Bridging the Gap between Visual Exploration and Agent-Based Pedestrian Simulation in a Virtual Environment

Martin Brunnhuber
VRVis Research Center
Donau-City-Strasse 1
Vienna, Austria
brunn@vrvis.at

Helmut Schrom-Feiertag
Austrian Institute of Technology
Giefinggasse 2
Vienna, Austria
helmut.schrom-feiertag@ait.ac.at

Christian Luksch
VRVis Research Center
Donau-City-Strasse 1
Vienna, Austria
luksch@vrvis.at

Thomas Matyus
Austrian Institute of Technology
Giefinggasse 2
Vienna, Austria
thomas.matyus@ait.ac.at

Gerd Hesina
VRVis Research Center
Donau-City-Strasse 1
Vienna, Austria
hesina@vrvis.at

ABSTRACT

We present a system to evaluate and improve visual guidance systems and signage for pedestrians inside large buildings. Given a 3D model of an actual building we perform agent-based simulations mimicking the decision making process and navigation patterns of pedestrians trying to find their way to predefined locations. Our main contribution is to enable agents to base their decisions on realistic three-dimensional visibility and occlusion cues computed from the actual building geometry with added semantic annotations (e.g. meaning of signs, or purpose of inventory), as well as an interactive visualization of simulated movement trajectories and accompanying visibility data tied to the underlying 3D model. This enables users of the system to quickly pinpoint and solve problems within the simulation by watching, exploring and understanding emergent behavior inside the building. This insight gained from introspection can in turn inform planning and thus improve the effectiveness of guidance systems.

Categories and Subject Descriptors

I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism—*Visible line/surface algorithms*; I.6.7 [**Simulation and Modeling**]: Simulation Support Systems—*Environments*

VRST'12, December 10–12, 2012, Toronto, Ontario, Canada.

Keywords

Human perception and performance, perceptual validation, virtual environments, pedestrian simulation, wayfinding

1. INTRODUCTION

Agent-based pedestrian simulations in virtual environments have recently become a powerful tool for planning and improving barrier-free environments in public buildings due to advances in both algorithms and available hardware solutions (see, e.g. [3]). Unfortunately, most of these simulation systems rely on a very simplified abstraction of the most active human sense during a way-finding process: the visual system. While this simplification may be sufficient to simulate an "average" person, the limitations associated with such approaches affect the proper simulation of handicapped or impaired people, whose visual scene perception and movement behavior differs significantly – although barrier-free access is of highest importance for them.

Another limiting aspect in this research field is the difficult analysis and interpretation of the simulation results: Due to the abstraction of the scene (e.g. to keep the simulation complexity within reasonable limits), usually only simple 2D visualizations are provided, making it hard to understand behavior and decision-making of an agent visually, and therefore leaving the interpretation of simulation results to an experienced expert in this field.

We propose a novel way to increase the importance of visual aspects in the field of agent-based pedestrian simulation by combining it with techniques from the area of Computer Graphics. By applying strategies for visibility calculations in semantically annotated 3D scenes as an additional input to the simulation system, autonomous way-finding and decision-making can be brought to a new level by taking plausible human visual perception into account. Moreover, we show how to visualize both simulation and visibility data in an interactive 3D environment, making it possible to easily understand the behavior of the simulated agent(s), verify the correctness and quality of the simulation, improve the simulation parameters through the gained insights, and simplify the planning and improvement of barrier-free buildings.

The application of our system is demonstrated in a real world example: The railway station “Vienna North” in Austria has been virtually reconstructed and semantically tagged, and used in a simulation scenario with both a walking person and a wheel chair driver as virtual agents.

The main contributions in our presented approach are therefore based on the interdisciplinarity prevalent in the system:

- We propose a 3D visibility evaluation method based on hardware-accelerated real-time techniques, that extends the typical 2D pedestrian simulation by introducing a plausible three-dimensional model for human visual perception as additional input for simulated agents.
- Together with semantic annotations in the 3D scene, this improved visual perception system supports decision-making in the pedestrian simulation, enabling a more realistic human movement behavior than in a simplified 2D abstraction.
- Finally, a real-time 3D visualization of simulated trajectory and visibility data helps to understand behavior and decisions of virtual agents and to verify and improve the simulation parameters and quality.

2. RELATED WORK

While the usage of computer graphics hardware to support decisions made by virtual agents in a person path simulation in a 3D environment *visually* is a new concept, there are still various publications and techniques related to our presented technique.

2.1 Visibility

Calculation of visibility is a major research topic in computer graphics as it is on the one hand needed for 3D applications to run efficiently and on the other hand to compute various visual effects (e.g. shadows). Especially culling techniques such as *frustum or occlusion culling* as described by Akenine-Moeller et al. [1] are widely used in 3D applications. An overview of visibility algorithms for walkthrough applications is given by Cohen-Or et al. [6]: They show various examples of visibility tests in computer graphics and indicate that occlusion culling and shadow algorithms have a lot in common – a fact that is heavily exploited in our work.

