International Journal of Applied Earth Observations and Geoinformation 103 (2021) 102507

Contents lists available at ScienceDirect

International Journal of Applied Earth Observations and Geoinformation

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

A chain navigation grid based on cellular automata for large-scale crowd evacuation in virtual reality

Pei Dang[a], Jun Zhu[a,*], Saied Pirasteh[a], Weilian Li[a,b], Jigang You[a], Bingli Xu[c], Ce Liang[d]

[a] Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu 611756, China
[b] Lab for Geoinformatics and Geovisualization (g2lab), HafenCity University Hamburg, Henning-Voscherau-Platz 1, Hamburg 20457, Germany
[c] Department of Information and Communication, The Army Academy of Armored Forces, Fengtai District, Beijing 100000, China
[d] Institute of Computing Technology, China Academy of Railway Sciences Co, Ltd, Beijing 100081, China

ARTICLE INFO

Keywords:
Evacuation simulation
Virtual reality
Chain navigation grid
Cellular automata
Vertex raster

ABSTRACT

Large-scale crowd evacuation simulation in virtual reality is of great significance for emergency response. However, the existing large-scale crowd evacuation simulation methods are challenging to apply to the virtual reality scene. Thus, the efficiency of large-scale crowd rendering in virtual reality needs to be improved. Here we proposed a chain navigation grid for virtual reality large-scale crowd evacuation simulation. The objective of this study was to apply traditional cellular automata crowd evacuation simulation algorithms to virtual reality scenes through a chain navigation grid. Combined with the characteristics of crowd movement, we proposed the vertex raster rendering method to efficiently visualize large-scale crowds in virtual reality scenes. Finally, we constructed two evacuation scenes and carried out experimental analyses. The results show that the chain evacuation navigation grid can transform the discrete crowd movement of cellular automata into continuous crowd movement in virtual reality and can connect multiple cellular automata corresponding to multi-floors. Compared with the evacuation results of Pathfinder software, the chain evacuation navigation can add the factor of environmental familiarity. Therefore, this study improved evacuation simulation by increasing about 6% efficiency in time and computing process. Compared with the traditional crowd evacuation rendering method, the vertex raster rendering method increases the maximum number of people rendering from 2 times to 31 times.

1. Introduction

Emergencies such as fire are very likely to cause large-scale casualties in immense public places (Chen et al., 2021; Yibin et al., 2020, 2021; Zhong et al., 2008) (e.g., shopping malls, subway stations). These places have characteristics like dense personnel, a narrow environment and few exits (Shi et al., 2009). For instance, a fire broke out in a shopping mall in Harbin, Heilongjiang Province, China, on January 2, 2015, with a burned area of 11,000 m^2 and a collapsed area of 3000 m^2 had caused 5 deaths and 14 injuries. Therefore, scientific and efficient evacuation simulation can provide a basis for the formulation of emergency plans and the protection of people's lives and property (Yoo and Choi, 2019; Wei et al., 2021).

Traditional evacuation exercise is time labour-intensive, time-consuming, and costly, although it can improve people's safety awareness and increase emergency knowledge. Virtual reality technology is one of the tools to study virtual geographic environments and is widely used in emergency evacuation (Chen et al., 2013; Lin et al., 2013a,b; Lü et al., 2018; Voinov et al., 2018). Virtual reality technology has the characteristics of immersion, interactivity and conception and provides users with immersive feelings and understanding beyond reality. Virtual reality evacuation simulation can make users get greater participation and stronger environmental perception (Lovreglio et al., 2018), more effective acquisition and retention of evacuation knowledge (Weilian et al., 2020; Wouters et al., 2013). Therefore, it is necessary to carry out evacuation simulation through virtual reality technology (Feng et al., 2018). Different from browsing the scene through the screen and virtual reality devices will allow observing more details. On the other hand, the number of people in a virtual reality scene determines the user experience (Dickinson et al., 2019). Therefore, crowd evacuation simulation in virtual reality needs higher crowd evacuation calculation efficiency and rendering efficiency. However, the current crowd evacuation simulation algorithms and rendering methods cannot achieve large-scale crowd evacuation simulation in virtual reality. Thus, many researchers

* Corresponding author.
E-mail addresses: zhujun@swjtu.edu.cn (J. Zhu), sapirasteh@swjtu.edu.cn (S. Pirasteh), vgewilliam@my.swjtu.edu.cn (W. Li).

https://doi.org/10.1016/j.jag.2021.102507
Received 18 June 2021; Received in revised form 5 August 2021; Accepted 16 August 2021
Available online 30 August 2021

attempted virtual reality evacuation using small-scale crowds (Van Kerrebroeck et al., 2017; Guo et al., 2020; Feng et al., 2020). Nevertheless, it is necessary to carry out large-scale crowd evacuation simulations in the virtual reality scene. When using virtual reality technology to conduct the evacuation process, the evacuation model is the foundation, and evacuation efficiency is the key for emergency response (Bourhim and Cherkaoui, 2020).

Current evacuation models include macroscopic and microscopic models. The macroscopic model considers the evacuees as a whole (Piccoli and Tosin, 2011; Li et al., 2018). However, it is difficult to describe the evacuation behavior of individuals (Li et al., 2019). The latter regards individuals in the crowd evacuation as the research object and expresses individuals' evacuation behavior is based on differential equations or other rules (Bellomo and Dogbe, 2011).

The social force model and cellular automata model are commonly used microscopic models (Yuan and Tan, 2007; Wan et al., 2014). However, the social force model algorithm is complex. The computing speed decreases geometrically with the increase of the number of people (Wei-Guo et al., 2006; Zanlungo et al., 2011). This model is not suitable for large-scale crowd evacuation simulation. The cellular automata model has the advantages of a simple solution, strong parallel computing ability and strong expansibility (Li et al., 2019). Cellular automata are more suitable for large-scale evacuation simulation (Yang et al., 2002; Weifeng and Hai, 2007). It also further improves the accuracy of large-scale crowd evacuation simulation results by adding factors such as panic mood and behavior (i.e., movement, direction, and exit path) (Lizhong et al., 2003; Yang et al., 2005; Hu et al., 2018). Therefore, cellular automata are more suitable for large-scale and high-density population simulation (Ji et al., 2018; Feliciani and Nishinari, 2016). Cellular automata are widely used in disaster management, a common Geographical Information System (GIS) model for disaster simulation (Avolio et al., 2000; Abdalla and Li, 2010; Liang and Gao, 2010). However, when cellular automata are applied to the simulation of large-scale crowd evacuation in virtual reality, it will face the following challenges. (1) How to transform discrete simulation into a continuous process. The calculation process of the cellular automata model is based on time steps. Thus, the calculation process is discrete in time and space, while the movement of people in virtual reality scenes is continuous (Pelechano and Malkawi, 2008; Sirakoulis, 2015). (2) How to apply a 2D grid to a multi-floor scene (Guo and Huang, 2008; Gwizdałła, 2015; Zhao et al., 2021)? The cellular automata represent evacuation scene based on a two-dimensional plane, but virtual reality evacuation scene usually has multi-floor structure. (3) How to efficiently render large-scale crowd simulation process in virtual reality scene? A high frame rate is necessary for virtual reality, which helps users have a better sense of immersion and experience (Fu et al., 2021).

In order to solve the above problems, this study proposes a chain navigation grid to solve transforming the discrete simulation process into the continuous process and applying cellular automata to a multi-floor scene. In addition, this study also used the vertex raster rendering method to visualize the evacuation calculation results efficiently in virtual reality. Finally, we verified the method in the subway evacuation scene constructed by virtual reality and compared the evacuation efficiency with Pathfinder software.

Table 1
Dataset for initialization of cellular automata.

