Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Shadow obstacle model for realistic corner-turning behavior in crowd simulation*

Gao-qi HE[†1,2,3], Yi JIN[1], Qi CHEN[1], Zhen LIU[4], Wen-hui YUE[2], Xing-jian LU[1]

([1]*Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai 200237, China*)

([2]*MOE Key Laboratory of Geographic Information Science, East China Normal University, Shanghai 200241, China*)

([3]*State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China*)

([4]*College of Information Science and Technology, Ningbo University, Ningbo 315211, China*)

[†]E-mail: hegaoqi@ecust.edu.cn

Received Aug. 6, 2015; Revision accepted Dec. 5, 2015; Crosschecked Dec. 30, 2015

Abstract: This paper describes a novel model known as the shadow obstacle model to generate a realistic corner-turning behavior in crowd simulation. The motivation for this model comes from the observation that people tend to choose a safer route rather than a shorter one when turning a corner. To calculate a safer route, an optimization method is proposed to generate the corner-turning rule that maximizes the viewing range for the agents. By combining psychological and physical forces together, a full crowd simulation framework is established to provide a more realistic crowd simulation. We demonstrate that our model produces a more realistic corner-turning behavior by comparison with real data obtained from the experiments. Finally, we perform parameter analysis to show the believability of our model through a series of experiments.

Key words: Corner-turning behavior, Crowd simulation, Safety awareness, Rule-based model
http://dx.doi.org/10.1631/FITEE.1500253 **CLC number:** TP391

1 Introduction

Human crowd is a fascinating phenomenon in the real world. The simulation of human crowds has recently drawn increasing attention from experts in various types of fields, including computer graphics, social sciences, traffic engineering, architecture, urban planning, and robotics.

The primary challenges of crowd simulation lie in two aspects (Thalmann *et al.*, 2009). One is the realistic simulation of a crowd, while the other is the efficiency of the simulation model. This study focuses mainly on the improvement of realistic crowd simulation.

Several factors need to be considered in realistic crowd simulation, including crowd behavior, individual animation, and scene rendering. Among these factors, the generation of realistic crowd behavior is crucial to crowd simulation. Many approaches have been proposed, such as the social force model (Helbing *et al.*, 2000), the rule-based model (Reynolds, 1999), and reciprocal velocity obstacle (RVO) (van den Berg *et al.*, 2008). However, current simulation models are still inconsistent with the empirical observations because of the complexity of human behaviors. Corner-turning behavior is one of the problematic issues that have not been studied in depth. Although several studies on this topic exist (Watt, 1993; Snook, 2000; Rojas *et al.*, 2013), only a few have considered the effect of human psychology and individual differences in this issue.

* Project supported by the National Natural Science Foundation of China (Nos. 61170318 and 61300133), the Open Research Funding Program of Key Laboratory of Geographic Information Science, China (No. KLGIS2015A05), the Fundamental Research Funds for the Central Universities, China (No. 222201514331), and the Opening Project of Shanghai Key Laboratory of New Drug Design, China (No. 14DZ2272500)

ORCID: Gao-qi HE, http://orcid.org/0000-0001-8365-0970

Hashimoto *et al.* (2013) analyzed human behaviors at corners and reported that when people turn into a corner, they tend to choose a safe route instead of the shortest path. Inspired by the observations in this survey, we present a novel model known as the shadow obstacle (SO) model, which generates a more realistic corner-turning behavior. The proposed model can be easily integrated into current local behavior models. We combine our SO model with a rule-based model in the current study. A corresponding framework is established to conduct a full crowd simulation. The believability of the model is demonstrated through a series of simulations.

2 Related work

Extensive research has been conducted on crowd simulation, and we refer the readers to some of the recent surveys (Pelechano *et al.*, 2008; Thalmann *et al.*, 2009; Zhou *et al.*, 2010). This section highlights some of the most relevant studies on crowd simulation.

2.1 Global navigation

Traditional crowd simulation generally consists of two parts: global navigation and local behavior simulation. The most common methods to deal with global navigation are A* and Dijkstra's algorithms (Cui and Qin, 2010; van Toll *et al.*, 2012). These approaches are accessible but may cause various unexpected problems when integrated with local behavior models. For example, agents may be stuck if they deviate from the planned trajectory. The navigation field method (Patil *et al.*, 2011) uses a vector-based field to direct the crowd movement. Therefore, this method is well suited for local force-based models such as the social force model and the rule-based model. Jin *et al.* (2008) proposed a simple but effective way for authoring crowd scene using radial basis functions (RBFs) based vector fields, which are similar to the navigation field. Furthermore, the way portals method (Curtis *et al.*, 2012) is a great improvement for traditional path planning approaches.

2.2 Local behavior models

The social force model (Helbing *et al.*, 2000) is a well-known model for simulating local crowd behaviors, particularly for a panic crowd. Potential-based force is used to denote the psychological behaviors of crowds, such as collision avoidance. Body collision is explained as a large physical force. However, the individual in this model lacks difference because it treats every individual in the crowd the same. Another fatal issue is that the force for collision avoidance is generated based on only an individual's position. Therefore, the individual's visual information is ignored. Moussaïd *et al.* (2011) proposed a cognitive model to overcome this issue.

The rule-based model (Reynolds, 1999) is another famous model in the field of crowd simulation. In this model, each individual is represented as an autonomous agent with a certain level of cognitive and reasoning capability, such that an individual is able to sense its surroundings and make decisions based on its understanding of the current situation. The behavior of each agent is constrained by a specific rule such as seeking and collision avoidance. Agents can perform various complex patterns of behaviors by combining these rules. The rule-based model is often considered as the most natural model for crowd simulation. The weaknesses of this model lie in the design of the rule and its relatively low computation efficiency. Shao and Terzopoulos (2005) proposed a hierarchical model to improve the performance of the original rule-based model.

RVO (van den Berg *et al.*, 2008) is a newly proposed model, and its original model, i.e., velocity obstacle (Fiorini and Shiller, 1998), is derived from robotics. Rather than being affected by external forces, the agent itself chooses the proper velocity in the next simulation step so as to obtain an obstacle-free path. This is not a human-like behavior since people cannot always predict others' movement or choose a precise path. Therefore, the process of the decision is more like a robot. Based on this model, Guy *et al.* (2009) developed a distributed behavior model to address the problem of real-time collision avoidance in multi-agent systems. Snape *et al.* (2012) discussed the application of RVO in game development.

