



Article The Impact of Obstacle's Risk in Pedestrian Agent's Local Path-Planning

Thanh-Trung Trinh ^{1,*} and Masaomi Kimura²

- ¹ Graduate School of Engineering and Science, Shibaura Institute of Technology, Tokyo 135-8548, Japan
- ² Department of Computer Science and Engineering, Shibaura Institute of Technology, Tokyo 135-8548, Japan; masaomi@shibaura-it.ac.jp
- Correspondence: nb18503@shibaura-it.ac.jp

Abstract: While the risk from the obstacle could significantly alter the navigation path of a pedestrian, this problem is often disregarded by many studies in pedestrian simulation, or is hindered by a simplistic simulation approach. To address this problem, we proposed a novel simulation model for the local path-planning process of the pedestrian agent, adopting reinforcement learning to replicate the navigation path. We also addressed the problem of assessing the obstacle's risk by determining its probability of collision with the obstacle, combining with the danger from the obstacle. This process is subsequently incorporated with our prediction model to provide an accurate navigation path similar to the human thinking process. Our proposed model's implementation demonstrates a more favorable result than other simulation models, especially in the case of the obstacle's appearance. The pedestrian agent is capable of assessing the risk from the obstacle in different situations and adapting the navigation path correspondingly.

check for **updates**

Citation: Trinh, T.T.; Kimura, M. The Impact of Obstacle's Risk in Pedestrian Agent's Local Path-Planning. *Appl. Sci.* **2021**, *11*, 5442. https://doi.org/10.3390/ app11125442

Academic Editor: Ciro Caliendo

Received: 30 April 2021 Accepted: 9 June 2021 Published: 11 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). Keywords: pedestrian; path planning; risk assessment; reinforcement learning; agent; simulation

1. Introduction

Accurate replication of the human navigation behavior in a pedestrian simulation model plays an important role in the studies within the safety domain. Correspondingly, research in this particular field has been greatly active. For instance, many studies of pedestrian simulation for evacuation activities have been beneficial to the design in safety features of construction projects [1]. Another example is the studies in pedestrian behavior, which are crucial for urban planning and landscape design [2,3]. Recently, along with the rising trend of autonomous vehicles, pedestrian simulation studies have attracted increasing interest, especially in the situation of crossing with vehicles, to avoid possible fatal accidents [4,5]. While these studies could construct a sufficient reproduction of the pedestrian navigation behavior in certain applications, for example, the robot movement in pedestrian roads, their approaches might not be able to provide a human-like behavior needed for some research, in risk and safety problems for instance. The goal of a navigation model in robotics is to create a robust and efficient movement that is deemed safe and comfortable by humans, which does not require an accurate replication of human navigation.

Recent studies in pedestrian simulation have been approaching the problem of replicating a human-like navigation behavior by adopting various concepts in cognitive science and behavioral psychology. This could be challenging as the human cognitive system is excessively complex. The objective of the cognitive system is to process information and making decisions. Every minute, a large amount of information surges into the human mind. This, combined with memory and emotion, through various layers of the human conscious and subconscious mind, is formed into a cognitive map. The decision-making process is subsequently carried out using the data on the cognitive map and the person's own experience. As an example, in the pedestrian path-planning process, the two following tasks are carried out sequentially. In the global path-planning task, the pedestrian uses his experience and knowledge to specify his destination and plan the route to get there. In the local path-planning task, the surrounding environment is often observed via human vision and transformed into a topological map. Subsequently, the pedestrian estimates the path would be taken before carrying out the actual movements [6]. While there is a great deal of research that addresses the global path-planning, the route selection process to the destination [7,8], the studies of the local path-planning problem are generally scarce. For the few studies that focus on this problem, their models often try to optimize certain objectives, such as next state optimizing [9] or way finding [10]. In real life, people tend do not usually choose the most optimized solution [11]; therefore, these models may yield inaccurate navigation behavior in certain situations.

Addressing the obstacle's risk and danger is also a factor that is often overseen by many studies. Although the majority of research in pedestrian simulation considers the obstacle in collision avoidance, to our best knowledge, not many studies have addressed how its danger affects the pedestrian's choice. For the papers that discuss this problem [12], the models proposed are quite limited in using the empirical approach without considering the human cognitive factors. The results of these models could be consequently insufficient, especially in the case the danger of the obstacle greatly alters the path choice of the pedestrian. For safety-focused applications, this problem could produce undesirable results, possibly causing significant consequences as a result.

For that reason, we proposed a novel pedestrian simulation model focusing on the local path-planning process, considering the obstacle's danger and risk assessment while taking account of human cognitive factors. Our model adopts the concept of deep reinforcement learning (RL), a neural network-based machine learning technique, for the training of the pedestrian agent. The approach is inspired by the mechanism of human cognitive system. More specifically, in reinforcement learning, the agent learns to take actions depending on the states of the environment based on the proper rewarding, similar to human's trial-and-error learning approach. Deep reinforcement learning approaches also employ artificial neural networks, which were inspired by the mechanisms of the biological neural network in the human brain. Thanks to that, the aspects in obstacle's danger and risk assessment are further explored in a similar mean as to how humans address dangers in real life. The implementation of our model has demonstrated favorable results. The pedestrian agent in our model was able to plan a more realistic navigation path compared to traditional models, especially in the case of interacting with an obstacle within the environment.

The remainder of this paper is structured as follows. Section 2 presents the studies related to our research, followed by Section 3 which briefly explains the background of the concepts mentioned in this paper. Section 4 introduces the main methodology of our research, consisting of the path-planning training, point-of-conflict predicting, and obstacle danger and risk assessment, which later represented comprehensively in Section 5, Sections 6 and 7, respectively. Subsequently, Section 8 demonstrates the results of our implementation. Section 9 gives our discussion and finally, and Section 10 concludes our paper.

2. Related Works

Many studies propose pedestrian simulation models by using physics-based concepts, such as Newtonian mechanics or fluid dynamics [13]. One of the most influential models in pedestrian simulation Social Force Model (SFM), employs the concept of forces to the pedestrians and the obstacles [14]. The idea of this model is to treat every object within the environment as a force-based object, which impulses and attracts each other similar to magnets. Based on the concept of SFM, several studies have proposed other factors which could affect the pedestrian agent's navigation, such as heading direction [15] or the connection between speed and density [16]. Apart from force-based models, some models

approach the problem by utilizing fluid dynamics to simulate the motion of the pedestrian crowds [17]. Considering the pedestrian path-planning problem, while the behaviors replicated from these models could be sufficient in some basic circumstances; in most cases, there is a large distinction between the behavior and the anticipation within human thoughts. A possible cause is that humans rarely think of animals or other pedestrians as physical objects affected by different forces.