Ray tests are a further well-known technique for visibility calculation in computer graphics: An aggressive approach which is shown by Wonka et al. [21], where guided rays are used to detect visible triangles with a low amount of errors. A related approach to finding visibility of regions inside public buildings was presented by Brunnhuber et al. [5], where rays are used to mark visible regions of a complete crowd over a period of time. Their method requires a long processing time and produces only an approximative visualization of visibility. The view blocking aspects of other persons do not receive any attention in their work, but the views of every simulated person are accumulated into a so-called visibility map in order to point out well-noticed parts of the environment.

Shadows are essential for a realistic look of a rendered image. In [9], a recent state-of-the-art overview on techniques and research in real-time shadows is given. Shadow mapping, a widely used real-time shadow technique, was first presented 1978 by Williams [20] and consecutively refined in terms of quality and hardware adoptions. The basic idea is to first store a discretized version of the scene depth rendered from the light source view into a texture and then compare its distance to the light source with the stored depth values for each screen space fragment. Whenever the stored distance is smaller, the fragment lies in shadow. Hourcade and Nicolas [12] propose to store object IDs in the texture instead of depth values, so that the shadow evaluation is only based on an ID comparison, making the use of a depth bias unnecessary. In our work, this approach is used in different variations. A key novelty in our work is the transfer of the basic idea of the shadow mapping algorithm, i.e. the hardware-accelerated mapping of a texture containing visibility from a certain view point to the domain of person path simulations, where it is used for visibility identification of objects in a 3D environment.

2.2 Crowd Rendering

Tecchia et al. [17] describe several aspects and methods used for crowd rendering. Topics like collision detection and behavior modeling are tackled in this work, explaining the major aspects of crowd rendering in a detailed way. In our work, we use the technique described by Dudasch [7] to visualize large crowds with DirectX *hardware instancing* and animation data provided by a texture-lookup.

2.3 Simulation

Recent advances in computational technologies have led to the development of application-specific simulation models focusing on different aspects of the collective behavior, using different modeling techniques. It can be distinguished roughly between two broad areas of crowd simulations. The first area is high-quality visualization for movie productions or games, where usually realism of the behavior model is not the priority (for a survey see [18]). These applications aim at a convincing visual result.

The second group focuses on realism of behavioral aspects with usually simple 2D visualization like evacuation or crowd dynamic simulations. In this area, the behavior represented in the simulations is usually restricted to a narrow range with efforts to quantitatively validate the fit of results to real-world observations of particular situations [19]. Visualization is used to help understanding simulation results, but it is not crucial and in most cases a schematic representation with colored dots is used.

Techniques employed in both areas can be distinguished between the different levels of description related to Mirosław [14] ranging from *macroscopic models* that do not distinguish individuals and describes characteristics at the level of flow-speed-density equation (such as the models implemented in Pedflow, Space Syntax methods, Pedroute), *mesoscopic models* at the level of the statistical description and *microscopic models* at the level of interacting entities using cellular automata such as STEPS, PedGo or the social force paradigm such as Legion, the pedestrian module of VISSIM and CAST. Agent based microscopic modeling is an approach for simulating pedestrians as single autonomous individuals by supplying a detailed representation of their behavior, including decisions on various levels (e.g. related to orientation and navigation) and interactions with other pedestrians in the crowd. The goal is to reproduce realistic single autonomous and emergent collective crowd behavior.

Compared to other models, social force-based models have been found to describe pedestrian behavior more realistically [2]. The most prominent social force model is Helbing's model [10] which is also used in the presented approach. This model has been calibrated and adapted to real world data by Johansson et al. [13]. The social force model was also extended by Musse to include individualism [4]. Pelechano et al. [15] merged rule-based and social force-based models and incorporated psychological state into the pedestrian simulation model. Shao and Terzopoulos [16] used a complex cognitive and behavior model for planning, but did not attempt realistic small-scale motion behavior like the social force model.

Many currently available simulation models are based on the assumption that all pedestrians know the infrastructure and consequently all pedestrians choose the fastest path to reach their goal.

Not every pedestrian is familiar with the infrastructure and wayfinding abilities are influenced by a number of physical, psychological, and physiological factors that will influence the ability of people to detect and correctly interpret the information conveyed by the signs. It is therefore important to include wayfinding in pedestrian simulation to achive realistic simulation results.

3. SYSTEM OVERVIEW

The presented work is separated into three modules: *Visible Object Identification*, *Simulation*, and *Visualization*. The data flow is depicted in Figure 1. For each simulation cycle, the positions of the simulated persons (*agents*) are sent together with a query for visible objects of a certain person (*viewer*) to the *Visible Object Identification* module (see Section 4). The visible object identification uses the 3D model of the environment to provide visibility information of objects to the *Simulation*.

