Received December 9, 2018, accepted January 6, 2019, date of publication January 21, 2019, date of current version March 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2893919

A Cloud-Based Real Time Polluted Gas Spread Simulation Approach on Virtual Reality Networking

ZHENGWEI HE[1,2,3], LAN YOU[4], RYAN WEN LIU[1], FAN YANG[1], JIE MA[1], AND NAIXUE XIONG[5,6]

[1]School of Navigation, Wuhan University of Technology, Wuhan 430063, China
[2]Key Laboratory of Hubei Inland Shipping Technology, Yantai 430063, China
[3]National Engineering Research Center for Water Transport Safety, Wuhan 430063, China
[4]Faculty of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China
[5]College of Intelligence and Computing, Tianjin University, Tianjin 300350, China
[6]Department of Mathematics and Computer Science, Northeastern State University, Tahlequah, OK 74464, USA

Corresponding authors: Lan You (yoyo@hubu.edu.cn) and Naixue Xiong (xiongnaixue@gmail.com)

This work was supported in part by the National Science Foundation of China under Grant 51679182, in part by the Fundamental Research Funds for the Central Universities of China under Grant WUT: 2018ćó064GX, and in part by the Fund of the Hubei Key Laboratory of Inland Shipping Technology of China under Grant NHHY2017001.

ABSTRACT It is a difficult problem to realize the online simulation of fluid phenomena in a wide range of networking-based virtual reality scenes. Real-time solution and three-dimensional (3D) rendering of fluid dynamic process pose a great challenge to the online computing capability of networking. Aiming to model the spread characteristics of polluted gas, this paper constructs a polluted gas simulation model using computational fluid dynamics in a cloud computing environment. This model is flexible and accepts environmental parameters, such as the wind field and ground features in a cloud-based virtual world. Constrained boundary models are proposed for simulating the interaction between polluted gases and a virtual reality scene. An iterative semi-Lagrangian method with over-relaxation is adopted for a fast solution to the model with both simplicity and fidelity. The reported experiments show that our solution can realistically simulate the movement patterns and real-time temporal-spatial dynamics of polluted gas in a large-scale 3D online virtual world while responding to environmental changes interactively. Two typical fluid simulation pipelines are proposed to reasonable use of cloud computing resources for solution and rendering. The experiments demonstrated that the solution performance of our method is better than the same other methods, and this simulation approach achieves real-time simulation of polluted gas spread in a large-scale scene with online ordinary PC, laptop, or mobile device. This paper proposes a solution useful for cloud-based virtual reality and augmented reality networking, and some other real-time applications dealing with fluid phenomena. This paper may provide basic method and tool for urban computing and policy-making of the smart city.

INDEX TERMS Fluid simulation, cloud computing, online virtual world, temporal-spatial dynamics, VR/AR networking, smart city.

I. INTRODUCTION

Virtual Reality (VR) and Augmented Reality (AR) devices and platforms, such as Google Glass, Google Tango platform, and Microsoft HoloLens, have recently drawn tremendous attention from both the industry and the research communities. VR/AR are expected to be the next generation Killer-Apps in the future Internet, and will be projected to form a tremendous market. And some potential VR/AR applications in smart city have emerged, such as, digital heritage preservation, touristic guiding, mechanical engineering, information systems. However, many technical challenges need to be overcome to facilitate the embedded adoption of VR/AR. Real-time fluid animation generation is just one of those challenges. Realistic fluid simulations can deliver immersive VR/AR experience in future networks.

The associate editor coordinating the review of this manuscript and approving it for publication was Miltiadis Lytras.

2169-3536 © 2019 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Polluted gas is an important element of virtual scene in VR/AR. However, it is very difficult to real-time simulate and render fluid dynamics under network environment. This is due to the limited computing resources or rendering capabilities of network users' terminal device (e.g., personal computer, mobile, and VR glass).

The rapid simulation of fluid phenomena, such as polluted gas spread, has long been a long-standing challenging research problem [1]–[3]. Many real-time applications, such as forecasting, quick evaluation, emergency decision making and online Virtual Earth systems demand rapid simulation of fluid phenomenon [4], [5]. The common way to simulate the evolution process of fluid phenomena is by creating numerical models. Detailed polluted gas data (smoke, fog, etc.) is difficult to obtain however, and usually the measured data is in time slice form with very low spatial resolution. Polluted gases spread across a region in three dimensions. In order to simulate polluted gas changes realistically, the local characteristics and details must be described. Here, high resolution spatial-temporal data is indispensable but the volume of data is huge. It places great demands on both the hardware and software environment's computing and storage capacity [6]. Polluted gas motion is very complex, affected by wind, temperature, humidity, …etc. in the atmosphere. Those factors create great difficulties for fast modeling and rendering.