2.3 Corner-turning behavior

Corner-turning behavior is a specific individual behavior in crowd simulation. Several studies have been conducted on this topic (Watt, 1993; Snook, 2000; Rojas *et al.*, 2013). After obtaining the waypoints by using the global path planning algorithm,

the Bezier curve or B-spline approaches (Watt, 1993) can be used to generate a smooth trajectory. Rojas *et al.* (2013) used some predefined curves to lead individuals when turning a corner. A navigation field (Patil *et al.*, 2011) can also be used as leading information for individuals and is often more flexible than predefined curves. The drawback of all the aforementioned approaches is that they all use a man-made method to generate turning curves, and none of these methods take human psychology and individual differences into account.

2.4 Physical collision

Although physical collision between individuals is usually not important in crowd simulation, a bad physical model will lead to an unnatural phenomenon. In the social force model (Helbing *et al.*, 2000), physical force consists of two parts: 'body force' counteracting body compression and 'sliding friction' impeding relative tangential motion. Kim *et al.* (2013) used a rigid-body system to generate a physical force and proposed a framework to incorporate local behavior with physical force. Considering that rigid-body systems can better present the physical interaction between individuals, we establish a similar framework based on this system in the current study.

3 Shadow obstacle model

Hashimoto *et al.* (2013) showed several observations of human behaviors in corners. The trajectory result is shown in Fig. 1. After analyzing this work we obtain the following three observations:

1. Before turning (area marked as A), people tend to go near the centerline of the corridor.

2. When turning (area B), people will move to the centerline of the corridor again to obtain a wide viewing range.

3. After obtaining a wide viewing range (area C), people will walk inside the shortcut course in the corner even though the risk of collision is high.

Inspired by these findings, we present the shadow obstacle (SO) model to simulate realistic corner-turning behavior in this section. We first discuss the environmental representation of the simulation system. Then the definition of the SO is presented, which is the key component for the whole model. After defining the blind area, a rule-based approach is used to implement the full model.

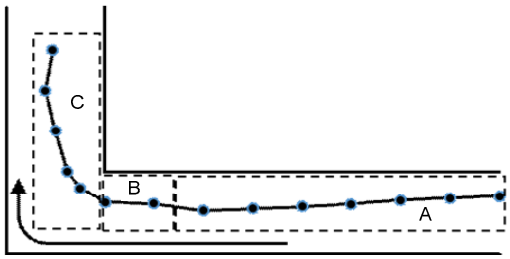

Fig. 1 Test scenario and the trajectory result in Hashimoto *et al.* (2013)

3.1 Environment representation

In our model, agents are constrained to move on a 2D plane. Given that the simulation is conducted in a 3D scenario, we project the boundary of the agents and obstacles onto the 2D plane (Fig. 2).

Each agent in the current model is represented as an open disk centered at position $\boldsymbol{P}$ with a radius R denoting the collision range. To generate individual differences, each agent is given a safety awareness factor δ that belongs to [0, 1]. Agents with high safety awareness will have large δ values. Obstacles are simplified as polygons (often as squares) in a 2D plane.

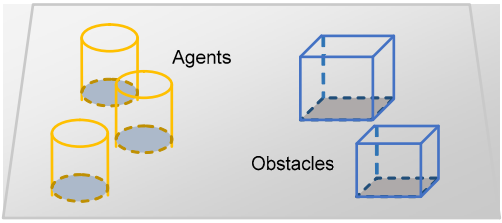

Fig. 2 Transformation of the 3D agents and obstacles onto the 2D plane

3.2 Shadow obstacle

To simulate the corner-turning behavior observed by Hashimoto *et al.* (2013), we place an intangible but effective SO on every turning point of the environment (Fig. 3). The corner in our study is defined as a 'convex corner', which is composed of two straight walls. The corresponding turning point is described as the intersection point of the two walls.

We can place the SO either manually, which is more accurate, or automatically, according to the location of the walls. The SO itself is only an assistive tool for the simulation and the setting of the SO is completed before the whole simulation; therefore, it will not affect the efficiency of the simulation.

Actually, the SO can have various shapes. For simplicity, we represent the SO as a circle (Fig. 3). In this case, each SO has four parameters: the range of influence (R^{SO}), which is the radius of the circle; the center position ($\boldsymbol{C}^{SO}$), which is the corresponding corner point; the inner angle of the corner (Ang^{SO}); and an extra vector ($\mathbf{Ex}^{SO}$), which is a vector that divides the corner into two parts. Parameters $\boldsymbol{C}^{SO}$, Ang^{SO}, and $\mathbf{Ex}^{SO}$ are used to describe the static information of the corner, and R^{SO} is related to agents' safety awareness factor δ, which will be discussed in detail in Section 5.5.

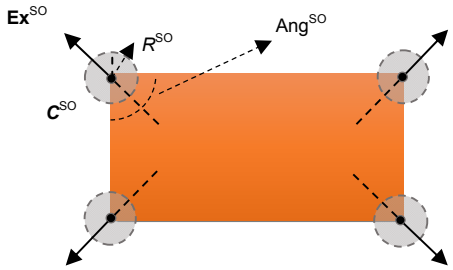

Fig. 3 Setting of the shadow obstacle

3.3 Definition of the blind area

When turning the corner, pedestrians usually anticipate to enlarge their viewing range. As shown in Fig. 4a, the viewing range itself is not determined by only a certain corner. In fact, enlarging the viewing range at a certain corner means that the pedestrian will decrease the area that cannot be seen. This area is defined as the 'blind area' in our study, which, at a certain corner, is denoted as the area between agents' sight and wall in the dark side (Fig. 4b).

Suppose agents are trying to turn corner k with one SO denoted as SO_k. Because the predefined extra vector of SO_k exactly divides the whole corner area into two equal parts, we are able to consider the corner-turning behavior from one side.