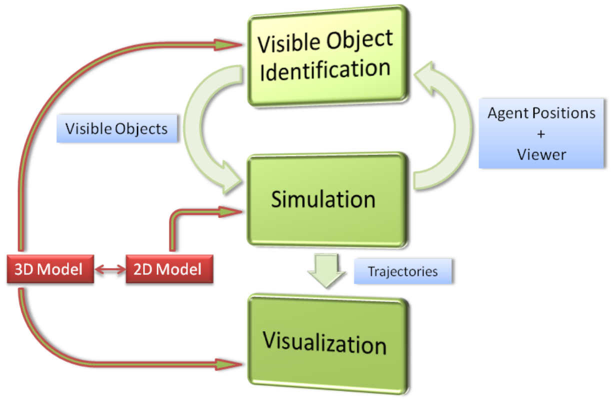


Figure 1: Visual description of the System which combines a pedestrian simulation using real-time rendering techniques to identify visible objects in a virtual 3D environment and a visualization module for visibility exploration and evaluation of the simulation results.

This information is used by the *Simulation* to calculate the next position of the viewer in a correlating 2D model (see Section 5). The simulation saves trajectory data of each single simulated agent and sends it to the *Visualization* module, where corresponding walking paths are visualized in the 3D model, making a visual evaluation of the results possible (see Section 6). Together with the visualization of the calculated 3D visibility of the environment, decisions made inside the simulation can easily be reproduced and understood.

The visualization module in the described system provides two different modes to show and explore visibility: The *viewing field visualization* shows visible regions of the viewer from a certain time step of the simulation. It is possible to step through time interactively to explore the data in a 3D environment and retrace decisions made in the simulation. The *attention region visualization* shows visibility over a time range of interest. Accumulated visibilities are shown in this view to see which regions are in main focus of the viewer and which ones are barely recognized during his way through the building. Section 6 describes both visualization modes in more detail.

4. VISIBILITY CALCULATION

The definition of an extended visibility model to emulate human sight in a virtual 3D environment is a major extension for an improved simulation with plausible behavior inside unfamiliar public buildings. By extending commonly used 2D methods with three-dimensional visibility information, occluding objects, moving crowd and access to semantic information can be taken into account in a realistic way.

4.1 Visibility Model

To describe the identification of objects, a visibility model based on the research of Xie et al. [22] has been defined. In the used model, objects in the center of a person's field of view should be recognized with full visual power (i.e. a defined value of perception - full visual power describes regions which are completely recognized by the viewer). The outer border of the viewing field is bound by an ellipse with a horizontal and vertical viewing angle (α, β) where visual power reaches zero. A falloff of visual power is modeled by a density function v dependent on the location relative to the center of the viewing field. Figure 2 illustrates this model. In the implemented examples the density function v describes a linear falloff over the viewing field, and the angles (α, β) have been set to $(50°, 30°)$.

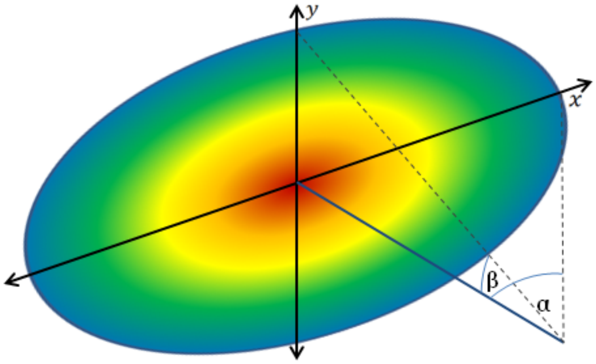


Figure 2: Model of the visual power density function v over an elliptic viewing field parameterized by the angles (α, β).

In the presented approach, visibility of an object i is assumed to be equivalent to its projected area in the viewing field weighted by the described visual power v at each differential surface fragment $p(x, y)$. The accumulated result described by the surface integral over the projected visible object in Equation 1 gives an absolute visibility value per object V_i. This allows the simulation to decide together with additional semantic information whether an object is recognized or not.

$$V_i = \iint\limits_D v(x,y) \cdot p(x,y) \mathrm{d}x\mathrm{d}y \tag{1}$$

4.2 Crowd Rendering on Various Paths

Inclusion of crowd rendering in the visualization and visibility object identification leads to a more realistic analysis of the viewer's environment recognition. The simulation calculates every position of the viewer person and the crowd to get point lists for the visualization of movement over time. For accurate visibility on the paths all simulated persons are handled. In order to animate large numbers of individuals we use DirectX *hardware instancing* as described by Dudasch [7]. With this method, the same pedestrian model is efficiently reused multiple times at varying locations with different parameters. Individual animation data is provided by a lookup-texture containing key-frame transformations.

With the described techniques a realistic environment for the viewer person inside a crowd is created. Pedestrians may block the viewing field of the viewer and visibility visualization and calculation for simulation becomes more realistic.

4.3 Visible Object Identification

A 3D model of the environment allows to accurately calculate possibly visible parts within that model from a single viewpoint. Object IDs are set for every object reasonable for decisions of the viewer in the simulation. These IDs are connected to semantic information to identify the reaction on the object recognized. The concept of visibility calculation for lights in real-time rendering applications - "Whatever the light sees is illuminated; what it does not see is in shadow." [1] - has been adapted for the calculation and visualization of visibility. In particular, an ID-based approach similar to the technique of Hourcade and Nicolas [12] is used.

