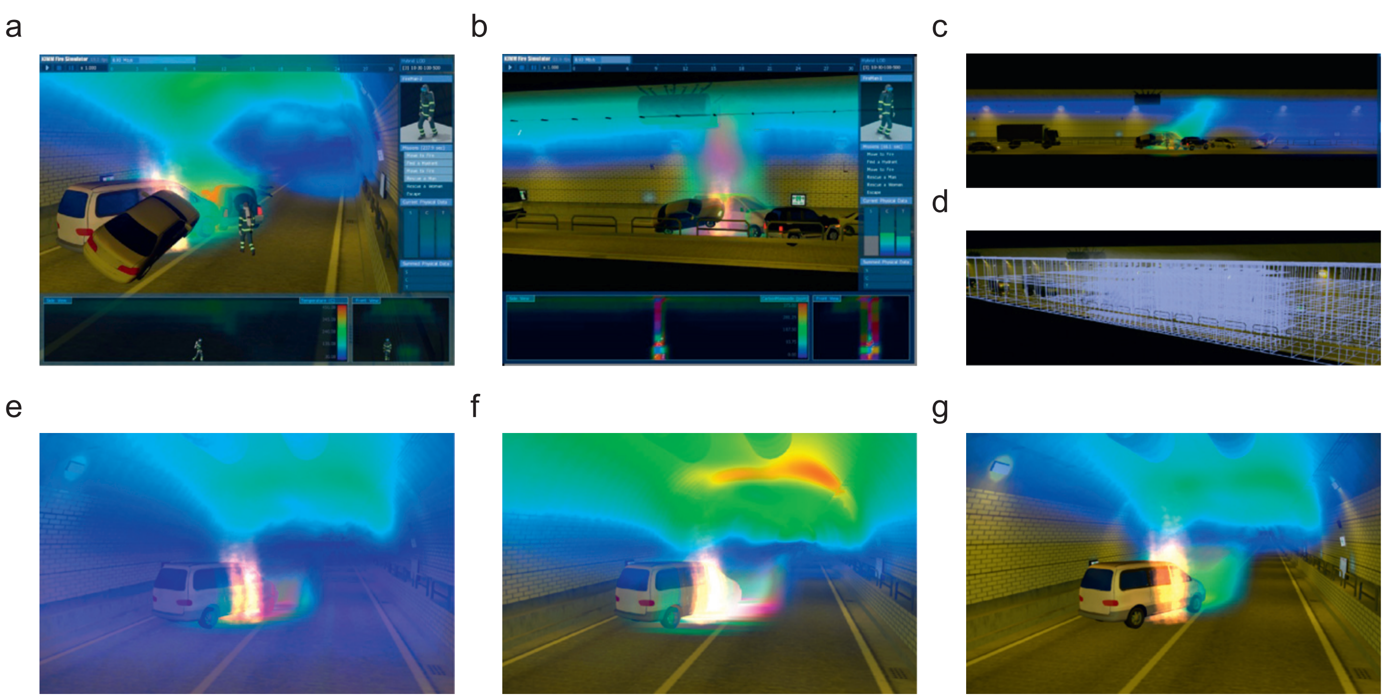

Fig. 9. Functional results of the CFD integrated simulator system. (a) CFD based virtual fire hazard situation, (b) Experience of toxic gases and heat, (c) Fluid flow caused by ventilation system, (d) Octree partitioned volume-rendering space, (e) Visualization of oxygen concentration, (f) Visualization of carbon monoxide concentration and (g) Visualization of temperature distribution.

overall level of lethality that multiple hazards have on the human body are calculated and visualized, this can be developed into a function that will help experience and assess the overall safety level in accident situations on a three-dimensional basis. High expectations are also held for the development of engineering-based training simulators to perform reliable preemptive training in wide-ranging industries, such as power plants and chemical plants, where human errors may translate into major disasters.

7. Conclusion

A virtual reality-based fire training simulator provides wide-ranging second-hand experience to the general public or inexperienced firefighters or commanders so that they can make prompt decisions and safe and organized responses in actual fire situations and thereby enhance human safety.

To undertake such computer-based fire training more effectively, this study has developed a real-time VR simulator system based on a highly reliable CFD simulation. Offline data conversion and real-time data processing logics have been proposed to apply massive CFD data, where various physical quantities of fire are calculated, to the real-time system. The performance and functions of the system have been verified through simulator implementation and experimentation. The fire CFD data, in particular, contain data on toxic gases and heat, which have direct impacts on fire responses and rescue activities, and thus realistic visualization and experience of these data have helped further improve the effects of the fire training and provide useful knowledge to the inexperienced users.