The magnitude of the blind area at corner k for agent i is quantified by angle α (Fig. 4b). Once the agent, such as agent j, obtains a full viewing range of the current corner, the blind area will disappear. Thus, we can measure the value of the blind area $\mathrm{BA}_i(\mathrm{SO}_k)$ for agent i at corner k by the following equations:

$$\mathrm{BA}_i(\mathrm{SO}_k)=\begin{cases}\alpha, & \alpha\geq 0,\\ 0, & \text{otherwise},\end{cases} \tag{1}$$

$$\alpha=\pi-\beta-\gamma, \tag{2}$$

$$\beta=\mathrm{Ang}_k^{SO}/2, \tag{3}$$

$$\gamma=\arccos\frac{(\boldsymbol{C}_k^{SO}-\boldsymbol{P}_i)\cdot\mathbf{Ex}_k^{SO}}{\left|(\boldsymbol{C}_k^{SO}-\boldsymbol{P}_i)\cdot\mathbf{Ex}_k^{SO}\right|}, \tag{4}$$

where Ang_k^{SO}, $\boldsymbol{C}_k^{SO}$, and $\mathbf{Ex}_k^{SO}$ are the three parameters for SO_k, as defined in Section 3.2, and $\boldsymbol{P}_i$ is the current position of agent i.

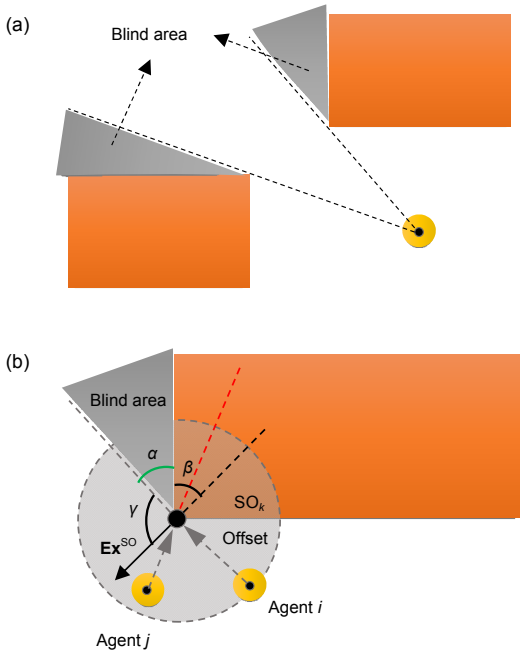

Fig. 4 Illustration of the blind area
(a) Definition of the blind area; (b) Computation of the blind area at corner k

From Observation (3) mentioned before, we know that once agents obtain a large enough viewing range, they would choose to walk in a shortcut course in the corner. Accordingly, we define an endurable blind area (EBA) for every agent. If the current blind area is smaller than EBA, the agent will ignore the influence of the SO.

3.4 Corner-turning rule

Our model can be easily integrated into current local behavior models. In this study, we choose the rule-based model as our underlying model, since it is always considered as the most natural approach for crowd simulation and also matches our model very well.

The first task is to check whether the agent is turning the corner or not. This is rather difficult, because in most of the crowd simulation systems, the agent just passively follows the route information to reach the destination. In this study, we use an approximate method to deal with this issue.

Suppose at a certain time t, the position of agent i is $\boldsymbol{P}_i$ and the velocity is $\boldsymbol{V}_i$. The following four conditions should be satisfied if agent i is currently turning corner k:

1. The agent is within the influence range of SO_k, which is to say $|\boldsymbol{P}_i-\boldsymbol{C}_k^{\mathrm{SO}}|<R_k^{\mathrm{SO}}$.

2. $\mathrm{BA}_i(\mathrm{SO}_k)>0$.

3. The agent is currently walking toward the corner.

4. The agent is currently walking toward its goal.

As shown in Fig. 5, supposing the agent will have a clockwise turn, we consider that only the agent in Case 1 is currently turning the corner. Conditions 1–3 will restrict the agent's position into a red zone (Fig. 5), and hence the agent in Case 2 will not be considered as a turning behavior. According to Condition 4, the agent in Case 3 is also not considered as a turning behavior. The specific form of Conditions 3 and 4 depends highly on the underlying global navigation strategy. For example, if a waypoint system is used and the current goal position of agent i is $\boldsymbol{G}_i$, Condition 4 can be easily translated to $(\boldsymbol{P}_i-\boldsymbol{G}_i)\boldsymbol{V}_i>0$.

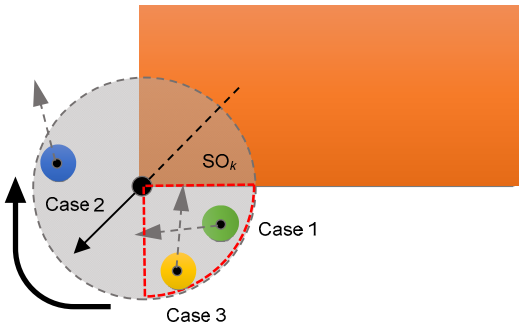

Fig. 5 Three cases of agents turning a corner (references to color refer to the online version of this figure)

Condition 3 can be defined with the help of the previous goal position $\boldsymbol{G}_i^{\mathrm{pre}}$ and therefore translated into $\left|\boldsymbol{G}_i-\boldsymbol{C}_k^{\mathrm{SO}}\right|<\left|\boldsymbol{G}_i^{\mathrm{pre}}-\boldsymbol{C}_k^{\mathrm{SO}}\right|$.

The observations by Hashimoto *et al.* (2013) show that people have a tendency to enlarge their viewing range when turning a corner. From a psychological perspective, this behavior can be explained as human safety awareness because with a larger viewing range, people can detect more potential collisions and avoid them earlier autonomously. Thus, the key for the corner-turning rule is to enlarge the viewing range, which implies decreasing the blind area.

Supposing at a certain time t, the position of agent i is $\boldsymbol{P}_i$, the velocity is $\boldsymbol{V}_i$, and the agent is currently turning corner k. If the agent has obtained a large enough viewing range, which is to say $\mathrm{BA}_i(\mathrm{SO}_k)<\mathrm{EBA}_i$, the agent will not be affected by the SO. Therefore, the force is set to zero. Otherwise, we have the following two assumptions:

1. In the next Δt time, agent i will maintain a constant speed $\mathrm{SP}_{\mathrm{const}}$.

2. Agent i is not able to obtain a full viewing range of corner k after the next Δt time.

These two assumptions will be asymptotically satisfied if Δt tends to zero. According to Assumption 1, we can understand that all reachable areas RA_i for agent i in the next Δt time are a circular area (Fig. 6). To obtain the largest viewing range after the next Δt time, the best direction of motion for agent i should be obtained by dealing with the following optimization problem:

$$\frac{\boldsymbol{V}^{\mathrm{best}}}{\left|\boldsymbol{V}^{\mathrm{best}}\right|}=\boldsymbol{P}^{\mathrm{best}}-\boldsymbol{P}_i, \tag{5}$$

$$\boldsymbol{P}^{\mathrm{best}}=\arg\min_{\boldsymbol{P}\in\mathrm{RA}_i}(\alpha^{\mathrm{new}}), \tag{6}$$

$$\begin{aligned}&\arg\min_{\boldsymbol{P}\in\mathrm{RA}_i}(\alpha^{\mathrm{new}})=\arg\min_{\boldsymbol{P}\in\mathrm{RA}_i}(\pi-\beta-\gamma^{\mathrm{new}})\\&=\arg\max_{\boldsymbol{P}\in\mathrm{RA}_i}\gamma^{\mathrm{new}}=\arg\max_{\boldsymbol{P}\in\mathrm{RA}_i}\frac{(\boldsymbol{C}_k^{\mathrm{SO}}-\boldsymbol{P})\cdot\mathbf{Ex}_k^{\mathrm{SO}}}{\left|(\boldsymbol{C}_k^{\mathrm{SO}}-\boldsymbol{P})\cdot\mathbf{Ex}_k^{\mathrm{SO}}\right|}.\end{aligned} \tag{7}$$

This optimization problem can be simply solved by a graphical solution shown in Fig. 6. Therefore, $\boldsymbol{P}^{\mathrm{best}}$ should be the tangent point between RA_i and tangent line T_i.

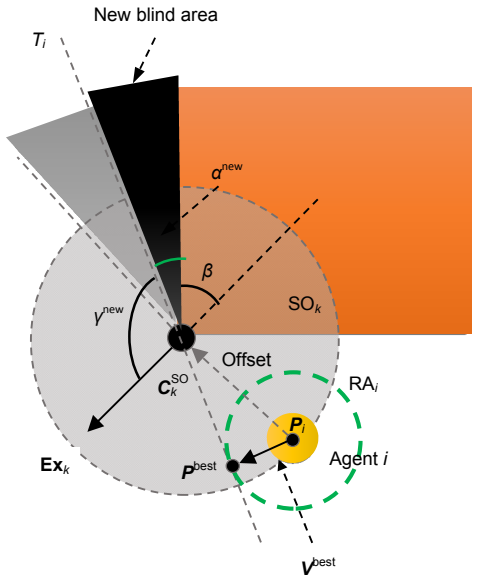

Fig. 6 Direction of the force for generating the largest viewing range after the next Δt time

We then assume that Δt tends to be zero. The reason why we can do this is the continuity of human thinking. Therefore, we can consider that humans will make their decisions continuously, which implies that the interval between the two decisions (Δt) is close to zero. Consequently, tangent line T_i will overlap with the offset vector shown in Fig. 6. Therefore, the computation of the best direction can be simplified as follows:

$$\frac{\boldsymbol{V}^{\text{best}}}{|\boldsymbol{V}^{\text{best}}|}=(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}^{*}, \tag{8}$$

where $(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}$ denotes a normalized vector that is perpendicular to $\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i$, and $(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}^{*}$ denotes one of the two $(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}$ vectors that can enlarge the current viewing range of agent i. It can be computed with the help of the extra vector of SO:

$$(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}^{*}=\begin{cases}(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}, & (\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}\cdot\mathbf{Ex}_k^{\text{SO}}>0,\\ -(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}, & \text{otherwise}.\end{cases} \tag{9}$$

After obtaining the best velocity direction, we can generate the corner-turning force based on the social force model:

$$\boldsymbol{F}_{\text{Corner}}=\frac{\boldsymbol{V}^{\text{best}}-\boldsymbol{V}_i}{\tau^{\text{COR}}}=\frac{(\boldsymbol{C}_k^{\text{SO}}-\boldsymbol{P}_i)_{\perp}^{*}\cdot\text{SP}_{\text{des}}-\boldsymbol{V}_i}{\tau^{\text{COR}}}, \tag{10}$$

where SP_{des} denotes the desired speed for the agent, and τ^{COR} denotes the time of relaxation at the corner.

3.5 Time complexity of the shadow obstacle model

For our SO model, the time complexity is $O(N)$, where N is the number of pedestrians in the scene. This is because a pedestrian will be affected only when he/she is in the influence range of a certain SO. Thus, a pedestrian will be affected only by a limited SO (often one or two) in a certain time.

4 Full crowd simulation

In this section, a full crowd simulation framework is established. The framework combines global navigation, local behavior, and physical collision. Fig. 7 provides an overview of the full simulation system.

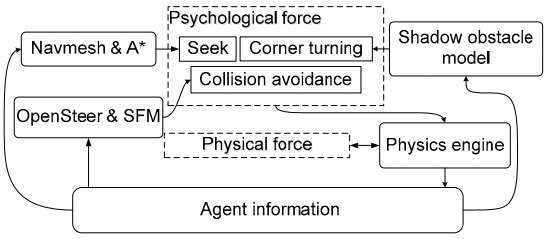

Fig. 7 An overview of the crowd simulation system

4.1 Global navigation strategy

For simplicity and generality, we choose the A* algorithm with navigation mesh (navmesh) as our global navigation strategy. Given a start position and a target position, we can obtain a series of waypoints $\{\boldsymbol{G}_k|k=0, 1, \ldots\}$. Given that the current seeking waypoint is $\boldsymbol{G}_i$, based on the social force model, the corresponding seeking force is defined as follows:

$$\boldsymbol{F}_{\text{seek}}=\frac{\text{Norm}(\boldsymbol{G}_i-\boldsymbol{P}_i)\cdot\text{SP}_{\text{des}}-\boldsymbol{V}_i}{\tau^{\text{seek}}}, \tag{11}$$

$$\text{Norm}(\boldsymbol{G}_i-\boldsymbol{P}_i)=\frac{\boldsymbol{G}_i-\boldsymbol{P}_i}{|\boldsymbol{G}_i-\boldsymbol{P}_i|}. \tag{12}$$

4.2 Collision avoidance

For agent–agent collision avoidance, a line-of-sight test is first used to remove the influence of agents who are out of the viewing range. Then the force for collision avoidance based on the OpenSteer and social force model is generated. The direction of the force is exactly the same as the direction in the OpenSteer. For the generation of a smooth trajectory, the magnitude of the force is slightly changed as follows:

$$|\boldsymbol{F}_{\mathrm{AA}}| = A\exp\left(\frac{r_{ij}-\mathrm{dist}(i,j)}{B}\right), \tag{13}$$

where $r_{ij}=r_i+r_j$ denotes the total radius of agents i and j, $\mathrm{dist}(i, j)$ denotes the distance between agents i and j, and A and B are two constant values. Considering that the total radius is always less than or equal to the distance, $|\boldsymbol{F}_{\mathrm{AA}}|$ should belong to $(0, A)$.