The viewer is the reference point of the visibility calculation and can be seen as a light source. The GPU is used to rasterize visible objects with the ID values from a viewpoint into an ID buffer. Therefore the simulation is able to make decisions based on recognized items. The viewing field of the viewer is bound by a horizontal and vertical angle α and β defining a perspective projection P. Together with the position of the head $\mathbf{l}$ and viewing direction $\mathbf{e}$, which define a view-space V, an image as seen by the viewer can be rendered using the view-projection transformation VP. The visibility model is used to accumulate a meaningful visibility value per object: Each pixel in the rendered image represents a discrete visible surface fragment of an object with equal area in the viewing field. The given weight of the viewing density function v at the pixel location adds viewing power to the visibility value of the object defined by the ID of the rendered pixel. Summing all discrete visibility values of an object requires grouping and accumulation of all pixels of the rendered image. This is no typical rendering step, but with GPGPU computations of modern graphics hardware this task can be processed in real-time.

The visibility value gives an absolute measurement of the objects visibility. From the object ID, the simulation retrieves additional predefined semantic information, and together with the distance from the viewer, the position of the object center and its viewing angle, an object can be identified as a sign and whether it is close enough to be read or not as mentioned by Xie et al. [22]. With the ID identification it is additionally possible to set any object of interest for the scene. The viewer is able to handle different tasks like buying a ticket at a railway station or to look for a shop to get something to eat. Such information is important for the simulation which is based on the defined needs of the viewer to decide his way realistically based on the seen environment.

5. VISIBILITY AIDED SIMULATION

Pedestrian motion behavior is often described in three different levels [11]. The *strategic level* determines the arrival time of the pedestrian at the infrastructure, its entry position and the pedestrian's goal (e.g. going to the train). In our case this level is not modeled but the origin and the final goal (e.g. getting a certain train) are predefined. The *tactical level* describes the route a pedestrian will choose to move through the infrastructure. This is where the 3D visibility calculation as described in Section 4.3 comes into play by making it possible for a pedestrian to select a way based on his current view. The *operational level* calculates the actual movements towards the next goal, including collision avoidance. In this work, an agent-based approach is used to assign and vary the characteristics of individual pedestrians as required. The model development will be supported by empirical data collection providing data of people's real life bahavior. This empirical data collection enhances the understanding of parameters and their influence to individuals' decisions to choose a certain route.

5.1 Operational Level

Human motion on an operational level is modeled based on a social force model [10]. Force models provide explicit equations for the movement of pedestrians. They are defined in continuous time and space by providing differential equations for the acceleration of an agent. Acceleration is then integrated once in order to obtain velocity, and integrated twice in order to obtain the new position of the agent. Using an analogy from physics, acceleration is identified with force. Different forces act on each agent i as follows:

$$F_i = F_i^a + F_i^p + F_i^w, \tag{2}$$

where F_i^a denotes the attractive force directed towards the pedestrian's goal, F_i^p denotes the repulsive forces directed away from other pedestrians and F_i^w prevents the pedestrian from colliding with walls or other obstacles. As a first approximation the same equations are used for modeling pedestrian and wheel chair movements varying two parameters: 1) The desired speed of the agent on which the attractive force is depending and 2) the body height on which the repulsive forces are depending.

5.2 Tactical Level - Wayfinding

Route planning and navigation are intuitive skills for pedestrians, but it is no easy task to simulate these abilities in a virtual environment. In conventional simulation models the fastest path is computed or stochastic rules are applied to get a route from the origin to the destination.

For the tactical level field tests at the train station Vienna North using thinking aloud voice recording and time-motion tracking were made. A detailed description of measuring behavior can be found in [8]. The results identified elements of the orientation system in the built environment that respondents use to navigate.

In the proposed approach the cognition of guidance systems is modeled using insights from field tests and creates the possibility to simulate agent navigation through an unknown infrastructure using present signage. No routing graph has to be defined in advance, only the visual information obtained from the 3D model is used to find the route to the destination. The order of tasks (e.g. buying a ticket, getting departure information) which has to be performed by the agent on his way through the station is predefined. Paths between goals have to be found according to the following wayfinding algorithm: Whenever the agent cannot see the next goal (i.e. the location of the next task) from his current position he looks (see previous section) for the nearest sign which is close enough to be readable and indicates the direction to the next goal. If he cannot find any appropriate sign he moves on with current speed and direction for one simulated second and then starts his search again. After finishing a task when no information referring to the next goal is available the agent moves back to the last visited sign. Information about the visible environment is gained by using visibility calculation and information identification as described in section 4.3.

6. VISUALIZATION

For a proper exploration of visible regions and recognized areas during the simulation, two visualization modes have been developed.