Besides physics-based models, other studies adopt the agent-based approach to model the pedestrian simulation. Compared to force-based models, adopting human thinking is more accessible in agent-based ones. Several studies, mostly in the robotic domain, have tried to simulate human behaviors in their models by proposing various concepts. For instance, Kruse et al. [18] introduced the concept of human comfort, indicating the factors that affect how the movement makes humans feel comfortable. Another example is a study by Cohen et al. [19], in which the pedestrian agent could consider its decision between exploitation and exploration. In addition, there are several studies adopting reinforcement learning for the pedestrian agent. For example, Martinez-Gil et al. [20] employed the q-learning algorithm to implement a multi-agent navigation system. These agent-based models could replicate basic situations, but for more complicated ones, especially when considering the danger of obstacles, such models have certain limitations. In our previous study [21], we have addressed the path-planning problem in accordance to obstacle's danger; however, the model is not well scalable in different environments. In addition to reinforcement learning, inverse reinforcement learning could also be adopted [22], but it is hard to export human behavior information from existing available datasets.

Regarding the problem of obstacle prediction, there have been many studies addressing different problems in the prediction of obstacles and pedestrians. Many of these studies depend on the processing of image or video data [23–25]. Others suggest predicting the movement of pedestrian obstacles based on human behaviors [26], or body language [27,28]. Several studies approach this problem by using a map of probability [29,30]. We also proposed an approach for obstacle prediction by introducing the concept of point-of-conflict [31], which performs well in both cases of moving obstacles and pedestrians.

3. Background

3.1. Reinforcement Learning

Sutton and Batto introduced the concept of reinforcement learning [32], in which agents learn to improve the outcome of their *actions* to the *states* of their environment. This mapping from action to state is called the *policy*. To define how good a policy is, a *reward* needs to be given, which could be positive or negative. In a noisy environment, an action could receive a positive reward, but that action may also eventually lead to a worse result. Because of this reason, the goal of a reinforcement learning agent is to optimize the policy to achieve an advantageous *cumulative reward*.

The principle reinforcement learning model is defined as a Markov Decision Process (MDP). An MDP is a set of $(\mathbb{S}, \mathbb{A}, P, R, \gamma)$ where \mathbb{S} is a set of states; \mathbb{A} is the set of agent's actions; *P* is the probability function and γ is the discount factor. The probability *P* is calculated by

$$P_a(s,s') = Pr(s_{t+1} = s'|s_t = s, a_t = a),$$
(1)

where *a* is the taken action, *s* is the previous state, and s' is the current state.

The reward function *R* is formulated as

$$R_a(s) = (R_{t+1}|s_t = s, a_t = a).$$
(2)

$$V(s) = \max E\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, \pi(s_t))\right],$$
(3)

where $\pi : S \to A$ is the policy for the action A in the state S.

3.2. PPO Algorithm

Reinforcement learning algorithms are categorized into 2 categories: model-based and model-free algorithms. A model of the environment could be interpreted as the understanding of the agent about the environment. A model-based algorithm uses the model of the environment for planning by estimating future states before taking action. On the other hand, a model-free algorithm learns mostly by trial-and-error without any planning.

Proximal Policy Algorithm (PPO) algorithm is a model-free reinforcement learning algorithm, introduced by Schulman et al. [33]. It approaches the reinforcement learning problem by using a neural network to train the agent to find an optimized policy. One of the most important factors in a neural network algorithm is the loss function to measure the accuracy of the model. In PPO algorithm, the loss function is specified using an advantage value \widehat{A}_t , the variation between the reward for the current state and the expected reward. The loss function is formulated as follows:

$$\mathcal{L}^{clip}(\theta) = \hat{\mathbb{E}}\Big[\min(r_t(\theta)\widehat{A_t}, \, clip(r_t(\theta), \, 1 - \epsilon, 1 + \epsilon)\widehat{A_t})\Big],\tag{4}$$

where $r_t = \frac{\pi_{\theta}(a_t|s_t)}{\pi_{\theta old}(a_t|s_t)}$, and ϵ is a clipping hyper-parameter, which is utilized to keep the current good policy from being replaced by a worse one in a noisy environment; θ is the policy parameter, and $\hat{\mathbb{E}}$ indicates the empirical expectation over predefined timesteps.

4. Materials and Methods

To accurately simulate the pedestrian path-planning process considering the obstacle's danger using reinforcement learning, we need to determine how the human brain works in doing that task. More specifically, we need to address the mechanism of planning a navigation path by the human pedestrian. The reason we could do that is, as suggested in Section 1, reinforcement learning techniques share many similarities with the operation of the human cognitive system. Moreover, reinforcement learning techniques using neural networks, such as the PPO algorithm, even use a resembling structure as the human's neural system.

In reinforcement learning, the agent needs to learn to do any specific tasks the same way as a human does, i.e., via trial-and-error. Regarding the navigation task, there are several goals a human being needs to learn before being able to plan a navigation path efficiently. This is particularly similar to how children learn to navigate. Other than learning to reach a destination, they also need to learn to walk in the right way and avoid other obstacles. The instructions come from encouragements, as well as punishments, from different people, which resemble to the reward signals in reinforcement learning. Although the neural network used in a machine learning technique could benefit from much higher training scenarios compared to actual human beings. For example, a child could learn to reach the correct destination after several tries, it could take a reinforcement learning agent a few minutes to learn through millions of states of the environment. For that reason, the neural network could still learn to accomplish the equivalent task despite the limitation in its network structure.

However, just learning the navigation task through a trial-and-error approach might not be enough for efficient path planning. For a grown-up human to carry out the pathplanning task, further thinking processes are utilized. In particular, the cognitive predictive process is essential in the way the human brain processes many tasks, including navigation. This helps the adult pedestrians navigate more competently with fewer collisions with surrounding obstacles.