Dataset Name	Detailed information
Building data	Floor information Movable area of evacuees
Obstacle data	Obstacle position Obstacle area Obstacle distribution
Evacuees data	Number of evacuees Evacuee's position Characteristics of evacuees

The rest of the article provides the methodology and design of the research frameworks explained in Section 2. This Section describes the chain navigation grid based on cellular automata and introduces the vertex raster rendering method. Section 3 delivers the experimental case area and introduces the setting of experimental parameters and analysis. We discuss the experimental results in Section 4 and finally wrap up the study in conclusion Section 5.

2. Methodology

2.1. Research framework overview

First, we created a dataset (Table 1) to initialize cellular automata. Then, we filtered and interpolated the results of cellular automata to construct a chain navigation grid. We then used a chain navigation grid to control the crowd's movement. The updated location of the crowd is added to the next cellular automata calculation as new data. Next, we processed the crowd model and animation to generate shader files and raster image data and then visualize the evacuees in virtual reality by vertex raster rendering method. Finally, we performed experiments and analyzed the results. Fig. 1 depicts the overview of the research framework.

2.2. Chain navigation grid based on cellular automata

2.2.1. Construction of cellular automata evacuation model

During the evacuation process, many factors such as the psychology, physiology of evacuees, the trend of crowd movement need to be characterized in the cellular automaton (Song et al., 2005; Alizadeh, 2011). The cellular automata evacuation model divides the evacuation area into grids of the same size (generally 0.5 m × 0.5 m). Each grid can only accommodate one person simultaneously and has only three states, including occupied by evacuees, occupied by obstacles and empty. There are two kinds of neighborhoods commonly used in cellular automata applying in the evacuation process and analysis. (a) Von Neumann neighborhoods and (b) Moore neighborhoods. This study uses the von Neumann neighborhoods; that is, the central cell has four upper, lower, left, and right neighborhoods.

In the process of evacuation, people always look for the nearest exit as the evacuation target. Therefore, according to the direction of people's movement, there will be a candidate target set around the evacuees. The closer the candidate set is to the exit, the more attractive the cell will be otherwise, it will be smaller. The attraction of the exit to the candidate cell is calculated as follows (Yuan and Tan, 2007):

$$P_{dis(i,j)} = \frac{d_{max} - d_{ij}}{d_{max} - d_{min}} \tag{1}$$

where $P_{dis(i,j)}$ is the exit attraction probability of the candidate cell (i, j), d_{ij} is the distance between the cell (i, j) and the exit. When there are multiple exits, the value is the minimum distance from the cell to each exit. d_{max},d_{min} represents the maximum and minimum distance from each candidate cell to the exit, respectively.

When the evacuees are in a dangerous environment, panic, fear, and other emotions therefore, the situation will make it difficult for them to judge and analyze the environment. Hence, this situation results in the phenomenon of conformity in the process of evacuation. Thus, in the cellular automata model, we considered the conformity factor influence. This factor influence is called directional attraction and is used to reflect the conformity psychology of personnel in an emergency. We used the following Eq. (2) to calculate the probability of direction attraction.

$$P_{dir(i,j)} = \frac{n_{ij}}{\sum_{k=1}^{m} n_k} \tag{2}$$

where $P_{dir(i,j)}$ is the probability of direction attraction of a cell with coordinates (i, j) in the candidate cell, $\sum_{k=1}^{m} n_k$ is the total number of

Table 2
Undetermined coefficient values.

Undetermined coefficient	Value
a	0.00823386
b	1.63691201e+03
c	1.86062789e+02
d	53.98641114
e	2.79438249e+05
f	1.37192577
g	5.87114662
h	6.96620857e+03
i	1.40075479e+04

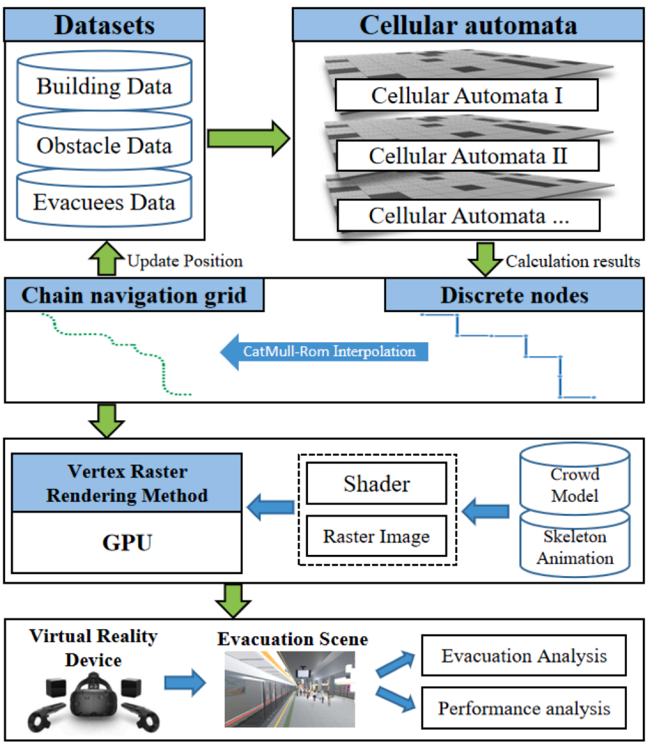

Fig. 1. Overview of the research framework.

personnel passing through all candidate cells as of the current time, the value of m is 4 because we used the von Neumann domain in this study. n_{ij} represents the total number of personnel passing through the cell (i, j) up to the current calculation step. If the denominator is 0, the value of P is specified as 0.

Furthermore, those familiar with the environment may choose the route they know and are more likely to select the nearest exit in the evacuation process. In contrast, those who are not familiar with the environment tend to follow the crowd. Therefore, this study introduces λ to express the degree of familiarity of evacuees with the environment (Yuan and Tan, 2007). The values λ range between 0 and 1, which means the degree of familiarity of individuals with the environment. The value is from completely unfamiliar (i.e., 0) to very familiar (i.e., 1). After introducing the λ factor, the comprehensive attraction of candidate cells can be defined as:

$$P_{dis+dir(i,j)} = \lambda P_{dis(i,j)} + (1-\lambda)P_{dir(i,j)} \tag{3}$$

When the evacuation scene has multiple exits, the above factors need to be considered, but the number of people and obstacles between the candidate cell and the exit (i.e., exits with fewer barriers) is easier to choose. Therefore, this study used the patency to express the convenience of the cell to the exit (Yang et al., 2002), and the patency probability of the candidate cell is defined as:

$$P_{smo(i,j)} = \frac{N_{max} - N_{ij}}{N_{max} - N_{min}} \tag{4}$$

where $P_{smo(i,j)}$ is the patency probability of the candidate cell whose coordinates are (i, j), N_{ij} is the number of grids occupied by people or obstacles in the field of view (a semicircle with radius R) of the candidate cell whose coordinates are (i, j). N_{max} and N_{min} is the maximum and minimum of the candidate cells (see supplementary material Fig. 1).

When we use the comprehensive attraction formula to calculate probability, the evacuees will move back and forth between the two cells. In order to avoid this situation, the weakening factor f_{times} is introduced to record the cells that the evacuees pass through. These cells are inversely proportional to the number of passes through the same cell. Therefore, we can use the corrected formula of comprehensive attraction as follows:

$$P'_{dis+dir(i,j)} = P_{dis+dir(i,j)} \cdot f_{times} \tag{5}$$

In the calculation, there will also be the case of multiple personnel competing for the same cell. Typically, we generate a random number for the personnel competing for the cell, and the evacuees with the biggest random number will enter the cell. Therefore, cellular automata cannot be directly applied to virtual reality scenes. In this study, the results of cellular automata are expressed in the virtual reality scene through the chain evacuation navigation grid.

2.2.2. Chain navigation grid

The chain navigation grid connects the iterative results of cellular automata. It stores them in the chain data structure as a data conversion layer between cellular automata and the virtual reality scene.

The simulation of cellular automata takes the step size as the unit, and the distance of each movement is the side length of the cell (0.4 m). At the end of cellular automata calculation, central cell A will move to a candidate cell B (Fig. 2).