6.1 Viewing Field

An explorative 3D visualization of the viewer's sight at a specified time step is the first mode in the presented system. The computation of this visualization is similar to the visibility calculation in Section 4.3 using an ID buffer. The visual power as described in the model of Section 4.1 is visualized in real-time. According to the visibility model, the visibility value of a surface fragment is dependent on its projected area in the viewing field. Since the visualization is directly applied on the object surface in 3D, the projected area in the viewing field of a person at location $\mathbf{l}$ is calculated with the fragment position $\mathbf{f}_p$ and its orientation $\mathbf{f}_n$ as follows:

$$p(f) = \frac{\mathbf{f}_n \mathbf{s}}{d^2}, \text{ where } \quad d = |\mathbf{l} - \mathbf{f}_p| \text{ and } \mathbf{s} = \frac{\mathbf{l} - \mathbf{f}_p}{|\mathbf{l} - \mathbf{f}_p|}, \qquad (3)$$

where d is the distance and $\mathbf{s}$ is the normalized direction vector to the viewers head at l. The projected area $p(f)$ is then weighted by the visual power density function v which is evaluated in the parameterized viewing field (Figure 2). Therefore the surface fragment position f_p is transformed by the view projection transformation VP giving the location used to calculate the visual power.

The resulting values are normalized depending on a user specified parameter which defines a distance that still allows recognition of an area in the center of the viewing field. This allows to visualize recognition of different types of objects, e.g. a sign is specified to be readable at a distance of 10 meters. The visual power in visible regions is colored based on a one dimensional transfer function. Figure 3 shows the viewer and the color coded region in sight. Another person blocks the sight of the viewer, but the sign at the exit is still perfectly recognized. By the exploration of the viewer walking around in the public building, while his viewing field is visualized, decisions made by the simulation can be explored.

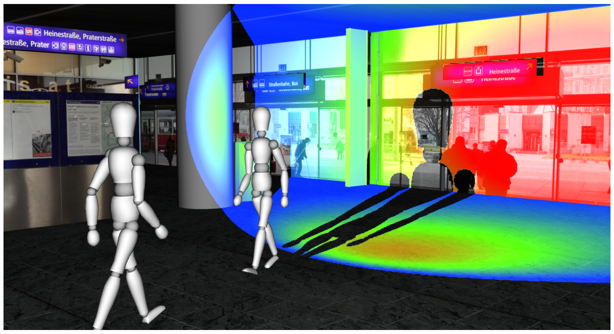


Figure 3: A viewer looks to the exit of the station - his view is partly blocked by another person.

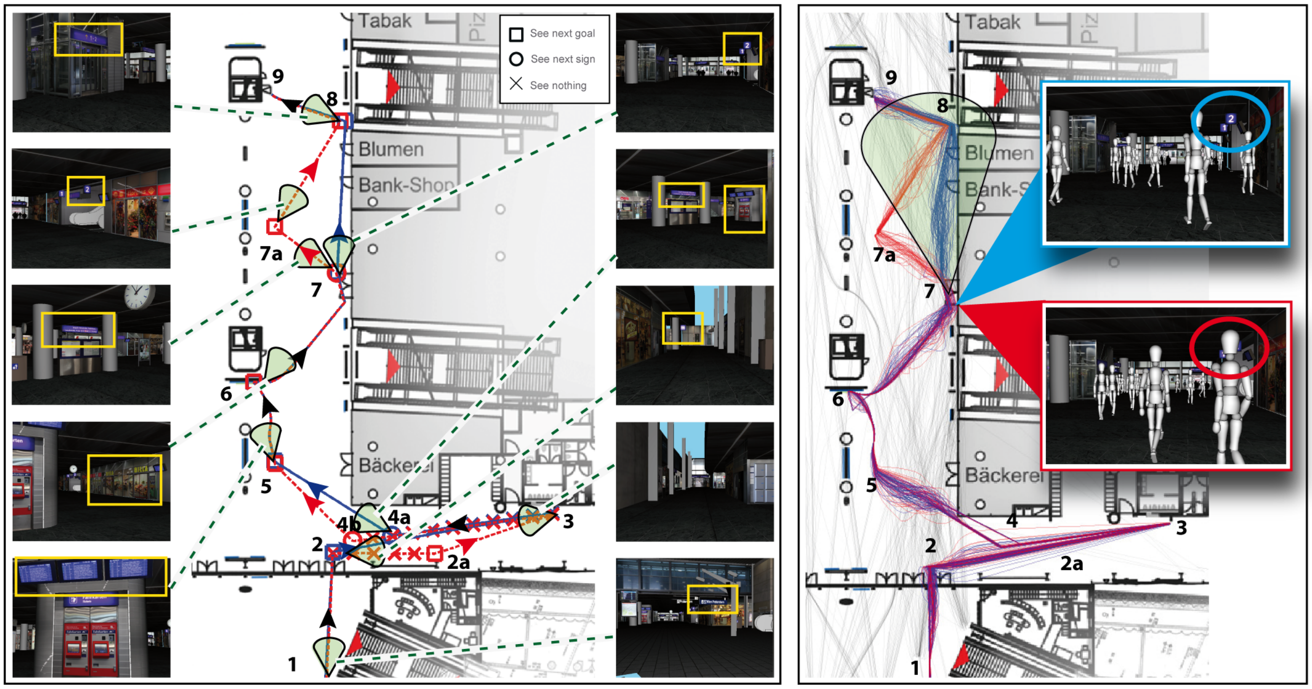
6.2 Attention Regions Visualization

The attention regions visualization is based on the viewing field visualization described above, but shows the accumulated visibility of the viewer person over time. Although this is a computationally expensive task, the time required for computing is relatively short (see Section 7.2) due to further exploitation of hardware acceleration, and depends on the number of time-steps in the specified range. The GPU is used to store the accumulated visibility power in a previously assigned texture atlas, allowing real-time exploration after precomputation.