Another important process which humans gradually learn through their lives is the risk assessment of obstacle's danger. In a study by Ampofo-Boateng [34], it is indicated that children at different ages perceive danger differently. The older children could identify the danger more correctly, while younger children usually could not specify the danger apart from moving vehicles.

Because of these reasons, we need to address the risk assessment process and the prediction in the path-planning task for the model to replicate the planned path more accurately. More specifically, the risk assessment process in our model aims to replicate the observation of risk for our pedestrian agent, to be subsequently employed by the reinforcement learning model. Section 4.1 discusses the problem of obstacle's danger and the risk assessment, followed by Section 4.2, which proposes the overview of our pedestrian path-planning model.

4.1. Obstacle's Danger and Risk

The obstacle in our model is any person, animal, or object that would be considered as an obstruction in the pedestrian's thinking. Occasionally, the obstacle could be physical or abstract, such as a restricted area defined by traffic laws, for instance. The observed obstacle is defined by *spatial effect*, a term introduced by Chung et al. [35]. An example of this is a group of other pedestrians walking together. Theoretically, these are considered multiple obstacles, but because planning a path through these obstacles is viewed as unnatural and even impolite, such practice is not encouraged. In our model, these obstacles would be considered as a single obstacle. In addition, because of the spatial effect, the obstacle may dynamically change its properties. An example of this is the crossroad. If the light is red, the entire crossroad would be treated as an obstacle, but if the light is green, it is no longer viewed as an obstacle from the pedestrian agent's perspective.

For the agent's path-planning task, the most critical property of an obstacle would be its *risk* perceived by the agent. The difference in the perceived risk of the obstacle could greatly change how the agent plan the path. For example, if the obstacle is a highly dangerous one (e.g., a deep hole on the street), the pedestrian would very likely stay further away from it, as represented in Figure 1a. On the other hand, if the obstacle is safer (e.g., a shallow water puddle), the pedestrian is less likely to avoid it too much. In certain situations, such as when the pedestrian is in a hurry, he may choose to walk over the water puddle obstacle, as presented in Figure 1b.



Figure 1. The agent's planned path with different obstacle's danger: (**a**) Navigation around a deep hole; (**b**) navigation around a water puddle.

The risk perceived from the obstacle could depend on many factors. As in the ISO/IEC Guide 51 in Safety Aspect [36], risk is defined as the "combination of the probability of occurrence of harm and the severity of that harm". For instance, the danger of a lion should be remarkably high, but if that lion is kept inside a cage, its risk should be close to 0 as the chance of the lion interacting with others is low. In pedestrian navigation, the

danger from a human should be lower than a construction machine, for example. However, the risk coming from a pedestrian running at high speed, toward the pedestrian agent, should have a greater risk compared to the construction machine moving slowly on the side. Accordingly, we model our obstacle consisting of the following properties: *danger*, *size*, *direction*, *speed*, and *type* of obstacle. Similarly to ISO/IEC Guide 51, the risk from the obstacle is formulated by the obstacle's harm and its probability of collision perceived by the agent, which is discussed in more detail in Section 7.

All risk, danger level, and other obstacle's properties used in our study are perceived by only the pedestrian's cognitive system, which could be different from the actual information of the obstacle.

4.2. Model Overview

Figure 2 demonstrates the overview of our pedestrian path-planning model. The model consists of two components:

- 1. *Path-planning training*. This component instructs the agent to learn the basic navigations and collision avoidance within the environment using reinforcement learning. The details of the component are presented in Section 5.
- 2. *Point-of-conflict (POC) prediction.* This component simulates the agent's prediction of the collision with the obstacle. The process of prediction updates the input of the path-planning components and is handled before the planning process. Section 6 explains the prediction model in detail.



Figure 2. Overview of the pedestrian path-planning model.

For a reinforcement learning model, the design of the environment plays an important role. An environment that is similar to the real-world environment is usually unsuitable, as its complexity often leads to multiple problems. First of all, generally, the agent is not able to earn efficiently in a complex environment. For example, if there are many obstacles within the environments, they would create an extensive number of different states, leading to a considerably noisy training environment. To learn in an environment like that could be difficult for the agent, as the training would be quite unstable. There is also the overfitting problem. This means the agent could learn to navigate in the training environment, but its knowledge could not be transferred into unfamiliar environments.

This also partially corresponds to the human cognitive system. In the human cognitive system, the environment reflected in the human brain is usually a distorted topology of the real-world environment. Instead of using an entire map of the environment for local path-planning, only a portion of the visible environment is collected, depending on the cognitive system's planning horizon. As a result, in real life, pedestrians usually accomplish the path-planning process within a short distance from the current location to a determined destination. Once that location is reached, another path-planning process is carried out to a new location.

For that reason, our environment is designed to have a fixed area size, and also there is only one obstacle that may exist inside. A complex environment would be scaled down or divided into multiple parts, depending on the situation. There are several methods to realize this. For instance, in a study by Ikeda et al. [37], the agent would treat each component navigation part as its sub-goal when planning the route to a certain location. This would greatly help stabilize the training process while still is able to expand its applicability to new environments.

Consequently, our environment is modeled as illustrated in Figure 3. The area of the environment is 22 meters by 10 meters. The position of the agent is randomized between the coordinates (-5, -12) and (5, -12). The agent's current destination is randomized between the coordinates (-5, 10) and (5, 10).

The navigation path from the agent's position to its current destination consists of 10 component nodes whose coordinates' y values are predefined. The x coordinates of these nodes correspond to 10 outputs of the neural network. This will be presented in more detail in Section 5.2.





Figure 3. Path-planning environment modeling.

5. Path-Planning Training

The path-planning training utilizes reinforcement learning for the pedestrian agent to learn the navigation behavior. In reinforcement learning, the agent needs to continuously observe the states (usually partially) of the environment and subsequently take appropriate actions. These actions would be rewarded using the rewarding functions to let the agent know how good these actions are. Consequently, the following issues need to be addressed: modeling a learning environment, specifying the agent's observation of the environment and actions taken, and rewarding for the agent's actions.

5.1. Environment Modeling

The environment is modeled as previously presented in Section 4.2. For the learning task, the chance of an obstacle appearing in the environment is randomized in each training episode. In the case of the obstacle's appearance, its size is randomized between 0.5 and 2, and its danger level is randomized between 0 and 1. The entire environment might be scaled along its length (the y axis as in Figure 3) so that the agent could adapt its actions better to different real-life environments. Accordingly, in each training episode, the environment's scale will be randomized between 0.2 and 1.