The start point and endpoint of the chain navigation grid are set as the center coordinates of the two cells (x_1,y_1) and (x_2,y_2). The condition for people to move between cells is: (a) when B is empty, and the personnel in A begin to move to B (A shows empty, and B demonstrates a cell occupied by personnel). (b) If there are already individuals in B, but

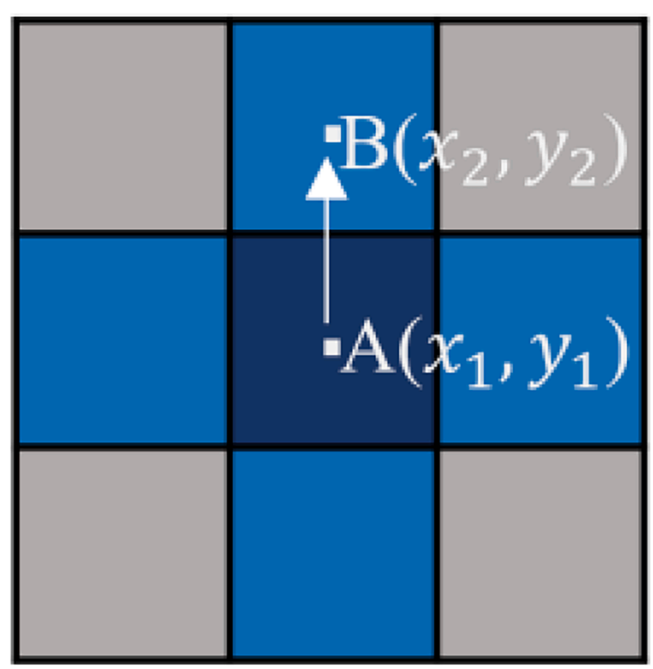

Fig. 2. Personnel movement.

the candidate cell B in the next step is empty, then A can move forward, and B is regarded as empty. (c) If there are already people in B and the candidate cell B is occupied by other personnel whose next step stays in place, then neither A nor B cells can move (Fig. 3).

The cellular automata model does not consider the difference in individual movement speed, so it needs to modify the rules of cellular competition. When cellular competition occurs, it no longer determines who can move by generating random numbers but calculates the movement speed of the person in the competing cell and selects the cell with the highest movement speed to enter, and this is more consistent with reality.

After introducing different moving speeds of crowd individuals, cellular automata need to wait for all individuals and move to the predetermined position before the next step of the calculation. The efficiency of the above process depends on the person who moves the slowest, which is not applicable in the virtual reality scene because other faster evacuees cannot stop moving and wait. So, in the interval of waiting for the next cellular automata calculation, the crowd moves depending on the data in the chain navigation grid.

On the contrary, the cellular automata calculation result does not change much compared with the last calculation result, which requires higher calculation performance. Therefore, the time interval should be within a reasonable range. Thus, the strategy adopted in this study is to plan a chain navigation grid for each evacuation person (Fig. 4).

After the first step of cellular automata calculation, the movement order of (0,1) cells is (0,2), (0,3), (3,1). For evacuee A, we take (0,2), (0,3), (3,1) as the three path points in his navigation route, which is called the chain navigation grid. The paths of other personnel are also planned accordingly. Four factors determine the length of the chain navigation grid. They are (i) evacuation area, (ii) the number of evacuees, (iii) maximum moving speed of crowd, and (iv) number of obstacles. These four parameters are non-linear related to the maximum calculation time, the length of the chain navigation grid G_l can be described using Eq. (6).

$$G_l = \left[\frac{\mu v_{max}}{1000}\left(a\left(\frac{S_e}{l}\right)^2 + bP^2 + cO^2 + d\frac{S_e}{l}P + ePO + f\frac{S_e}{l}O + g\frac{S_e}{l} + hP + iO\right)\right]\# \tag{6}$$

where v_{max} is the maximum moving speed of the crowd (m/s), S_e means the evacuation area (m^2), P is the ratio of people to cells. O is the ratio of obstacles to cells, l is the size of cells (m), μ is the adjustment coefficient based on computer performance, and a, b, c, d, e, f, g, h are polynomial coefficients. Finally, the calculation results are rounded up.

The moving direction calculated by cellular automata is only up, down, left, right. These moving directions are obviously unreasonable and unnatural, because in reality, the moving direction of the crowd has more angles. Therefore, we need to interpolate the chain navigation grid nodes obtained in the previous step to get a smoother and more visually reliable scattered route. Firstly, the chain navigation grid nodes need to be filtered, and the nodes at the corners need to be eliminated. Because people will not turn after accurately reaching the center of the specified grid but gradually deviate to the turning direction in the moving process, the points at the corners are discarded. After discarding the points, new nodes need to be added, as shown in Fig. 5.

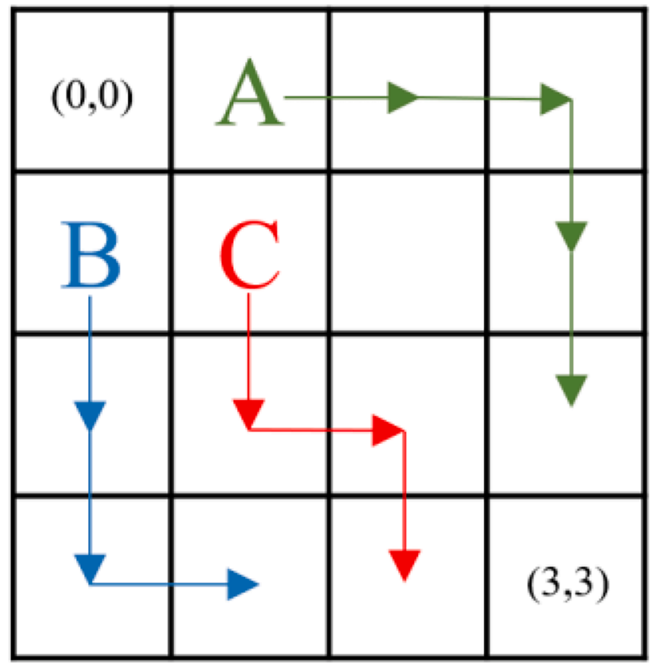

Fig. 4. Chain navigation grid.

P_{i+1} node needs to be discarded as a turning node, $P^{'}_{i+1}$ and $P^{''}_{i+1}$ nodes are added as new approach points. P_{in} is the intersection of P_i, P_{i+1} and P_{i+2}, Move the P_{in} along the X and Y axes towards the P_{i+1} by a quarter of the cell side length to get $P^{'}_{i+1}$ and $P^{''}_{i+1}$. Because the movement of evacuees needs to pass through the key nodes of the chain navigation grid, and the evacuees often move with the shortest path, this study used Catmull-Rom algorithm (Eq. (7)) to interpolate the chain navigation grid and to make the evacuation path smoother and more realistic.

$$p(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\tau & 0 & \tau & 0 \\ 2\tau & \tau-3 & 3-2\tau & -\tau \\ -\tau & 2-\tau & \tau-2 & \tau \end{bmatrix} \begin{bmatrix} p_{i-2} \\ p_{i-1} \\ p_i \\ p_{i+1} \end{bmatrix} \tag{7}$$

Where u is the interpolation point coordinate, τ is the degree of distortion of the curve and the value range is (0–1), in this research, the value of τ is 0.2, p_{i-2},p_{i-1},p_i,p_{i+1} are the coordinates of four points required for interpolation.

Because cellular automata can only represent a two-dimensional

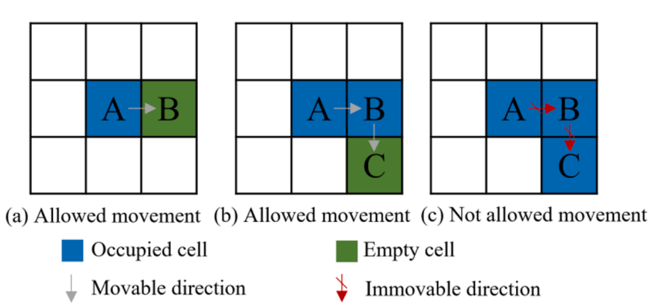

Fig. 3. The direction of personnel movement.