The purpose of this study is to simulate the spread process of polluted gas in a cloud-based virtual world, and try to provide a solution for fluid simulations in VR/AR networking. This requires the simulation model to describe the detailed characteristics of gas movement, to interact with the virtual scene, to be simple enough to quickly solve the simulation model problem and to generate real-time 3D fluid field data. Our study uses a simulation model based on Computational Fluid Dynamics (CFD), which can simulate more complex fluid dynamic processes, to describe the polluted gas spread process in network based virtual world. An iterative semi-Lagrangian method with over-relaxation is adopted for a fast solution to the model with both simplicity and fidelity. The cloud computing resources are used for the solution and rendering of polluted gas spread process.

II. RELATED WORK

In environmental research, spread models are usually used to describe the transport, diffusion and dilution of pollutants. Because of their complexity, however spread models are difficult for real-time polluted gas simulation in virtual reality scenes [7]–[9]. Techniques to simulate polluted gas spread are mainly image-based methods [10]–[12] and particle system methods [13]. The image-based simulation method, using two-dimensional (2D) image sequences to render the evolution process of pollution gas, does not express the detailed characteristics or internal properties of the phenomena. Image-based simulation suffers from information loss, and lacks a dynamic response mechanism within the virtual environment. This method renders polluted gas as a form of 2D performance, and therefore does not fit 3D virtual scenes. The method based on particle systems has better performance when representing fluid processes, but also needs to build a simulation model to drive particle system movement.

In the computer graphics research field, fluid phenomenon simulation is usually based on Computational Fluid Dynamics (CFD). CFD is widely used for smoke and pollution gas transmission simulation [14], and can provide detail and accurate data about transmission velocity, temperature, and density distribution of polluted gases. But its calculation is time consuming, has brought great challenges for personal computer-based and real-time applications. Stam proposed a fast solver for the Navier Stokes (N-S) equations, using the semi-Lagrangian scheme for solving the advection term, and calculating the viscosity diffusion term using an implicit iterative method. This method makes a very important contribution for real-time solution and the application of fluid dynamics equations. On this foundation, Fedkiw simulated the movement of smoke (Fedkiw et al. 2001), Harris simulated the movement of clouds, and Liu calculated the movement of fluid around an obstacle. In recent years, some other methods for quick fluid simulation started to appear, for example, particle-based fluid dynamic methods, Lattice Boltzman the model reduction method, and so on. However, most of the above methods are experiments in a simple small-scale scene, and don't consider the effects that complex environments bring to the spread process. In a three-dimensional network based virtual scene, the spread of contamination gas should be online simulated in real time, and can be viewable from any position and observer angle. The environment's restriction effects (such as terrain and buildings) must be taken into account. Current methods cannot meet these requirements.

This paper will propose to develop a polluted gas simulation model based on the N-S equation that can reflect the interaction between the gas and the virtual scene. Terrain and ground features in this case are regarded as boundary conditions restricting the polluted gas spread process. This study will balance the contradiction between simulation fidelity and quick solution, and simplify the simulation model to seek a quick and stable solution algorithm. To better implement the fluid simulation in VR/AR networking, two fluid dynamics solution pipelines are proposed, and some use cases are discussed. Finally, these simulation and visualization experiments of the polluted gas spread processes will be executed in a wide range of 3D virtual scenes under network environment.

III. CONSTRUCTION OF THE POLLUTED GAS SIMULATION MODEL

A large number of small solid particles, droplets or toxic gas molecules (such as smoke, fog, SO2, CO2) move in the air to form polluted gas; we will call these gas molecules as particles. Our modeling object is particles. After breaking away from the pollutant source, each particle is affected by gravity, wind, buoyancy, …etc.. These factors and the movement of particles, are used as parameters to build the polluted gas spread model.

A. SPREAD MODELING OF POLLUTED GAS

The key to fluid evolution simulation is to determine the current velocity field properly. A physical model, based on the Navier-Stokes equation, is used for fluid description. If the speed and pressure are known at the initial time when $t = 0$, the state changes of the fluid are shown as follows:

$$\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{t}} = -(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} - \frac{1}{\rho}\nabla \boldsymbol{p} + \boldsymbol{\mu}\nabla^2\boldsymbol{u} + \boldsymbol{F} \tag{1}$$

$$\nabla\cdot u = 0 \tag{2}$$

$$\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{t}} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} = \boldsymbol{F} - \frac{1}{\rho}\nabla \boldsymbol{p} \tag{3}$$

In the Cartesian coordinate system, Equation (3) actually includes three equations:

$$\frac{\partial u_x}{\partial t} + u_x\frac{\partial u_x}{\partial x} + u_y\frac{\partial u_x}{\partial y} + u_z\frac{\partial u_z}{\partial z} = f_x - \frac{1}{\rho}\frac{\partial p}{\partial x} \tag{4}$$

$$\frac{\partial u_y}{\partial t} + u_x\frac{\partial u_y}{\partial x} + u_y\frac{\partial u_y}{\partial y} + u_z\frac{\partial u_y}{\partial z} = f_y - \frac{1}{\rho}\frac{\partial p}{\partial y} \tag{5}$$

$$\frac{\partial u_z}{\partial t} + u_x\frac{\partial u_z}{\partial x} + u_y\frac{\partial u_z}{\partial y} + u_z\frac{\partial u_z}{\partial z} = f_z - \frac{1}{\rho}\frac{\partial p}{\partial z} \tag{6}$$