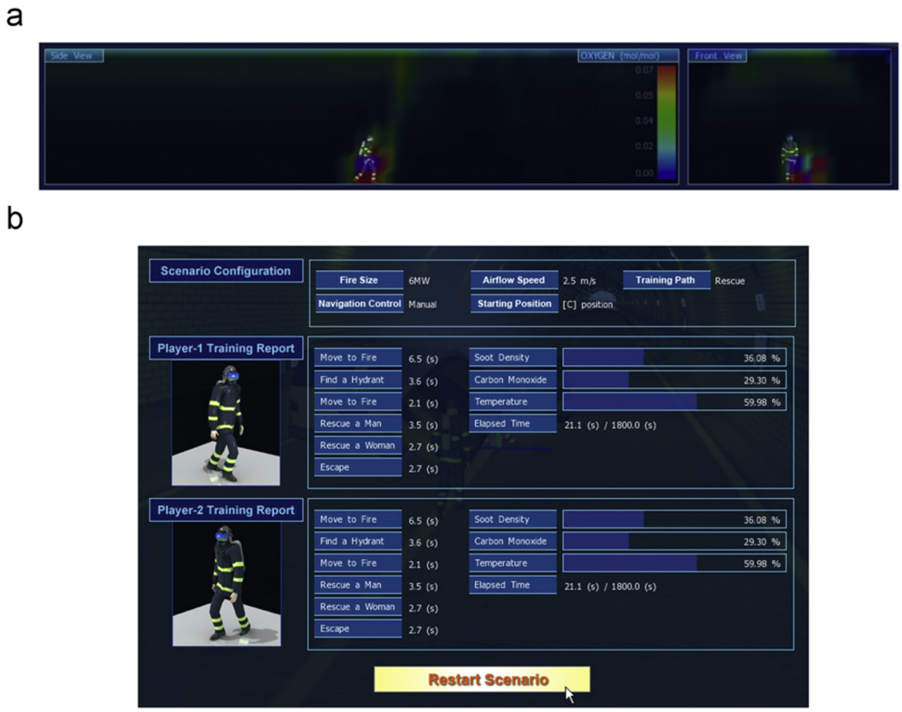

Fig. 10. Evaluation of safety achievement. (a) 2D slice view of physical quantities around the trainee and (b) The training report dialog containing total exposure to danger area.

Table 8
Comparison of real-time performance and realism by LOD method.

LOD method	Inside view			Outside view		
	No. of cells	FPS	Realism	No. of cells	FPS	Realism
None	18,680	13.9	High	52,789	22.5	High
View dependent	705	26.4	High	30	66.0	Low
Data dependent	11,418	18.2	High	37,905	8.3	High
Hybrid method	11,418	18.1	High	897	64.7	Medium
Selected method	View dependent			Hybrid method		

Fig. 11. Fire training experience using immersive devices.

Acknowledgment

This research was supported by WCU (World Class University) program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (R31-2008-000-10045-0). This research was also supported by the Korean Research Council for Industrial Science & Technology (Project Code:SC0830) and Business for Cooperative R&D between Industry, Academy, and

Research Institute funded Korea Small and Medium Business Administration in 2010 (Grants No.00041642-1).

Appendix A. Supplementary materials

Supplementary materials associated with this article can be found in the online version at doi:10.1016/j.firesaf.2012.01.004.