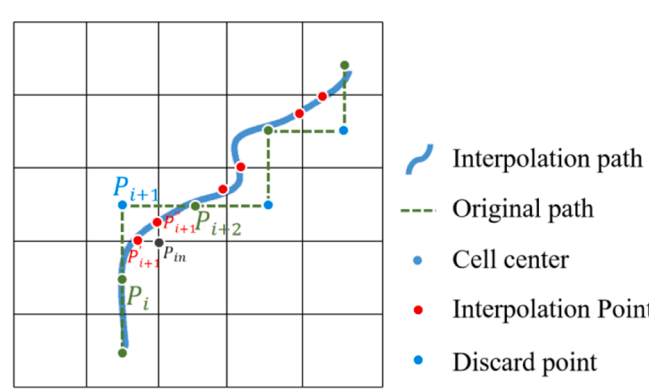

Fig. 5. Chain navigation grid interpolation.

plane, it is necessary to consider the problem of multi-floor in virtual reality evacuation scene. As shown in Fig. 6, when the cellular automata model is performed on multiple floors, each floor needs to be calculated as a separate plane and connected by a chain navigation grid.

In the case of three floors, the stairway of each floor is regarded as the start or endpoint of the corresponding cellular automata. When Eq. (3) is applied to calculate the exit attraction, the distance to exit should include the distance (d1, d3, d5) through each floor and the distance (d2, d4) of the up and downstairs. When the person is on the stairs, it only navigates it through the chain navigation grid and does not participate in calculating any floor cellular automaton. Only when the person leaves the stairs and enters a particular floor, it can participate in the calculation of the corresponding cellular automaton of the floor.

Through the above methods, the crowd behavior is more authentic because it considers the psychological, physiological, environmental, and other factors of evacuation personnel. On this basis, we need to consider further how to render a large-scale crowd model in a virtual reality scene.

2.3. Vertex raster rendering method

Large-scale crowd rendering is a key problem in virtual reality. The rendering efficiency determines the number of people in the scene at the same time. The improvement of the number of people can enhance the immersion of virtual reality. The evacuation crowd model needs to be built according to gender, age and behavior. The evacuation crowd model can use the same skeleton animation when running, walking and standing. In traditional virtual reality crowd rendering, each individual is bound with skeleton animation, submitted to Central Processing Unit (CPU) for checking (Li et al., 2021), and then rendered by Graphic Processing Unit (GPU), which is inefficient. In this study, the evacuation crowd model is divided into two types according to gender, and different textures and model scales distinguish the individual. All the evacuation crowd models share three kinds of skeleton animation (i.e., running, walking, and standing).

After the above processing, we reduced the basic model of evacuees to two kinds. Each model has three types of actions, and the rendering efficiency of the crowd can be improved by GPU instantiation technology. However, this rendering method still needs to use the traditional animation rendering method, and the relevant model data must go through the steps of CPU check. This study uses a raster to store the position of each vertex in each frame of the animation. The GPU can directly read the raster file to decode and render, improving the efficiency of crowd rendering by 2–31 times. Fig. 7 shows the vertex raster rendering method process.

Crowd examples include static and animation models. First, build a

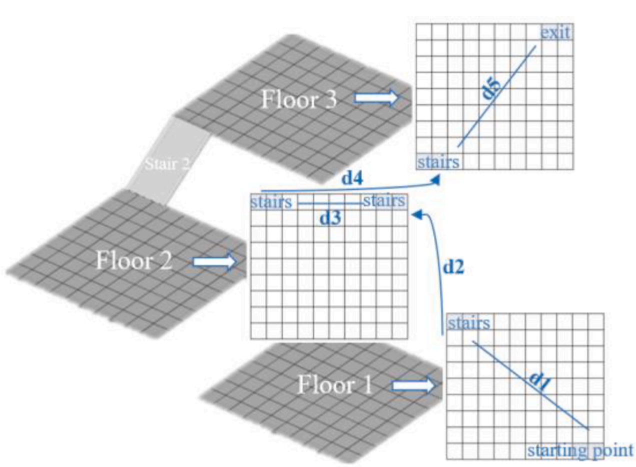

Fig. 6. Connecting multiple floors.

raster with length and width of l and h, respectively. If the static model needs to be rendered, h = 1; if the animation model needs to be rendered, h is the number of animation frames. Next, the colors R, G and B of each pixel in each row of the raster are respectively stored with the values of x, y and z of the corresponding vertex. The former values range is from 0 to 255. In contrast, the latter's values are uncertain, therefore, we need to use Eq. (8) to map the coordinate data to 0–255.

$$R = \left(x - Min_{xyz}\right)^*F$$
$$G = \left(y - Min_{xyz}\right)^*F \tag{8}$$
$$B = \left(z - Min_{xyz}\right)^*F$$
$$F = 256/\left(Max_{xyz} - Min_{xyz}\right)$$

Min_{xyz} and Max_{xyz} is the minimum and maximum of the data coordinates x, y and z of the model vertex. F is the scaling factor stored in A in the RGBA (red, green, blue, alpha) color mode of the raster. The F values range is 0–255. The coordinate range that RGBA can be stored is 0–65025. The model coordinates that are not in this range need to be scaled and stored. The vertex position of each frame of the animation model is correspondingly stored in each row of the raster (see supplementary material Fig. 2).

The static and animation model are stored in the form of the raster. The texture coordinates corresponding to each vertex in the model are unchanged. We used a shader to read the raster and texture information and then directly rendered the static model vertex-by-vertex with the vertex coordinates and texture coordinates. For the animation model and make rendered frame by a vertex at a certain rate (when the animation model is converted to raster data). The playback time of each frame is calculated by following Eq. (9):

$$t_{perFrame} = \frac{t_{animation}}{n_{frames}} \tag{9}$$

where $t_{perFrame}$ is the time used to draw each frame, which is stored in the shader file in the form of float data to facilitate the reading. $t_{animation}$ is the total duration of the animation, and n_{frames} is the number of animation frames. When the shader is executed, each row of data in the raster is read at a time interval of $t_{perFrame}$, and the pixel values in each row of data are converted to vertex coordinates. Each vertex color is determined by the UV (a 3D modeling process of projecting a 2D image to a 3D model's surface) coordinates stored in the material file.

3. Experiment and results

3.1. Setting of experimental parameters

In order to set the appropriate parameters in Eq. (6), this study tests the computational performance of traditional cellular automata and sets the parameters according to the test results. We designed two experiments: the relationship between the size of cellular automata (evacuation area) and the single-step maximum calculation time.

The cell size of the two tests is 0.4 m × 0.4 m and records the maximum interval between the two calculations of cellular automata. The area of the region of interest (i.e., location test) (a) is 50,000 square meters, and the proportion of evacuees in location test (b) is 5%. The distribution of people and obstacles is completely random. Table 3 and Table 4 illustrate the set the parameters of the evacuees. The test results are shown in Fig. 8.