For agent–obstacle collision avoidance, if the 2D presentation of an obstacle is a circle, the force can be generated as done in agent–agent collision avoidance with one of the agents having zero velocity. If the 2D presentation of an obstacle is a polygon, then the presentation is further decomposed into several line segments and the social force model is used to compute the corresponding force.

4.3 Physical collision

To obtain a realistic crowd behavior, a rigid-body system is used to generate the physical force between agents during collision. Although the psychological forces are generated in a 2D space, the rigid-body system uses the exact 3D representation of the environment for collision detection. In the simulation scenario, capsule colliders and box colliders are used for the 3D representation of the agents and obstacles, respectively.

5 Experiments and results

In this section, the proposed SO model was tested under various situations in a specific scene (Fig. 8). The framework established in Section 4 was used for simulation. The whole model was implemented in C# and the Unity3d Game Engine was used for visualization. The entity models used in the scene can be found in the asset store of Unity3d.

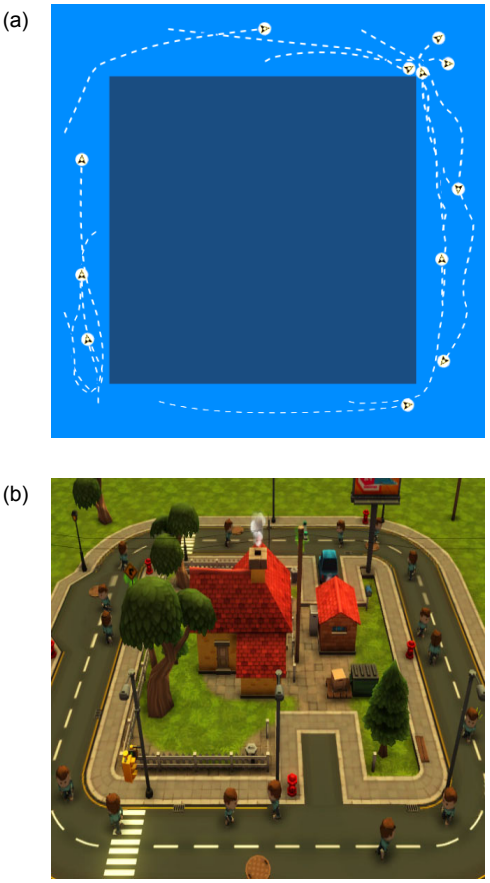

Fig. 8 The 2D (a) and 3D (b) representations of the scene

5.1 Comparison with data from the recent survey

To demonstrate the believability of the SO model, we compared the trajectories computed by our algorithm with the trajectories shown in Hashimoto *et al.* (2013).

The setting of the scenario was exactly the same as that in Hashimoto *et al.* (2013). Once the experiment started, an agent was placed on the start position. The start position was located at the right side of the corridor where the individual's view against the corner was bad. The agent was given a clockwise navigation field such that it walked toward the left corner and then turned right at the corner autonomously. To match the real data as well as possible, we tested a large number of cases with different data settings, and the most matched parameters were set as follows:

A=1, B=0.75, τ^{seek}=1,
EBA=20, τ^{COR}=1, R^{SO}=2.

As shown in Fig. 9, the result matched closely with the trajectory result in Hashimoto *et al.* (2013). Since the parameters were chosen manually, the trajectory itself may not be very convincing. Here, we note that the real data (trajectory) can also be different in different situations. Even the same experiment may generate different results because of the complexity and unpredictability of human behaviors.

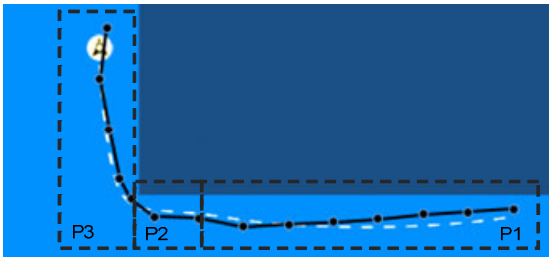

Fig. 9 Comparison of the trajectory reported in Hashimoto *et al.* (2013) (black) with the trajectory generated by the SO model (white)

However, the corner-turning behavior patterns actually exist. To show that our model can generate these patterns, the total simulation process is divided into three phases.

In the first phase, the agent walks toward the left corner with the help of the navigation force. The agent then went closer to the centerline of the corridor that exactly meets Observation 1 in Hashimoto *et al.* (2013). This event is caused by the force for avoiding the wall. Given that the magnitude of the force is inversely proportional to the distance between the agent and the wall, the agent will always keep a certain distance from the wall. If two walls exist on both sides of the agent, the agent will be passively pushed to the centerline of the corridor.

In the second phase, the agent tries to turn right at the corner. In this phase, a force generated by the SO is exerted on the agent. To enlarge its viewing range, the agent passively gets close to the centerline of the corridor again. Here, we note that getting close to the centerline of the corridor is only an observation in a specific scene. We conclude that getting a wider viewing range is the cause of this phenomenon. Therefore, if the corridor is rather narrow, it is possible that people will go far away from the centerline to get a wider viewing range.

In the third phase, the agent has a large enough viewing range and thus is not affected by the SO. The navigation force becomes the leading force such that the agent chooses the inner course even though the risk is still high.

5.2 Comparisons with Rojas's method

In this section, we compared our model with Rojas's method (Fig. 10). In Rojas's method, predefined curves are used to lead individuals when turning a corner. Therefore, if the position and goal of two agents are the same, the trajectories of the two agents would also be the same. Furthermore, agents' trajectories will be limited to those predefined curves when turning the corner. Our method takes human psychology and individual differences into account and therefore can generate a more realistic corner-turning behavior. The trajectories generated by our method are diverse and more natural.