References

[1] W.T. Reeves, Particle systems-a technique for modeling a class of fuzzy objects, ACM Trans. Gr. 2 (2) (1983) 91–108.
[2] T. Ilmonen, J. Kontkanen, The second order particle system, J. WSCG 11 (2) (2003) 240–247.
[3] S. Zhou, Y. Sun, L. Lu, et al., Fire simulation model based on particle system and its application in virtual reality, Proc. ICAT2006 Conf. (2006) 642–645.
[4] E. Wu, Y. Liu, X. Liu, An improved study of real-time fluid simulation on GPU, Comput. Anim. Virtual Worlds 15 (3-4) (2004) 139–146.
[5] J. Krüger, R. Westermann, GPU simulation and rendering of volumetric effects for computer games and virtual environments, Comput. Gr. Forum 24 (3) (2005) 685–693.
[6] K. Crane, I. Llamas, S.Tariq, Real-time simulation and rendering of 3D fluids, GPU Gems 3, Addison Wesley, 2007, pp.633–675.
[7] N.L. Ryder, J.A. Sutula, C.F. Schemel, A.J. Hamer, Consequence modeling using the fire dynamics simulator, J. Hazardous Mater. 115 (1-3) (2004) 149–154.
[8] K.B. McGrattan, S. Hostikka, J. Floyd, et al., Fire Dynamics Simulator (version 5) Technical Reference Guide, NIST Special Publication, NIST, 2008 1018-5.
[9] G.P. Forney, D. Madrzykowski, K.B. McGrattan, et al., Understanding fire and smoke flow through modeling and visualization, IEEE Compu. Gr. Appl. 23 (4) (2003) 6–13.
[10] G. Hadjisophocleous, Y.J. Ko, Using a CFD simulation in designing a smoke management system in a building, Proc. Winter Simulation Conf. (2006) 2071–2076.
[11] L.H. Hu, N.K. Fong, L.Z. Yang, W.K. Chow, et al., Modeling fire-induced smoke spread and carbon monoxide transportation in a long channel: fire dynamics simulator comparisons with measured data, J. Hazardous Mater. 140 (1-2) (2007) 293–298.
[12] W. Schroeder, K. Martin, B. Lorensen, The Visualization Toolkit, Kitware Inc, 2004.
[13] A. Squillacote, The paraview guide, Kitware Inc., 2008.
[14] D. Thompson, J. Braun, R. Ford, OpenDX—Paths to visualization, Visualization and Imagery Solutions Inc., 2001.
[15] CEI (Computational Engineering International) Inc. URL: 〈http://www.ensight.com〉.
[16] D.L. Tate, L. Sibert, T. King, Virtual environments for shipboard firefighting training, in: Proceedings of IEEE 1997 Annual International Symposium on Virtual Reality, IEEE Comput. Soc. Press, 1997, pp.61–68.
[17] P. Backlund, H. Engstrom, C. Hammar, M. Johannesson, M. Lebram, Sidh-a game based firefighter training simulation, Information Visualization, 2007, 11th International Conference, IEEE, 2007, pp.899–907.
[18] L. Chittaro, R. Ranon, Serious games for training occupants of a building in personal fire safety skills, in: Proceedings of VS-GAMES09: IEEE First International Conference on Games and Virtual Worlds for Serious Applications, 2009, pp. 76–83.
[19] S. Smith, E. Ericson, Using immersive game-based virtual reality to teach fire-safety skills to children, Virtual Reality, Springer London, 13(2), 2009, pp. 87–99.
[20] R. Bukowski, C. Séquin, Interactive simulation of fire in virtual building environments, in: Proceedings of the ACM SIGGRAPH97, 1997, pp. 28–35.
[21] R.D. Peacock, W.W. Jones, G.P. Forney, et al., CFAST–Consolidated model of fire growth and smoke transport (version 6) user's guide, NIST Special Publication 1041, NIST, 2005.
[22] T.U. St. Julien, C.D. Shaw, Firefighter command training virtual environment, Proc. Diversity Comput. Conf. (2003) 30–33.
[23] A. Ren, C. Chen, Y. Luo, Simulation of emergency evacuation in virtual reality, Tsinghua Sci. Technol. 13 (5) (2008) 674–680.
[24] G.P. Forney, Visualization—A tool for Understanding Fire Dynamics, NISTIR7431, NIST, 2007.
[25] G.P. Forney, User's Guide for Smokeview Version 5—A Tool for Visualizing Fire Dynamics Simulation Data, NIST Special Publication, NIST, 2007 1017-1.
[26] ADMS—The advanced disaster management simulator, etc USA, URL: 〈http://trainingfordisastermanagement.com/〉.
[27] E. LaMar, B. Hamann, K.I. Joy, Multiresolution techniques for interactive texture-based volume visualization, Proc. IEEE Visualization Conf. (1999) 355–361.
[28] S. Piccand, R. Noumeir, E. Paquette, Region of interest and multiresolution for volume rendering, IEEE Trans. Inform. Technol. Biomed. 12 (5) (2008) 561–568.
[29] S. Kühner, E. Rank, M. Krafczyk, Efficient reduction of 3D simulation results based on spacetree data structures for data analysis in Virtual Reality environments, Proc. AVR II CONVR2001 Conf. (2001) 1–8.
[30] I. Boada, I. Navazo, R. Scopigno, Multiresolution volume visualization with a texture-based octree, The Visual Comput. 17 (3) (2001) 185–197.
[31] A. Knoll, A survey of octree volume rendering methods, in: Proceedings of 1st IRTG Workshop, 2006.
[32] S. Guthe, M. Wand, J. Gonser, W. Straβer, Interactive rendering of large volume data sets, Proc. IEEE Visualization Conf. (2002) 53–60.
[33] K. Engel, M. Hadwiger, J. Kniss, C. Rezk-Salama, D. Weiskopf, Real-Time Volume Graphics, A K Peters, 2006.
[34] P.P. Walatka, P.G. Buning, L. Pierce, P.A. Elson, PLOT3D User's Manual, NASA Technical Memorandum 101067, NASA, 1990.
[35] OGRE (Object-Oriented Graphics Rendering Engine) project, URL: 〈http://www.ogre3d.org〉.
[36] PyroSim—A graphical user interface for the fire dynamics simulator (FDS), URL: 〈http://www.thunderheadeng.com/pyrosim/〉.
[37] B. Yoo, M. Cha, S. Han, A framework for a multi-sensory VR effect system with motional display, International Conference on Cyberworlds, 2005, pp.237–244.
[38] Y. Kim, J. Yang, S. Han, A multichannel visualization module for virtual manufacturing, Comput. Ind. 57 (7) (2006) 653–662.
[39] P. Hur, J. Yang, S. Han, B. Yoo, An underwater vehicle simulator with immersive interface using X3D and HLA, Simulation: Trans. Soc. Modeling Simulation 85 (1) (2009) 33–44.