We could finish the training episode and reset the environment every time the path is planned. However, this could cause the environment to be much noisier, which could lead to subsequent problems with the training of the neural network. An example of this is when the training environment has an obstacle in an episode, but no obstacle in the next one. In this case, even if the agent could not plan a path that successfully avoids the obstacle in the first episode, it is easy for the agent to do that in the second one and achieve a more favorable reward. This makes the agent accommodate the newer policy despite it may achieve worse results than the previous one. With a noisy environment like this, it would take much longer for the neural network to successfully converge the cumulative reward, and occasionally the policy could not be improved any further due to its inability to notice a better policy over the timesteps. To prevent this, we designed a different resetting mechanism for our environment. Instead of resetting immediately, we only reset the environment if the agent could plan a path without conflicting with the obstacle. Otherwise, the current states are kept so that the agent could try planning again. If the agent takes over a predefined number of steps without being able to plan a successful path, we also need to reset the environment, or the agent could be stuck in finding the appropriate policy.

5.2. Agent's Observations and Actions

In each step, the agent will observe the following states:

- x position of the agent's position;
- x position of the agent's destination;
- whether the obstacle appears in the environment or not;
- (if the obstacle is present) the obstacle's position, size, and risk; and
- the scale of the environment.

The y positions of the agent's position and destination do not need to be observed, as they are constantly determined in the modeling of the environment, as presented in Section 4.2.

For the learning task, the risk of the obstacle has the same value as its danger level. The purpose of this is to let the agent learns how to act differently with diverse values of risk. This does not teach the agent how to assess the risk from the obstacle's danger, however, as this would be carried out in the prediction task of the agent.

We need to specify the actions that the agent takes following its observations. In our model, these are a set of 10 values corresponding to the x coordinates of the navigation path. Each output is mapped to the x coordinate of the navigation nodes. Specifically, assuming the outputs of the network are $x_1, x_2, x_3...x_{10}$, the navigation of the agents would be the path through the following nodes: $(x_1, -10), (x_2, -8), (x_3, -6)... (x_{10}, 8)$, and finally, the agent's destination.

5.3. Rewarding Formulation

The rewards are used to tell the agent how good its taken actions are, which in this case are the planned path to the destination. Rewarding is an essential task in any reinforcement learning model. Different from rule-based models, rewarding is usually based on the results of the agent's actions or the effect of the agent's actions on the states. The rewarding formulation problem is equivalent to the task of instructing the pedestrian agent to aim at certain aspects. Consequently, to conduct a natural navigation behavior, we studied how humans consider the navigation as natural or not. As a result, we adopt the idea of human comfort, introduced by Kruse et al. [18]. The idea consists of several factors contributing to a simulated movement, which could make observing humans feel comfortable or human-like. Within the scope of our study, we choose to adopt the following factors for our rewarding mechanism:

- choosing the shortest path to the destination;
- avoiding frequently changing direction;
- following basic navigation rules and common-sense standards; and
- colliding with obstacles.

The first factor, which is also considered a decisive factor in many studies, is to plan the shortest path to the destination. While in real life navigation, human pedestrians may subconsciously aim at the shortest navigation time, they still consider shortest path to be the highest-ranking factor, as in a study conducted by Golledge [11]. As each rewarding factor correlates with the aspect that the human pedestrian is aiming at or wants to achieve, planning the shortest path would be formulized.

Consequently, we calculate the rewarding for this behavior by placing a negative reward corresponding to the sum of the squared length of each component path. This means if the path is longer, the agent would receive a larger penalty. This rewarding is formulated as follows:

$$\mathcal{R}_1 = -\lambda \sum_{i=0}^{11} \|p_i\|^2, \tag{5}$$

where λ is the environment's scale, and p_i is the vector of each component path.

Regarding the rewarding for changing direction, we only consider the changes in angle which are larger than 30 degrees. Any changes in angle which are smaller than this could be acceptable and are still considered natural. For this reason, we formulate the rewarding for this behavior by placing a penalty each time there is a large change of direction in the planned path as follows:

$$\mathcal{R}_{2} = -\sum_{i=0}^{10} \theta(angle(p_{i}, p_{i+1})), \qquad (6)$$

where $angle(p_i, p_j)$ is the angle value between the vectors p_i and p_j ; $\theta(x)$ is the Heaviside step function, specified by

$$\theta(x) = \begin{cases} 0, & \text{if } x < 0, \\ 1, & \text{if } x \ge 0. \end{cases}$$
(7)

As for the rewarding based on following basic navigation rules and common-sense standards, the rules may vary between different regions and cultures. From our observation, the following rules are applied in our study:

- 1. Following the flow of navigation by walking parallel to the sides.
- Walking on the left side of the road. While pedestrians are not required to strictly follow this, in real life, people still choose to follow this as a general guideline to avoid accidents. Similarly, in right-side walking countries, pedestrians would choose to walk on the right side of the road.
- 3. Avoiding getting close to the sides.

To define the appropriate rewarding formulations, the planned path of the agent is sampled into N values s_i with i ranges from 0 to N. The respective rewarding functions are calculated as follows:

$$\mathcal{R}_{3} = -\lambda \sum_{i=0}^{N} \theta \left(\| x_{pos}(s_{i+1}) - x_{pos}(s_{i}) \| - H_{1} \right), \tag{8}$$

$$\mathcal{R}_4 = -\lambda \sum_{i=0}^N \theta(-x_{pos}(s_i)),\tag{9}$$

$$\mathcal{R}_{5} = -\sum_{i=0}^{N} \theta(\|x_{pos}(s_{i})\| - H_{2}),$$
(10)

where $x_{pos}(s_i)$ function returns the *x* coordinate of the point s_i .

The value H_1 in Equation (8) is the threshold value for the difference in *x* coordinates that the agent could make in each sample navigation part. The smaller difference in *x* coordinates of the navigation produces the path that is more parallel to the sides. In our model, with N = 200, H_1 is given a value of 0.4. In addition, our model will put a negative reward on the agent whenever its *x* coordinate is less than 0 as in Equation (9), meaning the agent is at the left side of the road. Regarding Equation (10), as suggested in other studies [14,38], the agent would stay approximately 0.5 meters from the walls to avoid possible accidents. In our model, the navigation path has a width of 10 meters; therefore, the value H_2 is set to 4.5 so that, when the agent's position has an x coordinate higher than 4.5 or less than -4.5, it would receive a negative reward.