In contrast to the single time step visualization, the computation is performed in texture space. Therefore, for each texel in the texture atlas, the corresponding surface position $\mathbf{f}_p$ and orientation $\mathbf{f}_n$ need to be precomputed. For every time step, a visibility calculation like in the first visualization mode takes place and the results are accumulated. Occlusion is again calculated using the shadow mapping algorithm. While the evaluation of a single time step can be performed in real-time, this visualization is calculated at a rate of several hundred time-steps per second.

7. RESULTS

The combined approach of simulation and visualization makes it possible to evaluate visibility of the guidance system and to show areas with leaks of guidance information for people unfamiliar with the infrastructure, especially for elderly and handicapped people with reduced reception capabilities. The agent moves through the station using the described wayfinding algorithm depending on the visibility of the guidance information. This could be shortened by a reduced agent vision or a very dense crowd, especially for wheel chair drivers with a lower point of view. The used 3D environment including 3D human models for the simulated agents gives the opportunity to calculate possibly occluded sections from one viewpoint realistically.



a)

b)

Figure 4: Trajectories of simulated wheel chair drivers. The left hand side (a) shows a scenario in the empty hall comparing full (blue) and reduced reception capabilities (red), Circles indicate positions where the agents can read a sign, squares where the agent can see the next goal and 'x' where the agent cannot find any helpful information and therefore moves on. The right hand side (b) shows the morning peak hour scenario with direct routes (blue) and detours to point 7a (red), caused by the occlusions of signage through the dense crowd.

We show the results achieved with the visibility aided simulation in Section 7.1, and discuss the visual output and the performance of the visualization in Section 7.2.

7.1 Visibility Aided Simulation Results

In order to demonstrate the wayfinding algorithm's applicability to real-world environments, different scenarios have been simulated and the results are shown in Figure 4. All scenarios are based on the same tasks: A wheel chair driver starting at the elevator exit into the main hall of the "Praterstern" station has to reach the train to "Stockerau". Apart from getting the departure information and buying a ticket, he needs to go to the toilet and wants to get some snacks for the journey. The order of activities is predefined: First go to the toilet (3), then buy the ticket (6), look for a monitor to get the departure information, go to the supermarket (7) and finally, use the elevator (9) that brings you up to the platform where the train is departing.

The three scenarios for the wheel chair driver differ in the level of crowdedness and the agent's vision: Scenario 1) empty hall, full vision with no visual impairments, scenario 2) empty hall, half vision meaning half of the viewing distance shown in 4a and scenario 3) crowded hall according to the morning peak (7:15-8:15), full vision to show the applicability of the approach for occlusions in crowds.

Figure 4a) shows the results of the first two scenarios. In both cases the agents start at position 1 where they can get an overview of the hall. The overhead sign which indicates the direction to the toilet at position 2 is readable for both of them. Having walked past the sign they both stop and search for the toilet in the given direction. The agent with full vision can read the sign at entrance of the toilet at position 3 already from here and moves directly to the toilet. The agent with reduced vision moves on in the given direction searching five times for the toilet until position 2a where he can read the toilet sign at position 3. Leaving the toilet both agents cannot find any helpful information. Therefore they move back to the last visited sign. At position 4a the agent with full vision can already read the sign at position 5 which indicates the direction to the ticket machine. The second agent needs to move to position 4b where he is near enough to read the sign at position 5. Having reached position 5 both can see the ticket machine at position 6 where they buy their tickets. Since a monitor is mounted above the ticket machine they get the departure information there as well. The train to "Stockerau" is leaving at platform 2. From position 6, the entrance of the supermarket at position 7 is visible for both and they move directly there. Leaving the supermarket the agent with full vision can read the sign at the staircase to platform 2 (position 8), the second agent can only read the sign at position 7a which indicates the direction to platform 2. At position 7a he can read the sign at position 8 as well. Having realized that at position 8 there are only stairs and escalators they look around and find the elevator to platform 2 at position 9.

Figure 4b) shows the results of the third scenario where the agent with full vision is moving through the hall during the morning peak hour (7:15-8:15). Hundred simulation runs with slightly different start times were selected and the trajectories are plotted in the single figure (blue lines) to show the differences in dense crowds (light gray lines). In all runs the trajectories till the supermarket (position 7) are identical among each other and also with the trajectory of the agent with full vision in Scenario 1. The only difference at position 4 after the toilet back to the main hall comes from the occlusion of the sign at position 5 through the pedestrian flow crossing in front of the agent.