For an incompressible homogeneous fluid, the density is constant, as shown in Equation (2), and its form in the Cartesian coordinate system is:

$$\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} + \frac{\partial u_z}{\partial z} = 0 \tag{7}$$

The forces affecting polluted gases mainly include: gravity F_g, wind force F_w, buoyancy F_b. Thus:

$$F = F_g + F_b + F_w \tag{8}$$

Particles of polluted gases are affected by a force F_w from the surrounding atmospheric wind field. The effect that force F_w imposes on the particles of polluted gases, will be reflected by fluid velocity u in the equation (3). The atmospheric wind field can be obtained directly from the wind field data, or calculated from the wind field model.

Assume the density of particles of polluted gas is ρ; the given density of the atmosphere around polluted gases is ρ_0. Particles of polluted gases are affected by the joint action of gravity F_g and buoyancy F_b, which together produce a vertical lift force F_v, then F can be expressed as:

$$F = F_v = F_g + F_b = \frac{\rho_0 - \rho}{\rho}g = \left(\frac{\rho_0}{\rho} - 1\right)\cdot g \tag{9}$$

Finally, there's a need to calculate ρ the density of polluted gases transmitted by the fluid velocity ρ can be described by the advection term in Equation (1):

$$\frac{\partial \rho}{\partial t} = -(u\cdot\nabla)\rho + s_0 \tag{10}$$

where s_0 is a density source term for pollutants, for example by fire, a chemical reaction, car exhaust, etc..

Then equations (2), (3), (9) and (10) constitute the spread model of polluted gas.

B. BOUNDARY CONDITIONS

We assume that the polluted gas emission source is an isolated point source. Suppose that the initial time t=0, and the emission velocity of polluted gases is the initial velocity that the particles of polluted gas leave from pollutant source, and the pressure is 0 everywhere;

(1) When the polluted gas spreads around in atmosphere, the polluted gas particles can freely move in the flow field space. The boundary between the polluted gas and atmosphere is taken as a free boundary. In the other place, such as ground, building, etc., the polluted gas particles' movement would obviously be restricted by these solid boundaries. These boundaries are taken as fixed boundaries.

(2) In urban areas, the ground surfaces are composed of different materials, such as concrete, lawn, asphalt. If polluted gases flow into ground which is concrete, i.e., asphalt or stone area, the normal velocity of fluid will be taken as zero,

$$u_n = 0$$

If the polluted gas flow into lawn, shrubs or water field on the ground, along these ground boundary, the velocity of fluid will vanish, i.e.

$$u_x = u_y = u_z = 0$$

(3) Buildings on the urban ground have an impact on the flow field. Along these boundaries, the normal velocity of fluid is zero, i.e.

$$u_n = 0$$

(4) Relative to the effects of the prevailing wind force direction, the buoyancy effects of polluted gases in the atmosphere, can be neglected;

(5) On the boundaries, according to Neumann boundary condition, the change rate of pressure in the normal direction is 0, expressed as.

$$\partial p/_{\partial n} = 0$$

C. CONSTRAINED BOUNDARIES FOR POLLUTED GASES INTERACT WITH VIRTUAL SCENE

Polluted gas spread is influenced by the surrounding environment. On the ground, it is restricted by the terrain and ground features; it will diffuse freely where there is no obstruction. The preceding simulation equations considered the impact caused by wind, gravity, etc.. Next, we adopt constraint conditions to reflect the interactive effect between polluted gas and terrain or ground features. The specific approach is to take the outline of terrain and ground features as the gas's lower or side boundary to restrict its spread.

1) THE TERRAIN CONSTRAINT BOUNDARY

A terrain model can be used to describe the terrain boundary limiting polluted gas spread. Usually a digital elevation model (DEM) is used to represent the topographic surface. In this case the Terrain Boundary (TB) can be described as the original surface fitted by a series of surfaces or planes, which connect with a limited number of sampling points. It is defined as the collection of surfaces S_j, connecting sampling points or interpolation points V_i in region D according to some kind of rule, for example, piecewise linear interpolations over a triangular mesh.

$$TB = \left\{S_j = \gamma\left(V_i\right) \mid V_i\left(x_i, y_i, H_i\right) \in D, i \in N; j \in N\right\} \tag{11}$$

where, H_i is the elevation value at the two-dimensional coordinates (x_i, y_i) in a two-dimensional space; N is the set of positive integers. When terrain is fitted with a triangulation net, γ is the rule for triangular-subdivision, and V_{ai}, $V_{bi}V_{ci}$ are discrete spatial points. Then,

$$TB = \{S_i = \gamma\left(V_{ai}, V_{bi}, V_{ci}\right) \mid V_{ai}, V_{bi}, V_{ci} \in D\} \tag{12}$$

2) CONSTRAINT BOUNDARIES FOR GROUND FEATURES

Where buildings and other ground features impose restrictions on the polluted gas, the geometry convex hull [15]–[16] of each object can be regarded as the constraint boundary. Each geometry convex hull is the minimum volume convex polyhedron that contain the building or feature object. The convex hull boundary (CHB) is the set of geometry convex hulls of all the ground features.