Lastly, with respect to collision avoidance, the agent needs to keep a certain distance from the obstacle. The highest risk would seemingly be at the center of the obstacle, and the risk gradually decreased with longer distance. However, once the agent has reached a certain distance with the obstacle, any further than this would be unnecessary. For example, if the pedestrian in real life would like to avoid stepping on a puddle, as long as the navigation path does not conflict with the puddle, it does not matter if the path needs to be much further away from it. Because of this reason, we formulate our rewarding for the collision avoidance behavior as follows:

$$\mathcal{R}_{6} = \sum_{i=0}^{N} \begin{cases} \frac{\delta(s_{i}, obs)}{R_{obs}^{2}} r^{2}, & \text{if } \delta(s_{i}, obs) \leq 0, \\ 0.01 r^{2}, & \text{if } \delta(s_{i}, obs) > 0, \end{cases}$$
(11)

with $\delta(s_i, obs) = d(s_i, obs)^2 - R_{obs}^2$, where $d(s_i, obs)$ is the distance from the sampled position s_i and the obstacle; R_{obs} is the radius of the obstacle's area; and r is the risk from the obstacle. In the training task, r has the value of obstacle's danger, as presented in Section 5.2.

The resulted reward given to the agent's policy is the sum of all components rewards multiplied by the corresponding coefficients:

$$\mathcal{R} = \sum_{i=1}^{6} \mathcal{R}_i \kappa_i, \tag{12}$$

where κ_i is the coefficient of the appropriate reward.

Each variation of a set of κ_i results in a different personality in the agent's path planning process. In real life, different people have different priorities in how the navigation path is formed. For example, to simulate the pedestrian who prioritizes following the regulations, the coefficient for \mathcal{R}_4 , walking on the left side, should be higher. Similarly, to replicate the behavior of a cautious pedestrian, the model should use a higher value for \mathcal{R}_6 , obstacle avoidance rewarding.

6. Point-of-Conflict Prediction

To accurately simulate the navigation of a pedestrian, the incorporation of the prediction is necessary. This prediction might not be accurate, as humans in real life usually make inaccurate predictions. As a result, the prediction process in our model also focuses on replicating a similar prediction mechanism.

We proposed a concept called *point-of-conflict* (POC), a location within the environment that the agent thinks could collide with the obstacle or at the predicted position of the obstacle when it is closest to the agent [31]. Even in the case of a low chance of collision (e.g., when the agent and the obstacle are navigating on two sides of the road), a POC is still predicted. The motivation is that, when the human has already learned the appropriate prediction method, the prediction process would occur in most cases. This would happen naturally inside human cognition without much reasoning.

When the prediction task is handled, the agent would use the information from the POC instead of the actual obstacle in the path-planning training task as introduced in Section 4. The location of the POC will be predicted by the agent depending on the obstacle's type, which will be demonstrated in more detail subsequently. Figure 4 illustrates the path-planning process of the agent after the prediction task is utilized.

The position of the POC depends on the type of obstacle. For example, if the obstacle is *stationary*, the POC's position should be the same as the position of the obstacle. Apart from stationary obstacle, we define two other obstacle's types: *single diagonal movement* obstacle, and *pedestrian* obstacle. Each type of obstacle has a different method of calculating the POC's position. To simplify the prediction of the POC, we assume the agent has the information of the obstacle's speed and heading direction. It is worth noting that the heading direction is the direction toward the obstacle's destination instead of its current orientation, but could turn in another direction for various reasons (e.g., steering to the left-hand side). There have been several studies addressing the problem [24,25,39], which could be applicable to our study.



Figure 4. Obstacle avoidance with point-of-conflict.

6.1. Single Diagonal Movement Obstacle

A single diagonal movement obstacle is an obstacle that is mostly moving in one direction and with a uniform speed. Some examples of this obstacle's type are a pedestrian crossing the environment or a road construction machine moving slowly on the sidewalk. This type of obstacle does not include a vehicle moving at normal speed. In that case, the pedestrian agent should exclude its navigation area from the model's environment, as it would be too dangerous to navigate inside that area.

Figure 5 illustrates the POC prediction process in the case of a single diagonal movement obstacle. In order to specify the area of the POC, we need to figure the approximate time until the obstacle is getting close. As the prediction process is carried out before the path-planning task, we could only estimate this using the agent's general direction toward its destination. The calculation for this approximate time is formulated as:

$$t = \frac{\delta}{v_{agent}\cos\theta_a + v_{obs}\cos\theta_o} \quad \text{if} \ (v_{agent}\cos\theta_a + v_{obs}\cos\theta_o) > 0, \tag{13}$$

where δ is the distance in y coordinate between the agent and the obstacle, and v_{agent} and v_{obs} are the velocity of the agent and the obstacle; θ_a is the agent's direction angle relative to the upward vertical axis, and θ_o is the obstacle's direction angle relative to the downward vertical axis.





As a result, the POC's position (x_{POC}, y_{POC}) is specified as follows:

$$(x_{POC}, y_{POC}) = (x_{obs}, y_{obs}) + t \lambda v_{obs} \hat{\mathbf{e}}_{obs},$$
(14)

where (x_{obs}, y_{obs}) is the position of the obstacle, $\hat{\mathbf{e}}_{obs}$ is the unit vector having the direction of the obstacle, and λ is the environment's scale as presented in Section 5.1.

If the $(v_{agent} \cos \theta_a + v_{obs} \cos \theta_o) \leq 0$, it is unlikely for the agent to collide with the obstacle. In this case, the POC is omitted in the planning task of the model. In addition, if the calculated POC's position is outside the range of the agent's environment, the POC is also ignored in our model.

6.2. Pedestrian Obstacle

Pedestrian obstacles are usually the most common type of obstacle that could interact with the pedestrian agent. However, the definition of pedestrian obstacle in our study does not include a pedes-trian crossing the environment, as it is considered as a single diagonal movement ob-stacle discussed above. To predict the position of a POC, the agent needs to specify the navigation path that the obstacle might take. While the model for single diagonal movement obstacle could also be adopted in this case, its result would be fairly inaccurate, and more importantly, does not conform to the human predictive system.