After leaving the supermarket at position 7, the agents are following different paths. In 59 of the 100 simulation runs the agent had free sight to the platform 2 sign at position 8 (see red circle in the upper screen-shot) and moves on like in scenario 1 (blue path). Whereas, for the agent in the other 41 simulation runs the sign is occluded behind one or more pedestrians and therefore the agent makes a detour via the sign at position 7a (red path) which indicates the direction to platform 2. The two screenshots in Figure 4b) shows the two different situations with free sight and occlusion through to other pedestrians to the sign.

Each scenario showed different routes compared to the shortest route due to differences in the range of vision and crowdedness. Especially for wheelchair users crowdedness has a significant impact. In 41% of our simulation runs crowds occlude signage or form obstacles leading to a higher rate of maneuver, longer routes and travel discomfort. The approach has shown to be capable of simulating handicapped people finding their way through a transport infrastructure using signage information only and provides an outlook about the impact of occlusions through dense crowds.

7.2 Visualization Results

During our tests for the *Attention Regions Visualization* (Section 6.2), the system needed about 1.3 seconds to accumulate 750 timesteps, using a shadow map resolution of 512x512 pixels and a 2048x2048 pixel texture atlas. The testing system was a PC with an Intel Core2Quad processor with 2.83 GHz, 8 GB RAM and a Geforce GTX 280 Graphics Card with 1 GB RAM. The main scene was the model of the mentioned railway station Vienna North with more than 140000 triangles.

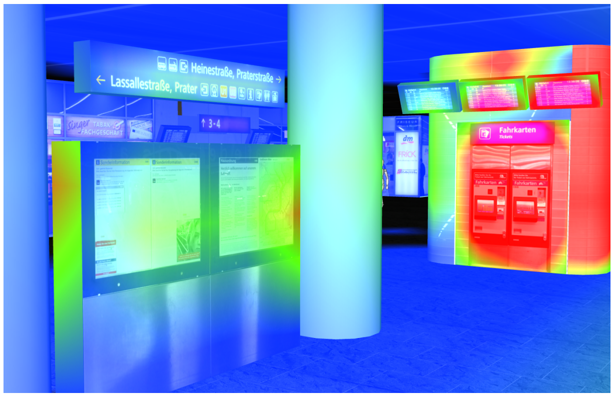


Figure 5: Color-coded visualization of the station over the whole simulated time-range showing the degree of the viewer's attention at different regions.

Figure 5 shows a section of the guidance system in the second visualization mode. The ticket machine and the monitors above are coded in red, meaning well recognized by the viewer. Information tables and the guidance system above did not get much attention. If this sign had been important for the viewer to perform his given task (e.g. finding a ticket machine on the way to the bus station), this would indicate a problem in the guidance system. Figure 6 gives another example of visualized attention regions during a simulated walk to the bus station

8. CONCLUSION AND FUTURE WORK

The presented work shows a prototype which combines human behavior simulation with real-time rendering and visualization. This cooperation of modern technologies of different disciplines enables an agent to find its way autonomously through a building based on visible information. The developed prototype has been applied to a realistic scenario to demonstrate the functionality of the algorithms. A 3D model of a railway station creates the basis for visibility calculation and visualization. 3D rendering including human 3D models representing the simulated agents gives at the one hand the opportunity for the simulation to calculate possibly occluded sections from a viewpoint realistically and at the other hand to explore the simulation results and to analyze the efficiency of the guidance system in public buildings.

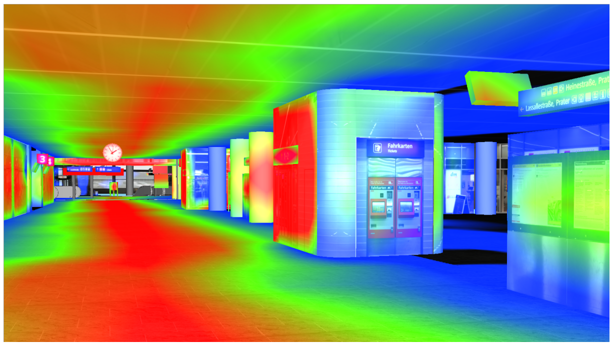


Figure 6: Attention regions in the station visited by a simulated viewer. Computation time for time-ranged based visualization: 1.3 sec including 750 time-steps in a scene with about 140.000 triangles

Visible regions in public buildings can furthermore not only be seen as guidance systems, but also as an indicator for the quality of locations where advertisements and shops are placed. This and similar analyses can easily be performed with the help of the visualized quantifications of human perception.

The calculation of visible sections can probably be improved by using psychological effects, like attention. Lighted or animated signs for example catch the focus of people, and active search for guidance system elements leads to different regions of interest. Another interesting aspect of human vision is how long an object is in focus and if visibility was long enough for recognition. The presented approach already includes this aspect with the attention region visualization. With advanced techniques in the field of temporal coherence it might be possible to explore such regions in real-time and include time dependent object recognition in the simulation.

Using visualization tools to improve a simulation seems to be a promising start for further research – especially in the case of understanding and analyzing movements of impaired people. While visual analysis of simulated trajectories and views of a person in a wheelchair has already given new insights and made it possible to improve the corresponding simulation parameters, our visual object identification system could easily be extended to account for visually impaired people (e.g. near-sighted persons could be simulated by reducing the visual power for objects in far distance, ...), having an immediate positive impact on the planning of easily accessible public buildings.