Let V be the set of 3D spatial points constituting the shape of a building or ground feature. Then the constraint boundaries around individual ground feature are:

$$CHB = \{S_i, S_i = \gamma\left(V_a, V_b, V_c\right) \mid V_a, V_b, V_c \in D\} \tag{13}$$

3) THE SYNTHESIS MODEL FOR THE CONSTRAINED BOUNDARY

Restrictions are mainly caused by terrain and features. Considering the effects caused by terrain and features together, here the Integrated Boundary (IB) is used to constrain the spread of polluted gas:

$$IB = TB \oplus CHB \tag{14}$$

where, $\oplus$ is the symmetric difference operation.

When computing the flow field, the boundary condition (2) is used to process the fluid velocity field at the $TB - (TB \cap CHB)$ boundary; and the boundary condition (3) is applied along $CHB - (TB \cap CHB)$ boundary.

D. MODEL SOLUTION

In practical applications, the flow field needs to be qualified and expressed discretely. The usual Eulerian method partitions the bounded flow field space into voxels. Supposing the velocity and density is constant in each grid cell (voxel), we usually take the voxel center value as the status or property value of whole voxel. All the values of the grid network change over time. By solving the simulation model, we obtain the spatial-temporal data for the flow field. To simplify boundary constraints, we establish additional grid layers to surround the flow field space. Along the grid boundary layers, we realize the boundary constraints according to those boundary conditions.

In most cases, an explicit method is an unstable way to calculate the advection term for a fluid simulation model. The calculation is stable only in cases where the time step is short. A short time step slows down the calculation speed. In order to solve the stability problem caused by the explicit method, this paper uses an unconditionally stable model, adopting the semi-Lagrangian method to solve the advection item. The core idea of this method is to use reverse processing and an implicit solution method. Each three-dimensional grid cell is treated as a particle to obtain its previous location by back-tracking the particle's movement in the velocity field, then calculating and copying this measurement q (it may be any measures carried by the fluid such as speed, density, temperature, etc.) at that location as the measurement at current measurement, namely:

$$q\left(s, t + \Delta t\right) = q\left(s - u\left(s, t\right)\Delta t, t\right)$$

The semi-Lagrangian method combines the Euler method's regularity and the Lagrange method's stability, to ensure the stability of all time steps with a simple effective solution for the advection term. The Fast Fourier Transform (FFT) method can be employed to handle both periodic and fixed boundary conditions [17]. This paper uses the semi-Lagrangian method to solve equations (3) and (10), and adopts Stam's method to solve for the pressure force term. Meanwhile, a trilinear interpolation method is carried out using the surrounding grid centers. For the differential equation solutions, this study uses the over-relaxation iterative method [18] to calculate varying values over time. During each iteration, the grids at the fixed boundary are calculated with the constrained boundary model, in accordance with the corresponding boundary conditions.

IV. THE CLOUD-BASED FLUID SIMULATION PIPELINES FOR MODEL SOLUTION OF POLLUTED GAS SPREAD

In networks, it is still a challenge to allocate the limited resources among users to meet their specific quality of service requirements [19]–[21]. Fluid simulation and visualization demand high computational performance and graphics rendering capability on the hardware environment. The online fluid animation in 3D virtual scene needs to set up the reasonable pipeline to rational use of cloud-based computing resources for solution and rendering. We propose two typical fluid simulation pipelines for cloud-based VR/AR systems in Figure 1.

This paper designed two ways of solving the fluid simulation model, shown as Pipeline 1 and Pipeline 2 in Figure 1. In Pipeline 1, the input parameters, such as terrain data, environment factors, and wind field data, are taken as initial conditions of fluid simulation model. The control instruction is the programming factors for control the process of fluid simulation Both initial conditions and control instruction are deployed on cloud server and accessed by the client application through cloud services. The solution of fluid simulation

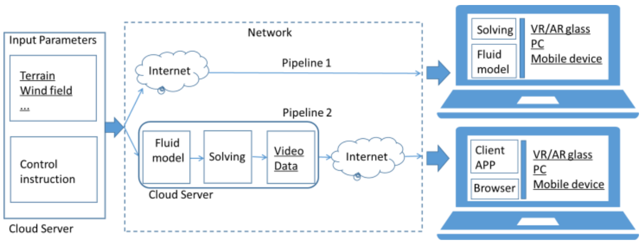


FIGURE 1. Two cloud-based pipelines for fluid simulation.

model is deployed to the terminal VR/AR device, such as personal computer, VR/AR glass, mobile phone.

In Pipeline 2, Both initial conditions and control instruction are accessed by the third cloud server, where the fluid simulation model is solved, 3D fluid animation are generated and transformed into video or compressed data stream to service client application. The client application could be client APP or web browser in terminal VR/AR device of network users, such as personal computer, VR/AR glass, mobile phone.