For that reason, we have proposed a unique method of predicting the POC for a pedestrian obstacle. Firstly, to define the predicted navigation path of the obstacle, we utilized our existing reinforcement learning path-planning model. By doing this, the predicted navigation path would have the same advantage as our reinforcement learning model and, therefore, could replicate a realistic navigation path. Subsequently, the POC will be specified on that navigation path, using the velocity of the agent and the obstacle. Figure 6 represents the POC's prediction in the case of a pedestrian obstacle.

Before the obstacle's navigation path could be constructed, its estimated destination needs to be determined. This could be achieved by projecting the obstacle's orientation to the end of its navigation environment (separate from the agent's environment). The projected destination $(x_D^{(obs)}, y_D^{(obs)})$ could be formulated as follows:

$$\left(x_D^{(obs)}, y_D^{(obs)}\right) = \left(x_{obs} - \frac{\lambda \ L \ v_x}{v_y}, y_{obs} - \lambda \ L\right),\tag{15}$$

where (x_{obs}, y_{obs}) is the obstacle's position, (v_x, v_y) is the orientation vector of the obstacle, and *L* is the length of the obstacle's environment. In our proposed model's environment, *L* has a length of 22 m.



Figure 6. Point-of-conflict of a pedestrian obstacle.

The observations of the obstacle consist of the obstacle's position and its projected destination. The observations do not include the observation of an obstacle (i.e., the pedestrian agent in the obstacle's environment) for two reasons. The first reason is that the POC prediction happens before the path-planning process; therefore, the obstacle cannot specify the agent's path. Trying to specify the paths of the agent and the obstacle at the same time would certainly cause conflict. Another reason is related to the process of human thinking in real life. When a pedestrian is predicting the navigation path of the obstacle, he would not consider himself as an obstacle, but rather trying to navigate in a way that could avoid a collision.

The RL model used in our obstacle's path-planning process is the same one used by the pedestrian agent. The reason is that usually a person often thinks other people would act the same way, for example, navigating the same way as he would do. Alternatively, the obstacle could use the mean RL model from multiple training.

The predicted position of the POC could be subsequently determined using the scale between the velocities of the agent and the obstacle. The calculation of the POC's y coordinate is formulated as follows:

$$y_{POC} = y_{obs} - \delta \frac{v_{obs}}{v_{agent} + v_{obs}},\tag{16}$$

where v_{agent} and v_{obs} are the velocity of the agent and the obstacle, respectively; δ is the difference in the y axis between the agent and the obstacle.

Finally, the location of the predicted POC is specified by the point on the pedestrian's navigation path at the y_{POC} value in the y axis.

7. Risk Assessment

With the point-of-conflict prediction, the pedestrian agent would observe the POC's position and assess its risk instead of using the obstacle's position and danger level. As previously discussed in Section 4.1, risk is calculated based on the obstacle's harm and the probability of collision, which is formulated as

$$r = harm \cdot P,$$
 (17)

where *r* is the risk, *harm* is the possible harm caused by the obstacle, and *P* is the probability of collision with the obstacle. To conform to the agent's observations in our reinforcement learning model, all values *r*, *harm* and *P* have a range from 0 to 1.

To estimate the probability of collision, we need to specify the proximity of the POC's position to the navigation path. Because the risk assessment is carried out before the path-planning task, the navigation path could be approximated as a straight line from the agent's position to its current destination.

The distance from the POC's position (x_{POC} , y_{POC}) and agent's estimated navigation line is calculated by:

$$\delta_{POC} = \frac{|Ax_{POC} + By_{POC} + C|}{\sqrt{A^2 + B^2}},$$
(18)

with $A = y_D - y_a$, $B = x_a - x_D$, and $C = (x_D - x_a)y_a - (y_D - y_a)x_a$,

where (x_a, y_a) and (x_D, y_D) are the coordinates of the agent and obstacle, respectively. The collision probability *P* is highest when $\delta_{POC} = 0$, and gradually decline with higher δ_{POC} . *P* is formulated in our model as follows:

$$P = \frac{M}{\delta_{POC} + M'}$$
(19)

where *M* is a distance constant. For instance, in our implementation, we adopted using M = 3, which means when δ_{POC} is 3 m, the collision probability *P* would be at 0.5.

To estimate the harm from the obstacle, we use the obstacle's danger level and also its speed, as the speed could also impact the harm caused by the obstacle [40]. As an example, the risk observed from a person running at a high speed should be higher than the risk observed from a person walking at a normal speed toward the agent, even when the perceived danger from the two persons are the same.

Arguably, the obstacle's speed could also contribute to the probability of the agent's avoidance. However, because the agent's navigation path was not formed at the current process, the avoidance probability is unspecified. Assuming the capability of avoidance of the pedestrian agent is constant, the obstacle's speed should not affect the probability of collision *P*.

In the case that the obstacle's speed is irrelevant, such as a static obstacle, the harm of the obstacle is equivalent to its danger level.

Otherwise, we adopt using the concept of kinetic energy to estimate the harm of the obstacle, similar to how humans feel the impact of a moving object when it hits. As a result, harm is formulated as

$$harm = max \left(1, danger \left(1 + \gamma \frac{K_{obs}}{K_{normal}} \right) \right), \tag{20}$$

where K_{obs} is the kinetic energy of the moving obstacle, K_{normal} is the kinetic energy of an object moving at a normal speed, and γ is the discount value.

Considering $K = \frac{1}{2}mv^2$, the harm of a moving obstacle could be formulated as follows:

$$harm = max \left(1, danger \left(1 + \gamma \left(\frac{v_{obs}}{v_{normal}} \right)^2 \right) \right), \tag{21}$$

where v_{normal} is the average speed of a moving object which could be perceived as normal. In several studies [41–43], v_{normal} is specified to be approximately 1.31 m/s.

Finally, with the harm value calculated, the risk of the obstacle is formulated by Equation (17). This risk value together with the POC's position specified in Section 6, is used in the agent's observations in the pedestrian reinforcement learning model. More specifically, regarding the obstacle's properties, the pedestrian agent will observe the POC's relative coordinates, the obstacle's size, and the risk formulated in this section. Consequently, the formulated reward in Equation (11) is updated (which, in the training process, uses the same value of the obstacle's danger for its risk). This could result in a more precise navigation path in the similar way the path is planned by a human pedestrian.