The maximum speed of the crowd is 1.6 m/s. If the traditional cellular automata calculation method is adopted, then the fastest-moving person reaches the predetermined position and starts to calculate the next step. The single-step calculation time cannot exceed 0.267 s, which will lead to the maximum number of cellular automata simulations, not more than 6000 people and the simulation area not more

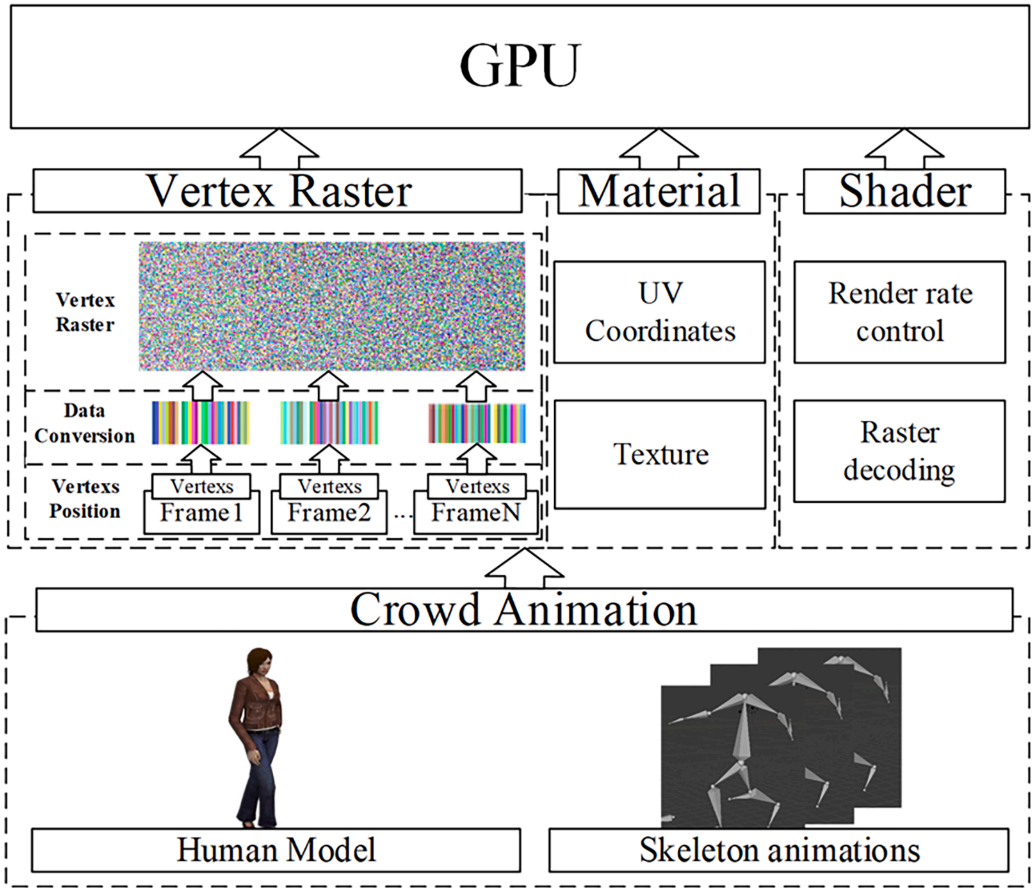

Fig. 7. Rendering process based on vertex raster.

Table 3
Population Proportion.

Character classification	The proportion of the population (%)
Adult male	35
Adult female	35
The aged	20
Teenagers	10

Table 4
Physiological Characteristics Parameters.

Parameter	Teenagers	Adult male	Adult female	Old people
Age(year)	13–18	19–65	19–65	>65
Walking speed(m/s)	0.90	1.20	1.00	0.70
Running speed(m/s)	1.30	1.60	1.40	1.00

than 12,000 m^2. The computational performance cannot satisfy the requirements of large-scale crowd evacuation simulation. This calculation also proves the necessity of using a chain navigation grid. We also tested the influence of the ratio of obstacles on the performance of cellular automata and combined it with the above test results. We have solved the number of undetermined coefficients in Eq. (6). The parameters are shown in Table 2, and these undetermined coefficients are suitable for the case of 1600 $m^2 < S_e <$ 58,000 m^2,1% < P < 5%,1% < O < 5% in Eq. (6).

The initial position of the evacuees' character models is randomly distributed. We used Table 3 to set the proportions of different character types. We also used Table 4 to set the physiological parameters of the evacuated crowd.

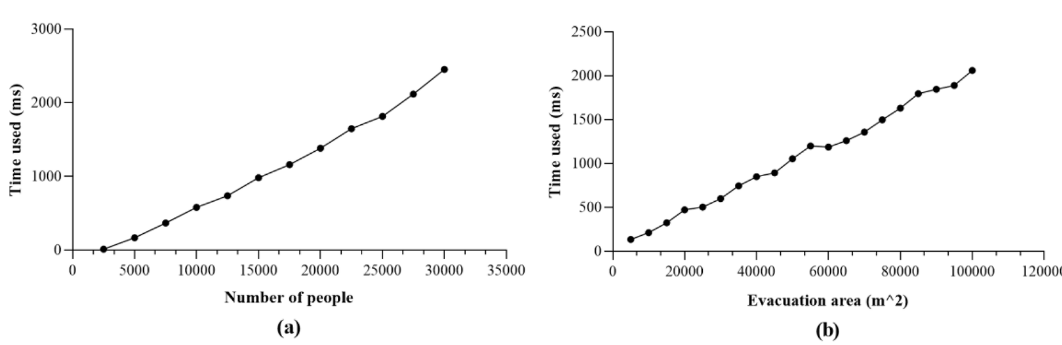

Fig. 8. (a) Number of evacuees with calculation time and (b) Evacuation area with calculation time.

3.2. Experimental cases

Based on the key methods mentioned above, we selected a subway station as the case area. It consists of three layers: platform, passageway, and exit (see supplementary material Fig. 3). The area of the platform is about 4500 m^2, the area of the aisles is about 3000 m^2, and the exit area is about 1000 m^2. The size of the cellular automata grid is 0.4 m $\times$ 0.4 m. We used a total of about 530,000 cells in this study. The proportion of obstacles and personnel is 1.76% and 0.09% to 1.32%, respectively. The positions of evacuees are randomly distributed in the scene and according to the parameters (Table 3) to initialize the prototype system. The prototype system interface is shown in supplementary material Fig. 4 and includes the tools for "*following mode*" and "*roaming mode*". In the following mode, users can play the role of evacuees and follow other evacuees to exit. Users can move to any position through the VR handle in roaming mode to observe the evacuation process. The software and hardware configuration table depicts in Table 5. The evacuation process diagram is shown in supplementary material Fig. 5. In order to make the results more accurate, we controlled the conditions in the virtual reality evacuation experiment. As a result, Users observe the evacuation process from multiple angles through roaming mode, and the user's moving route and distance remain the same in different experiments. The user experience is shown in Fig. 9.

3.3. Comparative experiment

3.3.1. GPU / CPU performance experiment

We used the experimental group for the chain evacuation navigation grid and vertex raster rendering method to carry out evacuation simulation. In comparison, the control group used the traditional model rendering method in Unity to carry out the evacuation simulation. We compared the rendering performance and CPU or GPU occupancy index of the groups through evacuation simulation of multiple groups of evacuees with different sizes. Fig. 10 illustrates the results we obtained.

The experimental results show that the rendering performance of the control group decreases sharply when the number of the crowd reaches 1000. Furthermore, although the GPU occupancy rate is not high, the frame rate is lower than 100FPS (Frames Per Second). The reason is that the CPU occupancy rate has reached 100%, and the frame rate is lower than 10FPS when the number of the crowd reaches 2000.

3.3.2. Evacuation experiment

We designed a comparative experiment using Pathfinder software version 2019.2 to evacuate. The experimental scene is a shopping mall. The shopping mall evacuation scene has four exits and two floors (Fig. 11). The area is about 1401 square meters with various obstacles and exits.

The experimental group adopts a chain navigation grid to evacuate. The control group is to evacuate through Pathfinder. The parameters of population distribution and moving speed are the same. Evacuation is conducted by 1000, 2000 and 3000 evacuees, respectively, and recorded the remaining number of evacuees in the scene every 10 s (Fig. 12).

Table 5
System Development Environment Configuration.

Equipment and software	Environment configuration	Detailed information
Hardware	CPU	Inter i7-4870HQ
	Memory	8G
	Graphics card	NVIDIA GTX 1070Ti
	Video memory	8G
	VR equipment	Oculus Rift cv1
Software	System	Windows10
	Software	Unity 3D 2018.3.7
		Steam VR
		Visual Studio 2017

Fig. 9. First-person evacuation simulation.