Visualization of visibility and the calculation of recognized objects in a virtual environment could moreover be used in different simulation scenarios. Traffic simulations might be improved with a derivation of the presented approach to see if cars are recognizable in narrow curves or to evaluate the positions of signs on streets.

9. ACKNOWLEDGMENTS

This work is part of the project MASIMO and has been funded by the research program line ways2go within the framework of the Austrian strategic initiative IV2Splus Intelligent Transport Systems and Services plus under the project number 819192.

10. REFERENCES

[1] T. Akenine-Möller, E. Haines, and N. Hoffman. *Real-Time Rendering 3rd Edition.* A. K. Peters, Ltd., Natick, MA, USA, 2008.

[2] D. Bauer. Comparing pedestrian movement simulation models for a crossing area based on real world data. In R. D. Peacock, E. D. Kuligowski, and J. D. Averill, editors, *Pedestrian and Evacuation Dynamics*, pages 547–556. Springer US, Boston, MA, 2011.

[3] N. Brändle, T. Matyus, M. Brunnhuber, G. Hesina, H. Neuschmied, and M. Rosner. Realistic interactive pedestrian simulation and visualization for virtual 3D environments. In *Proceedings of the 15th International Conference on Virtual Systems and Multimedia*, 2009.

[4] A. Braun, S. R. Musse, L. P. L. d. Oliveira, and B. E. J. Bodmann. Modeling individual behaviors in crowd simulation. In *16th International Conference on Computer Animation and Social Agents (CASA 2003)*, page 143, Los Alamitos, CA, USA, 2003. IEEE Computer Society.

[5] M. Brunnhuber, G. Hesina, R. F. Tobler, and S. Mantler. Interactive person path analysis in reconstructed public buildings. In *Proceedings of the 2009 Spring Conference on Computer Graphics*, SCCG '09, pages 145–150, New York, NY, USA, 2009. ACM.

[6] D. Cohen-Or, Y. L. Chrysanthou, C. T. Silva, and F. Durand. A survey of visibility for walkthrough applications. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER*, 9(3), 2003.

[7] B. Dudash. Animated crowd rendering, 2007.

[8] V. Egger, H. Schrom-Feiertag, L. Ehrenstrasser, and G. Telepak. Creating a richer data source for 3d pedestrian flow simulations in public transport. In *Proceedings of the 7th International Conference on Methods and Techniques in Behavioral Research*, MB '10, pages 20:1–20:4, New York, NY, USA, 2010. ACM.

[9] E. Eisemann, M. Schwarz, U. Assarsson, and M. Wimmer. *Real-Time Shadows.* A.K. Peters, 2011.

[10] D. Helbing and P. Molnar. Social force model for pedestrian dynamics. *Physical Review E*, 51:4282–4286, 1995.

[11] S. P. Hoogendoorn and P. Bovy. Pedestrian route-choice and activity scheduling theory and models. *Transportation Research, Part B: Methodological*, 38:169–190, 2004.

[12] J. C. Hourcade and A. Nicolas. Algorithms for antialiased cast shadows. *Computers & Graphics*, 9(3):259–265, 1985.

[13] A. Johansson and D. Helbing. Analysis of empirical trajectory data of pedestrians. In W. W. F. Klingsch, C. Rogsch, A. Schadschneider, and M. Schreckenberg, editors, *Pedestrian and Evacuation Dynamics 2008*, pages 203–214. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[14] L. Mirosław. Microscopic, mesoscopic and macroscopic descriptions of complex systems. *Probabilistic Engineering Mechanics*, 26(1):54–60, Jan. 2011.

[15] N. Pelechano, J. M. Allbeck, and N. I. Badler. Controlling individual agents in high-density crowd simulation. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 99–108, San Diego, California, 2007. Eurographics Association.

[16] W. Shao and D. Terzopoulos. Autonomous pedestrians. *Graph. Models*, 69(5-6):246–274, 2007.

[17] F. Tecchia, C. Loscos, and Y. Chrysanthou. Visualizing crowds in real-time. *Computer Graphics Forum*, 21:753–765, 2002.

[18] D. Thalmann and S. R. Musse. *Crowd simulation.* Springer, 2007.

[19] P. A. Thompson and E. W. Marchant. Testing and application of the computer model 'SIMULEX'. *Fire Safety Journal*, 24(2):149 – 166, 1995.

[20] L. Williams. Casting curved shadows on curved surfaces. *SIGGRAPH Comput. Graph.*, 12:270–274, August 1978.

[21] P. Wonka, M. Wimmer, K. Zhou, S. Maierhofer, G. Hesina, and A. Reshetov. Guided visibility sampling. *ACM Trans. Graph*, page 502, 2006.

[22] H. Xie, L. Filippidis, S. Gwynne, E. R. Galea, D. Blackshields, and P. J. Lawrence. Signage legibility distances as a function of observation angle. *Journal of Fire Protection Engineering*, 17(1):41–64, Feb. 2007.