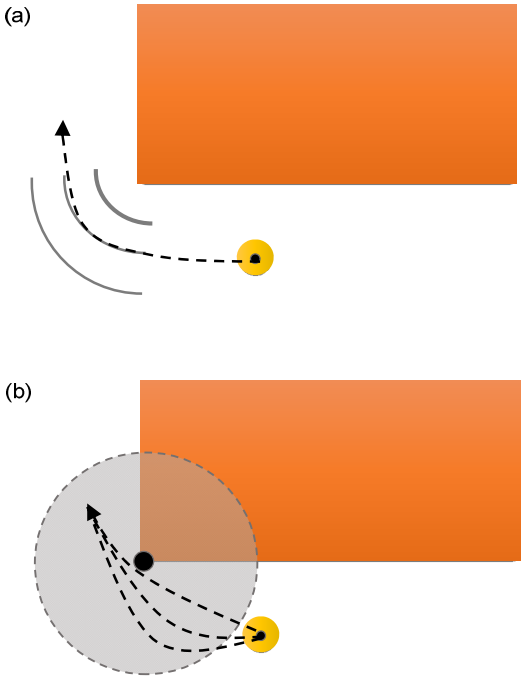

Fig. 10 Comparison of Rojas's method (a) with our method (b)

5.3 Influence of corner-turning behavior

To analyze the influence of the force on the corner-turning behavior, three experiments were

designed using the following combinations of the psychological forces discussed in Section 4:

1. seek;
2. seek+obstacle avoidance;
3. seek+obstacle avoidance+corner turning.

In each case, both the agent's trajectory and variation of the agent's viewing range are recorded through time.

In Fig. 11b, the blind area of the agent in Case 3 decreased drastically from 9 s because of the influence of the corner-turning force. This result matched well with the corner-turning rule designed in Section 3.3. However, the variation speed of the blind area was slowed down later. Eventually, the agent in Cases 1 and 2 received a full viewing range earlier than the agent in Case 3. The main reason for this phenomenon is the side effect of the corner-turning force (Fig. 11a). To enlarge the viewing range, the agents will strive to maintain a distance from the corner. However, agents who are not affected by the SO will continue to shorten the distance between them and the corner. Considering that an agent that is far away from a corner will have to move a considerable distance to enlarge the viewing range by an identical degree, the variation speed of the blind area is passively slowed down because of the relatively long distance.

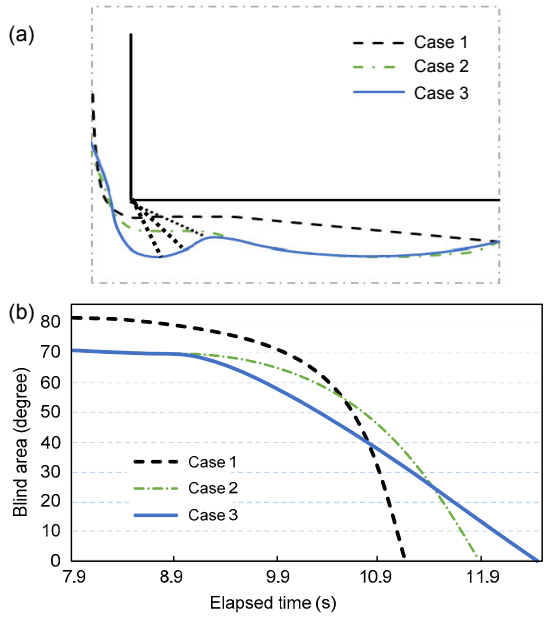

Fig. 11 Trajectories of the agent with different combinations of the psychological forces (a) and the variation of the blind area through time in the three cases (b)

5.4 Parameter analysis of the shadow obstacle model

The influence range of the SO (R^{SO}), endurable blind area of the agents (EBA), and relaxation time in corner (τ^{COR}) are the three dominating factors in the SO model. R^{SO} and EBA affect the duration time of the corner-turning force, whereas τ^{COR} determines the magnitude of the according force.

By changing the values of the three parameters, several experiments were implemented. The variations of the blind area through time were measured in each experiment.

Based on the results shown in Figs. 12–14, we find that a larger R^{SO} leads to an earlier influence of the SO, and the growth of the influence is

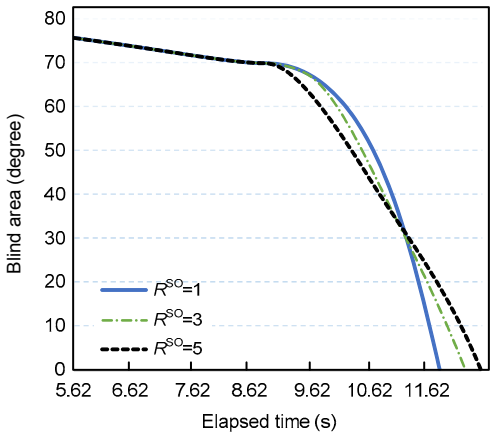

Fig. 12 Variation of the blind area through time with different R^{SO} (τ^{COR} and EBA are fixed at 1 s and 45°, respectively)

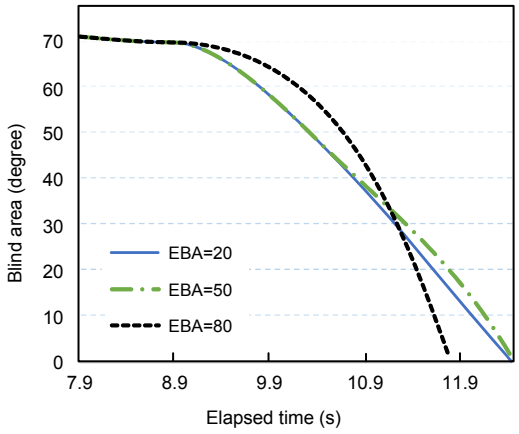

Fig. 13 Variation of the blind area through time with different endurable blind area (τ^{COR} and R^{SO} are fixed at 1 s and 5 m, respectively)

proportional to the growth of R^{SO}. A smaller EBA will extend the influence, but this effect is covered when EBA decreases to a certain value because of the waypoint setting and other disturbances. Parameter τ^{COR} enlarges the impact of the SO when it becomes smaller. Furthermore, the slow-down effect found in Section 5.3 is observed in Figs. 12–14.