The pipelines can provide different cloud-based implementation patterns for the simulation of polluted gas spread.

V. EXPERIMENTAL RESULTS AND DISCUSSION

Our experiments are developed on a cloud-based virtual earth platform called GeoGlobe (Wu and He 2010). The virtual city of GeoGlobe is constructed from Digital Elevation Model and Digital Building Model. The 3D urban model is based on simple primitives and a photorealistic model where the pictures are obtained by close frontal pictures, and the dimensions of the buildings by using aerial photogrammetry or Airborne Laser Scanning or LIDAR. These models are quite accurate, and their resolution is adequate. Due to the limited computing resources or rendering capabilities of users' terminal device usually are limited, we adopt 3 representative personal terminal devices as the hardware experimental environment to test the polluted gas spread simulation approach.

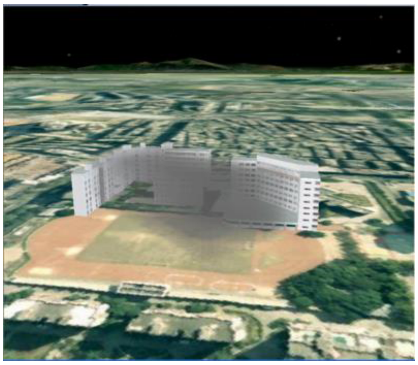
User terminal device 1
Device type: Personal computer
CPU: Intel Pentium IV, 2.66G, core 2
RAM: 1G
Graphics engine: Microsoft DirectX9.0
System software: Microsoft Windows XP Professional, Service Pack2
User terminal device 2
Device type: Laptop, LENOVO T540 p
CPU: Intel Core i5-4300M, 2.6G, core 2
RAM: 8G
GPU: HD Graphics 4600
System software: Microsoft Windows 7 Professional, 64-bit
User terminal device 3
Device type: Mobile phone, HUAWEI MHA-AL00
CPU: ARM ARMv8, 1.84G, core 8
RAM: 6G
System software: Android 8.0.0

Experiment 1 (The Solution Experiments for Solving a Spread Simulation Model of Polluted Gas at Different Grid Scales): The solution experiments were made on a personal computer as the User terminal device 1, without adopting any additional hardware accelerated methods. The computing capabilities of User terminal device 1 are roughly equivalent to or less than the experimental environment of Shin (2007) and Liu (2004). The experimental results are shown in Table 1, and the visualization effects are shown in Figure 2.

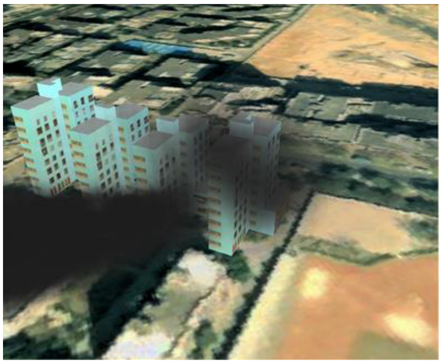
TABLE 1. Solution time at different grid scales.

Method	Device configuration	Grid scale	Solution time (s/step)	Frequency of solving (t/s)
Method of this paper	Pentium IV core 2 2.66, 2.67GHz; RAM 1GB; Without GPU	16×16×16	0.007	142
		32×32×32 (32,768)	0.048	20.7
		64×32×32 (65,536)	0.093	10.8
		64×64×64 (262,114)	0.461	2.17
(Shin 2007)	Pentium IV 3.0GHz; RAM 1GB	512×512 (262,114)	1~3	0.33~1
		128×128×128 (2097,152)	5	0.2
(Liu 2004) GPU based	Pentium 2.8GHz ; RAM 2GB; GeForceFX5 950 256M	64×34×16 (34,816)	0.0467	21.4
		64×17×16 (17,408)	0.0234	2.8

These experiments show that, on the same grid size, the solution time in this study ($64\times64\times64 = 262114$) is 0.461s, while Shin's method ($512\times512 = 262114$) is 1$\sim$3s (Shin 2007). Considering that the experimental computer has a dual CPU, this study's solution time can be converted into about 0.922s, and is faster than Shin's method. Liu's method (Liu 2004) is based on the parallel GPU acceleration method, this study's method is CPU-based only, on the same grid scale, and both use roughly the same computation time.



(a)



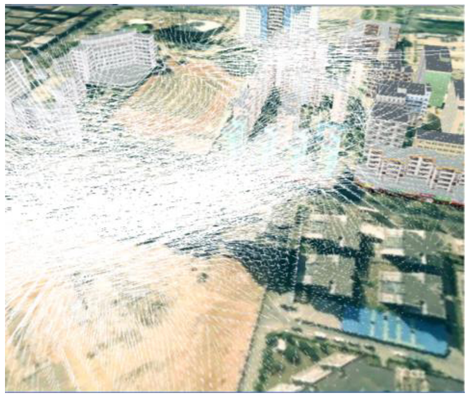
(b)

FIGURE 2. The polluted gas visualization rendering effect.