8. Results

The model of our study was implemented using the real-time development platform Unity. The source code is available at https://github.com/trinhthanhtrung/unitypedestrian-rl (accessed on 30 April 2021), by opening the scene *PathPlanningTask* within the *Scene* folders. Figure 7 presents a screenshot of our implemented application.



Figure 7. Path-planning task implementation screenshot.

For the training task of the pedestrian agent, we adopted the reinforcement learning library ML-Agents [44], which acts as a communicator between Unity and Python machine learning code. In each training episode, the information of the model, consisting of the agent's observations and actions, and the cumulative reward value, is sent to Python. The information is subsequently used for training the agent's policy in a neural network using the PPO algorithm, then the updated policy will be sent back to the pedestrian agent.

Because the environment's states are moderately noisy, it is recommended to train the agent with a large batch size. Furthermore, multiple instances of the same training environment are created to speed up the training process. We have been able to successfully get the cumulative reward to converge after three million steps with a learning rate of 1.5×10^{-3} . For a smoother navigation path, we used the mean of the agent's actions in multiple episodes of the same environment's state. Figure 8 shows the statistics of the training process in TensorBoard.



Figure 8. Cumulative reward statistics.

Figure 9 shows the planned path of the agent in different situations in our implementations. In these figures, the actor model at the bottom is the pedestrian agent, the red point on the top is the agent's destination, the black circle represents the predicted POC by the agent, and the red circle (covered by the POC in (b) and (c)) is the current obstacle.



Figure 9. Agent's planned path in different situations: (**a**) no obstacle; (**b**) with a static obstacle with a low danger level; (**c**) with a static obstacle with a high danger level; (**d**) with an obstacle moving straight in one direction away from the agent (**e**) with an obstacle moving straight in one direction toward the agent; (**f**) with a pedestrian obstacle.

Figure 10 shows how the implementation of our model compares to SFM. In each figure, our implementation is on the left and the SFM implementation is on the right (darker environment).



Figure 10. Our model (left) compared with SFM (right) in different situations: (**a**) no obstacle; (**b**) with a static obstacle; (**c**) with a moving obstacle; (**d**) with a pedestrian obstacle.

From observation, in Figure 9, the agent could be able to plan a sufficiently realistic path to the destination, while also considering the rules and following common conventions like walking to the left side and naturally changing direction. This can be seen in the situation (a), where the agent has chosen to walk on the left side of the road and gradually move toward the destination when needed, instead of walking straight to the destination. Although the planned path was constructed from the outputs of the neural networks, it is shown to be remarkably stable. The agent has also shown its capability to avoid the obstacle, as the planned path does not collide with the obstacle or its prediction in most situations.

Furthermore, its planned path also seems to be adapted to the risk from the obstacle. This is observable by comparing the paths planned by the agent in (b) and (c). We implemented the obstacles to have the same properties in both situations; however, the obstacle in (c) has a much higher danger level than the obstacle in (b). The result is that, in Figure 9b, the agent only almost avoided colliding with the obstacle, while in (c), the agent chose a path that steers much further away from the obstacle than in (b). This resembles actual human thinking when planning a navigation path where there is a dangerous obstruction on the road.

Figure 9d,e demonstrate how the agent adopts the prediction process into path planning. In both situations, the agent planned the path to avoid the possible point-of-conflict instead of the actual current position of the obstacle, which is similar to how an adult person plans the navigation path. However, the obstacle in (e) was moving at a higher speed; therefore, the risk perceived from the obstacle is higher. As indicated in the figure, the *harm* value calculated in (d) was 0.61, compared to the higher *harm* value calculated in (e) at 0.81. Additionally, the difference in the POC's position causes the change in their

probability estimations, which are 0.77 in (d) and 0.92 in (e). The increased probability also contributes to a higher resulted risk specified in (e) situation (0.745 in (e), compared to 0.470 in (d)). This was reflected in the navigation path by the pedestrian agent, as it is shown that the agent could plan the path to quickly avoid the possible collision. The risk formulation in (d) and (e) has shown that it could be greater to or less than the obstacle's danger level (0.6 in both situations) depending on the speed of the obstacle, consistent with human thinking in real life.

Figure 9f presents the planned path of the pedestrian agent in the case of another pedestrian obstacle. In this case, the obstacle's path was formed to predict the possible collision, which also resembles the human thinking process when a pedestrian trying to avoid another person while walking.

9. Discussion

The implementations have shown that the agent in our model could develop a relatively natural path compared to how humans plan the path right before navigation. As each individual thinks and plans differently, the planned path by the agent may not be identical to a specific person's thinking. However, this planned path could still be seen as natural or human-like thanks to several similar traits found in the result, such as smooth navigation and following common regulations. This also indicates that by providing the appropriate rewarding formulations, the reinforcement learning agent could develop a behavior similar to the human decision-making process, thus partly confirming the hypothesis raised by other studies [45]. By supplementing and refining the rewarding formulation, a more realistic and natural navigation could be replicated.

Nonetheless, admittedly with enough complex rule sets, a rule-based model could achieve a similar result as our model. However, it could be difficult to develop the rule sets for extended states of the environment, while with reinforcement learning, the agent could adapt well to an unfamiliar environment. Another advantage of utilizing reinforcement learning in the path-planning model is that a reinforcement learning model always retains a slight unpredictability, providing some sense of the same unpredictability in human nature, which makes the navigation path more believable. On the other hand, that could also result in unknown outcomes in unforeseeable situations.

When comparing to the Social Force Model's implementation, it is apparent that the two implementations take distinctive approaches, as this could be seen in Figure 10. For the human's local path-planning task, our model has shown a better result, mostly because humans rarely generalize the idea of "force" when planning the path in real life. The inaccuracy of the SFM's implementation is more noticeable in the case in which the pedestrian needs to navigate from one side to the other, as the SFM agent tends to disrupt the flow of the navigation path. The lack of a prediction method also makes SFM less ideal to realize the human's path-planning process. As a result, the path planned by the SFM agent heads straight to the destination without considering obvious possible collisions within the navigation. The path only avoids the walls and the obstacles when it is at a certain distance from those obstructions. Nonetheless, in the case when planning is difficult, like in a crowded environment for example, or for people who rarely plan before navigating, the SFM model could be sufficient.