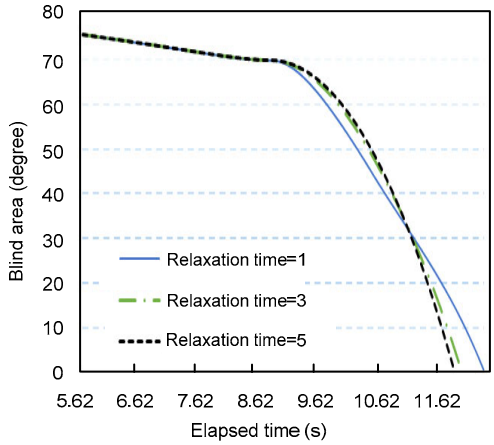

Fig. 14 Variation of the blind area through time with different τ^{COR} (EBA and R^{SO} are fixed at 45° and 5 m, respectively)

5.5 Individual differences

We first established the relationship between the safety awareness factor (δ) and the three dominating factors discussed in Section 5.4. Suppose that people with higher safety awareness will obtain greater influence from the SO. Therefore, the relationships are established as follows:

$$R^{SO}=10\delta, \quad (14)$$

$$\text{EBA}=(180-\text{Ang}^{SO})(1-\delta), \quad (15)$$

$$\tau^{COR}=0.5+5(1-\delta). \quad (16)$$

The safety awareness is then divided into three levels:

1. High level: δ=1;
2. Medium level: δ=0.5;
3. Low level: δ=0.

A corner-collision experiment was conducted based on the scenario shown in Fig. 15. Two agents walked toward the same corner from different directions. The force for avoiding the obstacles was slightly reduced to highlight the contribution of the corner-turning behavior.

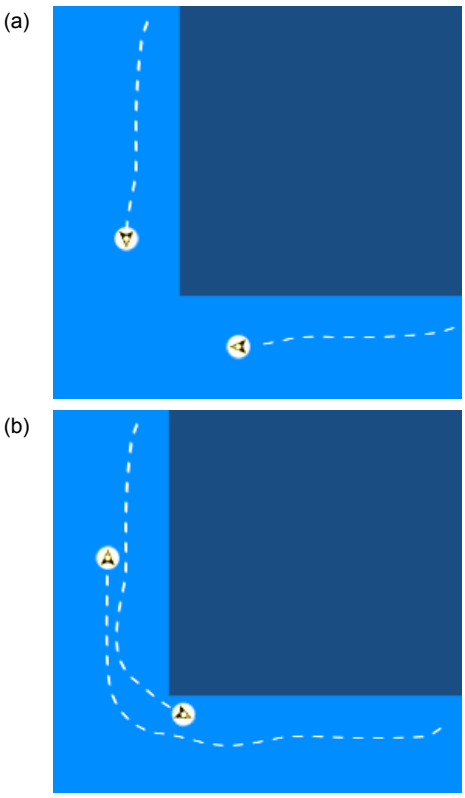

Fig. 15 Snapshots of the corner-collision experiment
(a) Before interaction; (b) After interaction

Various cases were tested by changing the safety awareness factor of the two agents. In each case, the average speed of the two agents and the distance between them when they first saw each other were recorded. The results are shown in Fig. 16.

The results show that when the safety awareness of the two agents increased, the average speed of the two agents slightly decreased and the distance between them increased in most of the cases. The lower speed was caused by its frequent variation, while the larger distance was due to the side effect of the corner-turning force discussed in Section 5.3. These two results exactly conform to the definition of safety awareness because either decreasing the speed or enlarging the distance will give agents more time to deal with the collision. Therefore, agents with higher safety awareness levels will have lower possibilities of colliding with others in the corner.

5.6 Generation of shadow obstacles in a complex scene

Assume that a single obstacle's bounding box is a cube in our scene. The complex polygonal obstacles

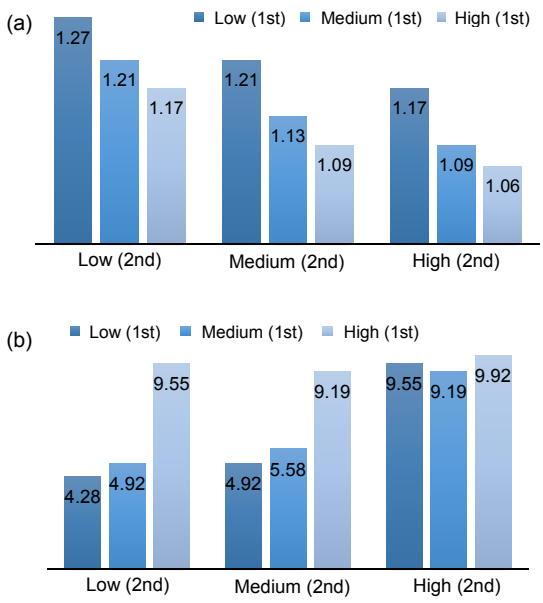

Fig. 16 Average speed of the two agents (a) and the distance between the two agents (b)

are formed with many obstacles overlapping with each other (Fig. 17). With no overlapping, we can automatically generate the SOs using the vertex information. Each SO (specifically, the center position) is set on the vertex. However, when obstacles overlap with each other, some vertices are no longer turning points and therefore we should eliminate them. In this study, we use a ray intersection test to eliminate those useless vertices. The test was implemented with unity's physics and layer system (Fig. 18a). The final result is shown in Fig. 18b.

6 Conclusions

This study presents a novel model, the shadow obstacle model, to generate the realistic corner-turning behavior in crowd simulation. The model was inspired by a recent survey that reported several observations of the safety-aware human behavior at a corner. Based on these observations, we implemented our model using a rule-based approach, and a corresponding framework was established to perform the full crowd simulation. We demonstrated the believability of our model through a series of simulations, including the comparisons with the real data and the parameter analysis of the proposed model.