Experiment 2 *(The Polluted Gas Spread Experiment Based on Cloud-Based Virtual Earth Platform):* In a virtual city scene of ShenZhen city in China, this experiment simulates the spread of polluted gas on User terminal device 1, without adopting GPU accelerated method. In this scenario, polluted gas leaks massively out from a point source, spreading over the city. The experimental time for the spread was 15 minutes, divided into three stages. During first stage, the [0,4] minute time interval, the initial velocity of gas released from the point sources is 0.5m/s; during the second stage, the [4,7] time interval, because gas spread was affected by a southwest breeze, the transmission velocity in the atmosphere was 1.1m/s; during the third stage, the [6,15] time interval, impacted by southeast wind, polluted gas diffused at the speed of 2.4m/s. We assume that during the spread process, the contaminated gas did not chemically change or experience phase change or exchange heat with the surroundings, and that the speed and direction of wind did not vary with location and altitude change. Figure 3 and Figure 4 show the experimental rendering effect with a coarse grid ($128\times32\times16$) in a wide range scene. The frame rendering rate is 10.3FPS in Pipeline 1, and the frame rendering rate is 128FPS in Pipeline 2.

Experiment 3 *(Fluid Simulation Experiments on Different User Terminal Device):* Base on Pipeline1, the experiments respectively simulated the spread process of polluted gas

(a)



(b)

FIGURE 3. The wind field rendering effect during the polluted gas spread process.

on 3 types of personal terminal device, such as personal computer (PC), laptop, mobile phone. The experimental scenario is set that, a column of thick smoke spreads out from a fired residential building in a virtual urban community, then it roll up and diffuse in the virtual scene along with surrounding airflow. The rendering of spread process animation is based on 65,536 grids in online virtual scene. Figure 5 shows the black smoke spread process in the virtual scene Table 2 figures out the frame rate of fluid simulation on User terminal device 1, User terminal device 2 and User terminal device 3.

Discussion: In Experiment 1, this study adopted an equivalent computing environments to compare with the methods of Shin and Liu. These experiments show that, the solution performance of our method is better than the same other methods.

Experiment 2 was made in a low end computer environment, without any additional hardware acceleration. In cloud-based VR networking, our solution time and visualization meet near real-time simulation demands. If the hardware and graphic processing capability are strengthened and if acceleration methods are adopted, such as parallel computation and programming based on GPU, the real-time simulation

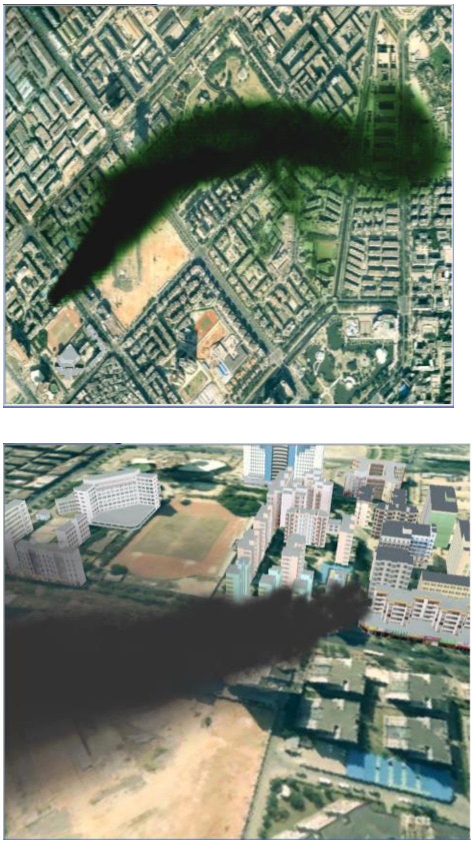


FIGURE 4. The cloud-based online visualization effect during the polluted gas spread process simulation.

TABLE 2. Fluid simulation process experiments on different user terminal device.

User terminal	Device type	Device configuration	Grid scale	Frame rate(fps)
1	PC	Pentium IV core 2 2.66, 2.67GHz; RAM 1GB	64×32×32 （65,536）	11.4
2	Laptop	Intel i5-4300M, 2.6G , core 2; RAM 8G ; GPU HD Graphics 4600	64×32×32 （65,536）	28.6
3	Mobile	ARM ARMv8, 1.84G , core 8; RAM 6G	64×32×32 （65,536）	27.2

of polluted gas spread can be implemented in an even wider range of complex scenarios.

In Experiment 3, the fluid simulation is carried out base on Pipeline1 with different user terminal device. The smoke spread process can be seamlessly browsed and observed from arbitrary angle, and it integrates well with 3D virtual scene. On laptop device and mobile device, the cloud-based online simulation can achieve real-time animation with the frame rendering rate 286 FPS and 27.2 FPS. Both fidelity and visual effect of our simulation method can make good user experience.