4. Discussion

We can see that the control group reached the performance bottleneck when the number of evacuees came to about 1500, and the frame rate decreased rapidly (Fig. 10). The reason is that the CPU utilization rate has reached 100%, which is caused by the traditional crowd rendering method that requires the CPU to do more resource processing. The CPU / GPU occupancy rate grows more slowly in the experimental group, and there is no performance bottleneck. The reason is that the vertex raster rendering method can directly submit the model to GPU for processing, which effectively balances the resource allocation between CPU and GPU. We only used two kinds of basic models and three types of skeleton animation. Although more evacuee instances add to the scene, they share the same vertex, texture and other resources, and therefore, the CPU needs to schedule these instances. Thus when the number of evacuees increases, the utilization rate of CPU is faster than that of GPU. According to the virtual reality industry's suggestion, when the frame rate is lower than 60FPS, users are prone to dizziness and other problems. When the frame rate is above 60fps, the control group can render about 1500 people, while the experimental group can render about 6000 people.

We can also observe that the experimental group results are similar to straight lines (Fig. 12). The results of the control group are similar to parabolas at the beginning of the experiment. The control group is also similar to the straight line in the later experiment period. It is because the familiarity of the evacuees to the environment is considered in the chain navigation grid. Therefore, the crowd has a clear direction of movement at the beginning of the evacuation. This crowd factor was not taken into account in Pathfinder; therefore, the control group needs to spend more time looking for the exit in the early stage of the experiment. We compared the differences of environmental familiarity factors on evacuation results (Fig. 13).

People familiar with the environment can choose exits more quickly than those unfamiliar with the environment. As a matter of fact, the chain navigation grid considers the situation of the movement of evacuees familiar with the environment and thus will affect the movement of crowd flow. With the evacuation proceeds, evacuees can move more reasonably to each exit and therefore, the congestion situation of each exit is more balanced. In this case, the evacuation efficiency is improved by about 6%. In the middle and later stages of the experiment, we observed that the gradient of the two lines in the experimental and control groups is similar. This gradient indicated that congestion occurs at the exit of both groups in the middle and later stages. Thus, the passing rate of the exit reaches the maximum. Comparing the two experiments revealed that the chain evacuation navigation grid could consider more factors (e.g., environmental familiarity, companionship, fear, etc.) and provide reliable evacuation results.

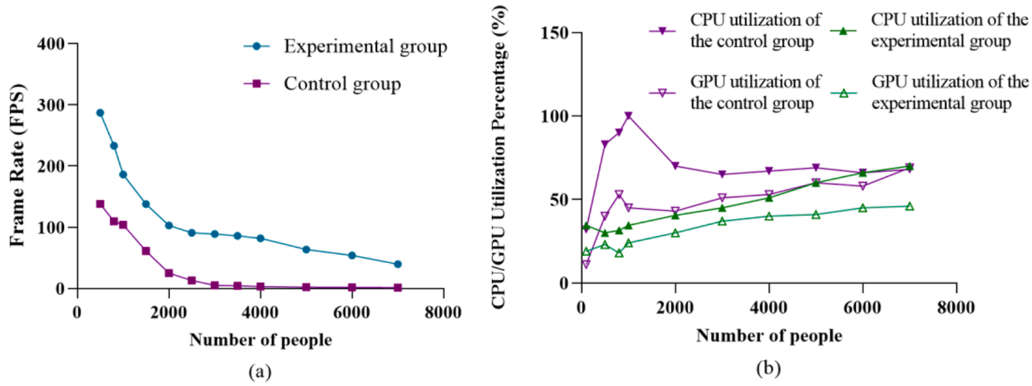

Fig. 10. (a) Average Frame Rate Comparison and (b) Average CPU/GPU Utilization Before and After Optimization.

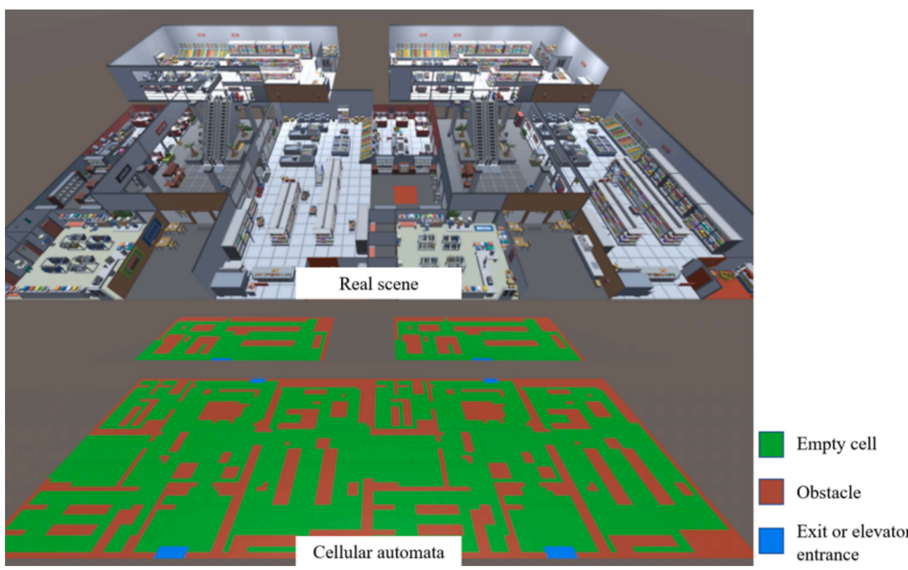

Fig. 11. Evacuation scenario and the construction of cellular automata.

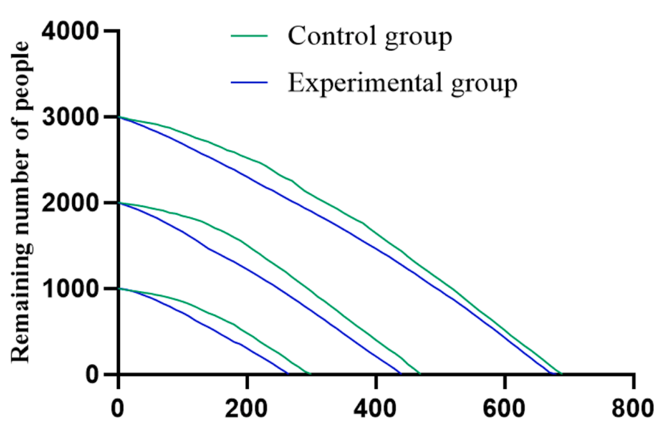

Fig. 12. Comparison of evacuees' surplus.

The chain evacuation navigation grid can make the cellular automata evacuation simulation and the crowd evacuation path execute asynchronously. Thus, the cellular automata ensure the reliability of the evacuation process and results. Furthermore, the vertex raster rendering method ensures the efficient expression of the calculation results in virtual reality. This proposed approach is one of the new methods in the virtual reality large-scale crowd evacuation simulation.