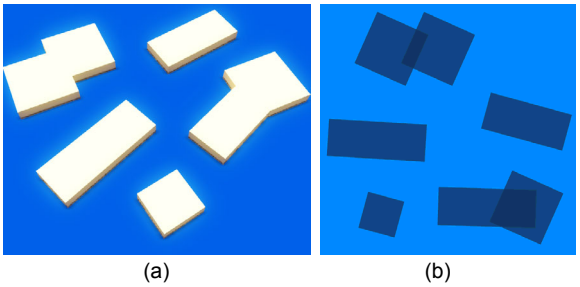

Fig. 17 The 3D (a) and 2D (b) representations of the complex scene

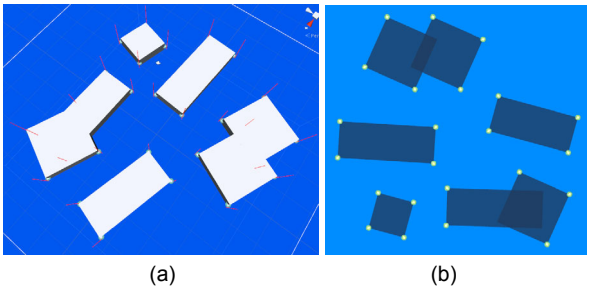

Fig. 18 Ray intersection test (a) and the final result of the generation of shadow obstacles (b)

Future research on this topic will include parameter tuning of the proposed model and the combination of different behaviors. The parameters in the current model are set manually and thus lack accuracy. The whole model may be restricted to some specific scenarios. Therefore, it is necessary to tune the parameters according to real data and experimental observations. In addition, the current model can be improved by adding different behaviors. To generate a more realistic simulation environment, Industry Foundation Classes (IFC) is a convenient tool to make various shapes of buildings, and a semantically enriched 3D environment is interesting for intelligent agents (Béhé *et al.*, 2014).

References

Béhé, F., Galland, S., Gaud, N., *et al.*, 2014. An ontology-based metamodel for multiagent-based simulations. *Simul. Model. Pract. Theory*, **40**:64-85. http://dx.doi.org/10.1016/j.simpat.2013.09.002

Cui, Y., Qin, G., 2010. Intelligent path planning in 3D scene. Proc. Int. Conf. on Computer Application and System Modeling, p.579-583. http://dx.doi.org/10.1109/ICCASM.2010.5620400

Curtis, S., Snape, J., Manocha, D., 2012. Way portals: efficient multi-agent navigation with line-segment goals. Proc. ACM SIGGRAPH Symp. on Interactive 3D Graphics and Games, p.15-22.

http://dx.doi.org/10.1145/2159616.2159619

Fiorini, P., Shiller, Z., 1998. Motion planning in dynamic environments using velocity obstacles. *Int. J. Robot. Res.*, **17**(7):760-772. http://dx.doi.org/10.1177/027836499801700706

Guy, S.J., Chhugani, J., Kim, C., *et al.*, 2009. ClearPath: highly parallel collision avoidance for multi-agent simulation. Proc. ACM SIGGRAPH/Eurographics Symp. on Computer Animation, p.177-187. http://dx.doi.org/10.1145/1599470.1599494

Hashimoto, K., Yoshimi, T., Mizukawa, M., *et al.*, 2013. A study of collision avoidance between service robot and human at corner—analysis of human behavior at corner. Proc. 10th Int. Conf. on Ubiquitous Robots and Ambient Intelligence, p.383-384. http://dx.doi.org/10.1109/URAI.2013.6677293

Helbing, D., Farkas, I., Vicsek, T., 2000. Simulating dynamical features of escape panic. *Nature*, **407**(6803):487-490. http://dx.doi.org/10.1038/35035023

Jin, X., Xu, J., Wang, C.L., *et al.*, 2008. Interactive control of large-crowd navigation in virtual environments using vector fields. *IEEE Comput. Graph. Appl.*, **28**(6):37-46. http://dx.doi.org/10.1109/MCG.2008.117

Kim, S., Guy, S.J., Manocha, D., 2013. Velocity-based modeling of physical interactions in multi-agent simulations. Proc. 12th ACM SIGGRAPH/Eurographics Symp. on Computer Animation, p.125-133. http://dx.doi.org/10.1145/2485895.2485910

Moussaïd, M., Helbing, D., Theraulaz, G., 2011. How simple rules determine pedestrian behavior and crowd disasters. *PNAS*, **108**(17):6884-6888. http://dx.doi.org/10.1073/pnas.1016507108

Patil, S., van den Berg, J., Curtis, S., *et al.*, 2011. Directing crowd simulations using navigation fields. *IEEE Trans. Visual. Comput. Graph.*, **17**(2):244-254. http://dx.doi.org/10.1109/TVCG.2010.33

Pelechano, N., Allbeck, J.M., Badler, N.I., 2008. Virtual Crowds: Methods, Simulation, and Control. Morgan & Claypool Publishers, USA. http://dx.doi.org/10.2200/s00123ed1v01y200808cgr008

Reynolds, C.W., 1999. Steering behaviors for autonomous characters. Game Developers Conf., p.763-782.

Rojas, F.A., Park, J.H., Yang, H.S., 2013. Group agent-based steering for the realistic corner turning and group movement of pedestrians in a crowd simulation. Proc. Computer Animation and Social Agents, p.1-4.

Shao, W., Terzopoulos, D., 2005. Autonomous pedestrians. Proc. ACM SIGGRAPH/Eurographics Symp. on Computer Animation, p.19-28. http://dx.doi.org/10.1145/1073368.1073371

Snape, J., Guy, S.J., Lin, M.C., *et al.*, 2012. Reciprocal collision avoidance and multi-agent navigation for video games. Workshops at the 26th AAAI Conf. on Artificial Intelligence, p.49-52.

Snook, G., 2000. Simplified 3D movement and pathfinding using navigation meshes. *Game Program. Gems*, **1**:288-304.

Thalmann, D., Grillon, H., Maim, J., *et al.*, 2009. Challenges in crowd simulation. Proc. Int. Conf. on CyberWorlds, p.1-12. http://dx.doi.org/10.1109/CW.2009.23

van den Berg, J., Lin, M., Manocha, D., 2008. Reciprocal velocity obstacles for real-time multi-agent navigation. Proc. IEEE Int. Conf. on Robotics and Automation, p.1928-1935. http://dx.doi.org/10.1109/ROBOT.2008.4543489

van Toll, W.G., Cook, A.F., Geraerts, R., 2012. Real-time density-based crowd simulation. *Comput. Animat. Virt. Worlds*, **23**(1):59-69. http://dx.doi.org/10.1002/cav.1424

Watt, A., 1993. 3D Computer Graphics. Addison-Wesley, UK.

Zhou, S., Chen, D., Cai, W., *et al.*, 2010. Crowd modeling and simulation technologies. *ACM Trans. Model. Comput. Simul.*, **20**(4):20.1-20.35. http://dx.doi.org/10.1145/1842722.1842725