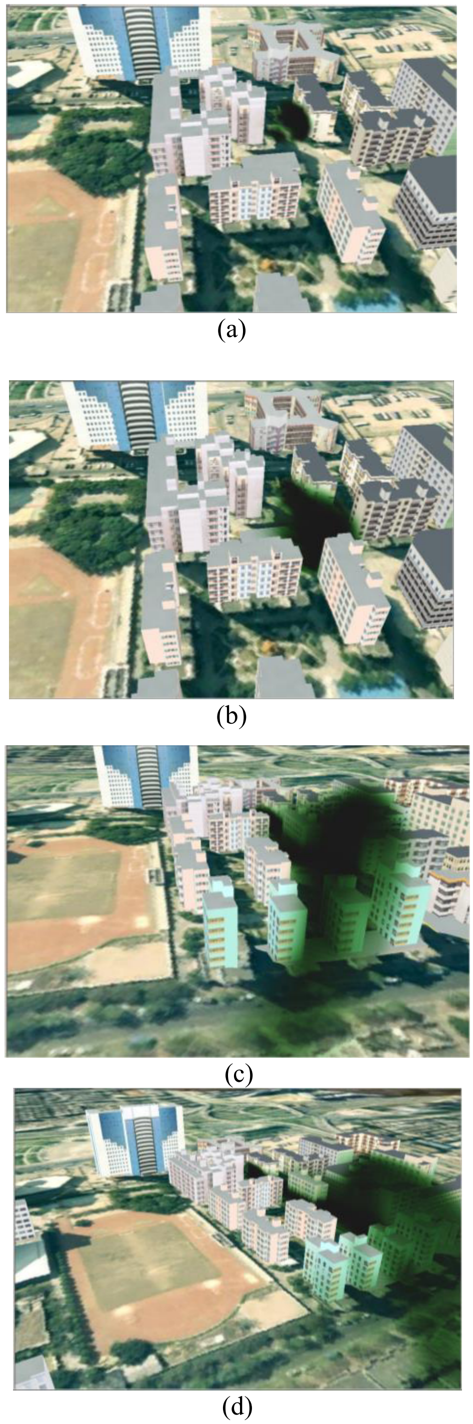


FIGURE 5. The polluted gas spread process in a virtual city.

VI. CONCLUSION

To solve the online simulation and visualization problem for modeling polluted gases in large scale virtual reality scenes, this study proposes a simulation model of a polluted gas spread process and introduces a fast solution under cloud architecture This model is robust and enables fast calculation. In particular, it meets the requirements of real-time rendering requirements for a network based virtual world, and realistically simulates the movement patterns and space-time characteristics of polluted gases. Two cloud-based solution pipelines are proposed for fluid simulation in VR/AR networking. Through the experiment detailed in this paper, it is demonstrated that the solution performance of our proposed method is better than several other competing methods. Our proposed method can integrate and balance the contradiction between the fidelity of simulation and fast solution, and achieves real-time simulation of polluted gas spread in a large-scale scene with online ordinary PC, laptop, or mobile device. This simulation approach is flexible and takes into account changing environmental factors as well. This study is of great significance to urban computing, policy making and public participation in smart city, such as urban planning, architectural design, environmental protection, fire protection and ventilation calculation.

Future studies could consider the full use of cloud-based GPU programming, and other cloud computing resources to realize even faster and more robust online fluid simulations, so as to improve system operational efficiency and enhance the visual effects in VR/AR networking In the future, some applications based on fluid dynamic computing in VR/AR networking could be developed to improve public participation in urban planning, policy-making and other fields.

REFERENCES

[1] E. Jahanbakhsh, C. Vessaz, A. Maertens, and F. Avellan, "Development of a finite volume particle method for 3-D fluid flow simulations," *Comput. Methods Appl. Mech. Eng.*, vol. 298, pp. 80–107, Jan. 2016.

[2] A. J. P. Gomes, "A total order heuristic-based convex hull algorithm for points in the plane," *Comput.-Aided Des.*, vol. 70, pp. 153–160, Jan. 2016.

[3] A. Widiatmojo, K. Sasaki, N. P. Widodo, Y. Sugai, A. Y. Sahzabi, and R. Nguele, "Predicting gas dispersion in large scale underground ventilation: A particle tracking approach," *Building Environ.*, vol. 951, pp. 171–181, Jan. 2016.

[4] P. Danišovič, E. Jančařková, J. Šrámek, and M. Zuziak, "Fire spread models and tunnel traffic & operation simulator," *Procedia Eng.*, vol. 192, pp. 92–95, Jun. 2017.

[5] J. Wen and H. Ma, "Real-time smoke simulation based on vorticity preserving lattice Boltzmann method," *Vis. Comput.*, vol. 36, pp. 1025–1035, May 2018. doi: 10.1007/s00371-018-1514-x.

[6] J. Mark, "Real-time cloud simulation and rendering," *ACM Special Interest Group Comput. Graphics Interact. Techn.*, no. 222, 2005.

[7] J. He, X. Chen, Z. Wang, C. Cao, H. Yan, and Q. Peng, "Real-time adaptive fluid simulation with complex boundaries," *Vis. Comput.*, vol. 26, no. 4, pp. 243–252, 2010.