5. Conclusion

This study proposed a chain evacuation navigation grid to reconstruct and reform the calculation results of cellular automata for large-

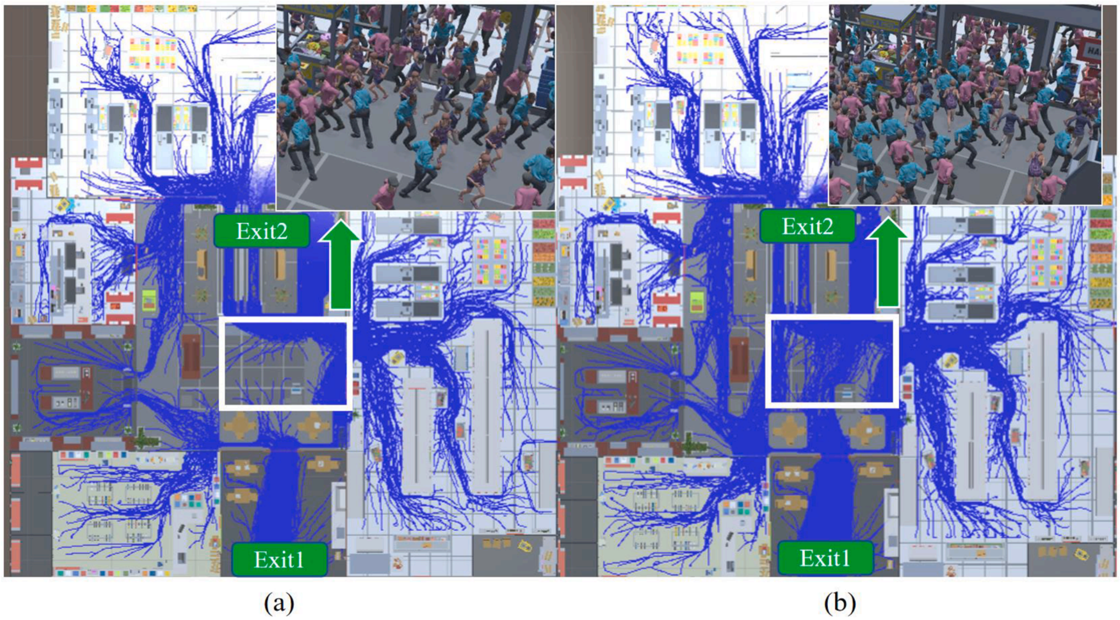

Fig. 13. (a) With environmental familiarity factor (b) Without environmental familiarity factor.

scale crowd evacuation simulation in virtual reality. Compared with the evacuation results of Pathfinder software, this study improved the evacuation efficiency by about 6%. The chain navigation grid makes up for the three defects of cellular automata (i.e., discrete results, multi-floor movement and the same movement speed). At the same time, in order to efficiently visualize the chain navigation grid data of evacuees, combined with the characteristics of crowd evacuation, we proposed a vertex raster rendering method to solve the problem of low rendering frame rate for large-scale crowds. Therefore, the simulation results of cellular automata displayed smoothly and naturally in the virtual reality scene. Finally, we selected a subway station and a shopping mall as evacuation simulation cases and presented the effectiveness and feasibility of our proposed method. The study concluded that cellular automata's discrete evacuation simulation results could be transformed into continuous crowd motion data in virtual reality scenes through interpolation. Thus, we solved the asynchronous problem of traditional cellular automata computing and virtual reality crowd rendering. In addition, the chain navigation grid can reduce the requirements of cellular automata on computer performance and improve the efficiency of large-scale crowd simulation in virtual reality. This improvement avoids the problem of unreliable simulation results caused by the small crowds in virtual reality evacuation.

Moreover, we conclude that the adaptation of the requirements of large-scale crowd evacuation simulation rendering and virtual reality scene rendering requirements changes the traditional model animation rendering method to the method of the vertex raster rendering method. Therefore, it can be used to render the crowd model animation directly by GPU. It can achieve real-time rendering of about 6000 people in ordinary performance computers. Thus, this study contributed the rendering efficiency to 2–31 times greater than the traditional virtual reality rendering method. In other words, we can reveal that our proposed model provides an efficient method to simulate the virtual reality large-scale crowd. Nevertheless, this study concluded that the evacuation and other crowd cases simulations might use the vertex raster rendering method to increase the efficiency.

In future work, we need to optimize the calculation accuracy of the chain navigation grid and the calculation efficiency of the cellular automata. We also need better balance crowd simulation, calculation, and crowd rendering and further improve the quantity and accuracy of large-scale crowd simulations as well.

Funding

This research was funded by the National Natural Science Foundation of China, No. 41871289, the Sichuan Science and Technology Program (2019YFG0460; 2020JDTD0003). The National Natural Science Foundation of China (U2034202).

CRediT authorship contribution statement

Pei Dang: Methodology, Software, Writing – original draft. **Jun Zhu:** Conceptualization, Validation, Writing – review & editing. **Saied Pirasteh:** Validation, Writing – review & editing. **Weilian Li:** . **Jigang You:** Resources. **Bingli Xu:** Resources. **Ce Liang:** Software.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jag.2021.102507.

References

Abdalla, R., Li, J., 2010.Towards effective application of geospatial technologies for disaster management . Int. J. Appl. Earth Obser. Geoinform. 12(6), 405-407. doi: 10.1016/j.jag.2010.09.003.

Alizadeh, R., 2011. A dynamic cellular automaton model for evacuation process with obstacles. Saf. Sci. 49 (2), 315–323. https://doi.org/10.1016/j.ssci.2010.09.006.

Avolio, M.V., Di Gregorio, S., Mantovani, F., et al., 2000. Simulation of the 1992 Tessina landslide by a cellular automata model and future hazard scenarios. Int. J. Appl. Earth Obser. Geoinform. 2 (1), 41–50. https://doi.org/10.1016/S0303-2434(00)85025-4.

Bellomo, N., Dogbe, C., 2011. On the modeling of traffic and crowds: A survey of models, speculations, and perspectives. SIAM Rev. Soc. Ind. Appl. Math. 53 (3), 409–463. https://doi.org/10.1137/090746677.

Bourhim, E.L.M., Cherkaoui, A., 2020. Efficacy of virtual reality for studying people's pre-evacuation behavior under fire. Int. J. Hum. Comput. Stud. 142, 102484 https://doi.org/10.1016/j.ijhcs.2020.102484.

Chen, C., Sun, H., Lei, P., et al., 2021. An extended model for crowd evacuation considering pedestrian panic in artificial attack. Physica A 571, 125833. https://doi.org/10.1016/j.physa.2021.125833.

Dickinson, P., Gerling, K., Hicks, K., et al., 2019. Virtual reality crowd simulation: effects of agent density on user experience and behaviour. Virtual Real 23 (1), 19–32. https://doi.org/10.1007/s10055-018-0365-0.

Fu, L., Zhu, J., Li, W., et al., 2021. Tunnel vision optimization method for VR flood scenes based on Gaussian blur. Int. J. Digit. Earth 1–15. https://doi.org/10.1080/17538947.2021.1886359.

Feliciani, C., Nishinari, K., 2016. An improved Cellular Automata model to simulate the behavior of high-density crowd and validation by experimental data. Physica A 451, 135–148. https://doi.org/10.1016/j.physa.2016.01.057.

Feng, Z., González, V.A., Amor, R., et al., 2018. Immersive virtual reality serious games for evacuation training and research: A systematic literature review. Comput. Educ. 127, 252–266. https://doi.org/10.1016/j.compedu.2018.09.002.

Feng, Z., González, V.A., Trotter, M., et al., 2020. How people make decisions during earthquakes and post-earthquake evacuation: Using verbal protocol analysis in immersive virtual reality. Saf. Sci. 129, 104837 https://doi.org/10.1016/j.ssci.2020.104837.

Sirakoulis, G.C., 2015. The computational paradigm of cellular automata in crowd evacuation. Int. J. Found. Comput. Sci. 26, 851–872. https://doi.org/10.1142/S0129054115400031.

Chen, M., Lin, H., Hu, M., et al., 2013. Real-geographic-scenario-based virtual social environments: integrating geography with social research. Environ. Plann. B Plann. Des. 40 (6), 1103–1121. https://doi.org/10.1068/b38160.

Guo, R.Y., Huang, H.J., 2008. A modified floor field cellular automata model for pedestrian evacuation simulation. J. Phys. A: Math. Theor. 41 (38), 385104 https://doi.org/10.1088/1751-8113/41/38/385104.

Guo, Y., Zhu, J., Wang, Y., et al., 2020. A Virtual Reality Simulation Method for Crowd Evacuation in a Multiexit Indoor Fire Environment. ISPRS Int. J. Geoinf 9 (12), 750. https://doi.org/10.3390/ijgi9120750.

Gwizdałła, T.M., 2015. Some properties of the floor field cellular automata evacuation model. Physica A 419, 718–728. https://doi.org/10.1016/j.physa.2014.10.070.

Hu, J., You, L., Zhang, H., et al., 2018. Study on queueing behavior in pedestrian evacuation by extended cellular automata model. Physica A 489, 112–127. https://doi.org/10.1016/j.physa.2017.07.004.

Ji, J., Lu, L., Jin, Z., et al., 2018. A cellular automata model for high-density crowd evacuation using triangle grids. Physica A 509, 1034–1045. https://doi.org/10.1016/j.physa.2018.06.055.

Li, L., Zhu, W., Hu, H., 2021. Multivisual Animation Character 3D Model Design Method Based on VR Technology. Complexity 2021. https://doi.org/10.1155/2021/9988803.

Li, Y., Chen, M., Dou, Z., et al., 2019. A review of cellular automata models for crowd evacuation. Physica A 526, 120752. https://doi.org/10.1016/j.physa.2019.03.117.

Lovreglio, R., Gonzalez, V., Feng, Z., et al., 2018. Prototyping virtual reality serious games for building earthquake preparedness: The Auckland City Hospital case study. Adv. Engg. Inform. 38, 670–682. https://doi.org/10.1016/j.aei.2018.08.018.

Liang, S., Gao, Y., 2010. Real-time notification and improved situational awareness in fire emergencies using geospatial-based publish/subscribe. Int. J. Appl. Earth Obser. Geoinform. 12 (6), 431–438. https://doi.org/10.1016/j.jag.2010.04.001.

Lin, H., Chen, M., Lu, G., 2013a. Virtual geographic environment: a workspace for computer-aided geographic experiments. Ann. Assoc. Am. Geogr. 103 (3), 465–482. https://doi.org/10.1080/00045608.2012.689234.

Lizhong, Y., Weifeng, F., Weicheng, F., 2003. Modeling occupant evacuation using cellular automata-Effect of human behavior and building characteristics on evacuation. J. Fire Sci. 21 (3), 227–240. https://doi.org/10.1016/j.jag.2010.04.001.

Lin, H., Chen, M., Lu, G., et al., 2013b. Virtual geographic environments (VGEs): a new generation of geographic analysis tool. Earth Sci. Rev. 126, 74–84. https://doi.org/10.1016/j.earscirev.2013.08.001.

Li, X., Zhou, J., Chen, F., et al., 2018. Cluster risk of walking scenarios based on macroscopic flow model and crowding force analysis. Sustain. Sci. 10 (2), 385. https://doi.org/10.3390/su10020385.

Lü, G., Chen, M., Yuan, L., et al., 2018. Geographic scenario: a possible foundation for further development of virtual geographic environments. Int. J. Digit. Earth 11 (4), 356–368. https://doi.org/10.1080/17538947.2017.1374477.

Pelechano, N., Malkawi, A., 2008. Evacuation simulation models: Challenges in modeling high rise building evacuation with cellular automata approaches. Autom. Constr. 17 (4), 377–385. https://doi.org/10.1016/j.autcon.2007.06.005.

Piccoli, B., Tosin, A., 2011. Time-evolving measures and macroscopic modeling of pedestrian flow. Arch. Ration. Mech. Anal. 199(3), 707-738. doi: 10.1007/s00205-010-0366-y.

Shi, J., Ren, A., Chen, C., 2009. Agent-based evacuation model of large public buildings under fire conditions. Autom. Constr. 18(3), 338-347. doi: 10.1016/j.autcon.2008.09.009.

Song, W., Yu, Y., Fan, W., et al., 2005. A cellular automata evacuation model considering friction and repulsion. Sci. China Mater. 48 (4), 403–413. https://doi.org/10.1360/03ye0486.

Weifeng, Y., Hai, T.K., 2007. A novel algorithm of simulating multi-velocity evacuation based on cellular automata modeling and tenability condition. Physica 379 (1), 250–262. https://doi.org/10.1016/j.physa.2006.12.044.

Wan, J., Sui, J., Yu, H., 2014. Research on evacuation in the subway station in China based on the Combined Social Force Model. Physica A 394, 33–46. https://doi.org/10.1016/j.physa.2013.09.060.

Wei, Y., Thompson, M.P., Belval, E., et al., 2021. Comparing contingency fire containment strategies using simulated random scenarios. Nat. Resour. Model. 34 (1), e12295 https://doi.org/10.1111/nrm.12295.

Van Kerrebroeck, H., Brengman, M., Willems, K., 2017. Escaping the crowd: An experimental study on the impact of a Virtual Reality experience in a shopping mall. Comput. Hum. Behav. 77, 437–450. https://doi.org/10.1016/j.chb.2017.07.019.

Voinov, A., Çöltekin, A., Chen, M., et al., 2018. Virtual geographic environments in socio-environmental modeling: a fancy distraction or a key to communication? Int. J. Digit. Earth 11 (4), 408–419. https://doi.org/10.1080/17538947.2017.1365961.

Wei-Guo, S., Yan-Fei, Y., Bing-Hong, W., et al., 2006. Evacuation behaviors at exit in CA model with force essentials: A comparison with social force model. Physica A 371 (2), 658–666. https://doi.org/10.1016/j.physa.2006.03.027.

Weilian, Li, Jun, Zhu, Lin, Fu, Qing, Zhu, Yakun, Xie, Ya, Hu, 2020. An augmented representation method of debris flow scenes to improve public perception. Int. J. Geogr. Inf. Syst. 35(8), 1521–1544. https://doi.org/10.1080/13658816.2020.1833016.

Wouters, P., Van Nimwegen, C., Van Oostendorp, H., et al., 2013. A meta-analysis of the cognitive and motivational effects of serious games. J. Educ. Psychol. 105 (2), 249. https://doi.org/10.1037/a0031311.

Yang, L.Z., Zhao, D.L., Li, J., et al., 2005. Simulation of the kin behavior in building occupant evacuation based on cellular automaton. Build. Environ. 40 (3), 411–415. https://doi.org/10.1016/j.buildenv.2004.08.005.

Yang, L., Fang, W., Huang, R., et al., 2002. Occupant evacuation model based on cellular automata in fire. Sci. Bull. 47 (17), 1484–1488. https://doi.org/10.1360/02tb9327Y.

Yibin, Ao, Hongying, Zhang, Linchuan, Yang, Yan, Wang, Igor, Martek, Gang, Wang, 2021. Impacts of earthquake knowledge and risk perception on earthquake preparedness of rural residents. Nat. Hazards 107, 1287–1310. https://doi.org/10.1007/s11069-021-04632-w.

Yibin, Ao, Xun, Zhou, Yan, Wang, Linchuan, Yang, Qiongmei, Wang, Igor, Martek, 2020. Flood disaster preparedness: experience and attitude of rural residents in Sichuan, China. Nat. Hazards 104, 2591–2618. https://doi.org/10.1007/s11069-020-04286-0.

Yoo, B., Choi, S.D., 2019. Emergency evacuation plan for hazardous chemicals leakage accidents using GIS-based risk analysis techniques in South Korea. Int. J. Environ. Res. Public Health 16 (11), 1948. https://doi.org/10.3390/ijerph16111948.

Yuan, W., Tan, K.H., 2007. An evacuation model using cellular automata. Phys. A 384 (2), 549–566. https://doi.org/10.1016/j.physa.2007.05.055.

Zanlungo, F., Ikeda, T., Kanda, T., 2011. Social force model with explicit collision prediction. EPL 93 (6), 68005. https://doi.org/10.1209/0295-5075/93/68005.

Zhao, R., Zhai, Y., Qu, L., et al., 2021. A continuous floor field cellular automata model with interaction area for crowd evacuation. Physica A 575, 126049. https://doi.org/10.1016/j.physa.2021.126049.

Zhong, M., Shi, C., Tu, X., et al., 2008. Study of the human evacuation simulation of metro fire safety analysis in China. J. Loss Prev. Process Ind. 21 (3), 287–298. https://doi.org/10.1016/j.jlp.2007.08.001.