[8] V. Kwatra *et al.*, "Fluid in video: Augmenting real video with simulated fluids," *Comput. Graph. Forum*, vol. 27, no. 2, pp. 487–496, 2008.

[9] T.-Y. Lee, X. Tong, H.-W. Shen, P. C. Wong, S. Hagos, and L. R. Leung, "Feature tracking and visualization of the Madden–Julian oscillation in climate simulation," *IEEE Comput. Graph. Appl.*, vol. 33, no. 4, pp. 29–37, Jul./Aug. 2013.

[10] Y. Liu, X. Liu, and E. Wu, "Real-time 3D fluid simulation on GPU with complex obstacles," in *Proc. 12th Pacific Conf. Comput. Graph. Appl.*, Seoul, South Korea, Oct. 2004, pp. 247–256.

[11] S. Alimirzazadeh, E. Jahanbakhsh, A. Maertens, S. Leguizamón, and F. Avellan, "GPU-accelerated 3-D finite volume particle method," *Comput. Fluids*, vol. 171, pp. 79–93, Jul. 2018.

[12] K. Reda *et al.*, "Visualizing large, heterogeneous data in hybrid-reality environments," *IEEE Comput. Graph. Appl.*, vol. 33, no. 4, pp. 38–48, Jul./Aug. 2013.

[13] S.-H. Shin and C.-H. Kim, "Target-driven liquid animation with interfacial discontinuities," *Comput. Animation Virtual Worlds*, vol. 18, nos. 4–5, pp. 447–453, 2007.

[14] A. Studziński and K. Pietrucha-UrbanikK, "Simulation model of contamination threat assessment in water network using the epanet software," *Ecological Chem. Eng. S*, vol. 23, no. 3, pp. 425–433, 2016.

[15] L. Tan, Z. Zhu, F. Ge, and N. Xiong, "Utility maximization resource allocation in wireless networks: Methods and algorithms," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 7, pp. 1018–1034, Jul. 2015.

[16] M. Wicke, M. Stanton, and A. Treuille, "Modular bases for fluid dynamics," *ACM Trans. Graph.*, vol. 28, no. 3, 2009, Art. no. 39.

[17] H. Wu, Z. He, and J. Gong, "A virtual globe-based 3D visualization and interactive framework for public participation in urban planning processes," *Comput., Environ. Urban Syst.*, vol. 34, no. 4, pp. 291–298, 2010.

[18] N. Xiong, X. Jia, L. T. Yang, A. V. Vasilakos, Y. Li, and Y. Pan, "A distributed efficient flow control scheme for multirate multicast networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 9, pp. 1254–1266, Sep. 2010.

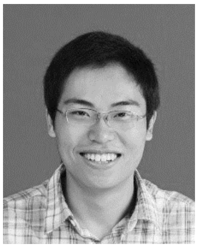
[19] D. M. Young, *Iterative Solution of Large Linear Systems*. New York, NY, USA: Dover.

[20] Y. Zhao, F. Qiu, Z. Fan, and A. Kaufman, "Flow simulation with locally-refined LBM," in *Proc. Symp. Interact. 3D Graph. Games*, Apr./May 2007, pp. 181–188.

[21] W. Tian, X. Han, W. Zuo, and M. D. Sohn, "Building energy simulation coupled with CFD for indoor environment: A critical review and recent applications," *Energy Buildings*, vol. 165, pp. 184–199, Apr. 2018.

ZHENGWEI HE received the Ph.D. degree from Wuhan University, China. He is currently an Associate Professor with the School of Navigation, Wuhan University of Technology, Wuhan, China. He is also a member of the Key Laboratory of Hubei Inland Shipping Technology and the National Engineering Research Center for Water Transport Safety, China. His research interests include traffic big data processing and mining, maritime information systems, and traffic environment simulation. His focuses are on artificial intelligence application technology, deep learning, and smart navigation.

LAN YOU is currently an Associate Professor with the School of Computer Science and Information Engineering, Hubei University, Wuhan, China. Her research interests include virtual geographical environment, SOA, and cloud GIS.



RYAN WEN LIU is currently an Associate Professor with the School of Navigation, Wuhan University of Technology, Wuhan, China. His research interests include computer vision, machine learning, maritime big data mining, and visual analysis.

FAN YANG is currently pursuing the M.Sc. degree with the School of Navigation, Wuhan University of Technology, Wuhan, China. His research interests include traffic data mining, machine learning, artificial intelligence, and maritime information systems.

JIE MA is currently an Associate Professor with the School of Navigation, Wuhan University of Technology, Wuhan, China. His research interests include cyber-physical systems and data driven intelligent transportation systems.

NAIXUE XIONG is currently an Associate Professor with the College of Intelligence and Computing, Tianjin University, Tianjin, China, and the School of Computer Science, Northeastern University, Boston, MA, USA. His research interests include computer science, big data processing and mining, and the Internet of Things Technology. His focus is on deep learning and artificial intelligence application technology.