Despite having shown a relatively natural path, assessing the model's resemblance to the human solutions is a challenging task since the path-planning process only happens in the thoughts of pedestrians. This makes evaluating the human-likeness of the result difficult, which is the major limitation of our study. We have considered several mechanisms in human cognition of assessing human likeness in pedestrian behavior. The problem is, when observing the movement, humans do not have the exact criteria to determine specific behavior is human-like or not. Instead, the human conscious and subconscious recognition processes will subjectively evaluate the movement by matching it with existing sensory data. Occasionally, even a more realistic behavior may trigger the uncanny effect, consequently leading humans to negate the human likeness of that behavior. As a result, to overcome this limitation, more insight into the human cognitive system needs to be carefully addressed.

The risk assessment seems to have contributed to the model's reasonable result. This corresponds to actual human pedestrians when perceiving different properties from an obstruction. The observable result seems to resemble how humans would perceive risks from the obstructions; however, as aforementioned, to estimate its resemblance to the task performed by humans could be demanding.

It should be noted that, in this paper, only the path-planning task happening inside a human pedestrian's thinking before navigating is replicated. This path could be different from the actual path taken by the pedestrian. When following the planned path, the agent should be able to interact with the surrounding obstructions, especially when the obstructions are not navigating as predicted. In our future work, the pedestrian interacting problem for those situations will be addressed to further improve the movement of the pedestrian agent.

10. Conclusions

In this study, we have developed a novel pedestrian path-planning model using reinforcement learning while considering the prediction of the obstacle's movement and the risk from the obstacle. The model consists of two main components: a reinforcement learning model to train the agent the behavior to navigate in an environment and interact with the obstacle, and a point-of-conflict prediction model to form the estimated interacting position of the agent with the obstacle. Both components of the model acknowledge the risk assessment of the obstacle to provide corresponding results. The implementation results of our model have demonstrated a sufficiently realistic navigation behavior in many situations, resembling the path-planning process of a human pedestrian. Nevertheless, the problem of quantitatively evaluating the human likeness of the pedestrian agent's path in our model remains a major challenge in our study.

Author Contributions: Conceptualization, T.-T.T. and M.K.; methodology, T.-T.T. and M.K.; software, T.-T.T.; validation, T.-T.T. and M.K.; writing—original draft preparation, T.-T.T.; writing—review and editing, T.-T.T. and M.K.; supervision, M.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were created in this study. This data can be found here: https://github.com/trinhthanhtrung/unity-pedestrian-rl/releases (accessed on 30 April 2021).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

- RL Reinforcement Learning
- PPO Proximal Policy Optimization
- POC Point-of-conflict
- SFM Social Force Model

References

- 1. Schadschneider, A.; Klingsch, W.; Klüpfel, H.; Kretz, T.; Rogsch, C.; Seyfried, A. Evacuation dynamics: Empirical results, modeling and applications. *arXiv* **2008**, arXiv:0802.1620.
- Foltête, J.C.; Piombini, A. Urban layout, landscape features and pedestrian usage. Landsc. Urban Plan. 2007, 81, 225–234. [CrossRef]

- 3. Zacharias, J. Pedestrian behavior pedestrian behavior and perception in urban walking environments. *J. Plan. Lit.* **2001**, *16*, 3–18. [CrossRef]
- 4. Rasouli, A.; Tsotsos, J.K. Autonomous vehicles that interact with pedestrians: A survey of theory and practice. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 900–918. [CrossRef]
- 5. Sewalkar, P.; Seitz, J. Vehicle-to-pedestrian communication for vulnerable road users: Survey, design considerations, and challenges. *Sensors* **2019**, *19*, 358. [CrossRef]
- 6. Carton, D.; Nitsch, V.; Meinzer, D.; Wollherr, D. Towards Assessing the Human Trajectory Planning Horizon. *PLoS ONE* 2016, 11, e0167021. [CrossRef]
- Andreev, S.; Dibbelt, J.; Nöllenburg, M.; Pajor, T.; Wagner, D. Towards realistic pedestrian route planning. In Proceedings of the 15th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2015), Patras, Greece, 17 September 2015; Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik: Dagstuhl, Germany, 2015; doi:10.4230/OASIcs.ATMOS.2015.1. [CrossRef]
- Zhang, L.; Liu, M.; Wu, X.; AbouRizk, S.M. Simulation-based route planning for pedestrian evacuation in metro stations: A case study. *Autom. Constr.* 2016, 71, 430–442. [CrossRef]
- Teknomo, K.; Millonig, A. A Navigation Algorithm for Pedestrian Simulation in Dynamic Environments. In Proceedings of the 11th World Conference on Transport Research World Conference on Transport Research Society, Berkeley, CA, USA, 24–28 June 2007.
- 10. Reitter, D.; Lebiere, C. A cognitive model of spatial path-planning. Comput. Math. Organ. Theory 2010, 16, 220–245. [CrossRef]
- Golledge, R.G. Path selection and route preference in human navigation: A progress report. In Proceedings of the International Conference on Spatial Information Theory, L'Aquila, Italy, 4–8 September 1995; Springer, Berlin/Heidelberg, Germany, 1995; pp. 207–222._14. [CrossRef]
- 12. Melchior, P.; Orsoni, B.; Lavialle, O.; Poty, A.; Oustaloup, A. Consideration of obstacle danger level in path planning using *A** and fast-marching optimisation: Comparative study. *Signal Process.* **2003**, *83*, 2387–2396. [CrossRef]
- 13. Caramuta, C.; Collodel, G.; Giacomini, C.; Gruden, C.; Longo, G.; Piccolotto, P. Survey of detection techniques, mathematical models and simulation software in pedestrian dynamics. *Transp. Res. Procedia* **2017**, *25*, 551–567. [CrossRef]
- 14. Helbing, D.; Molnar, P. Social force model for pedestrian dynamics. *Phys. Rev. E* 1995, 51, 4282. [CrossRef]
- 15. Farina, F.; Fontanelli, D.; Garulli, A.; Giannitrapani, A.; Prattichizzo, D. Walking ahead: The headed social force model. *PLoS* ONE **2017**, *12*, e0169734. [CrossRef]
- 16. Seyfried, A.; Steffen, B.; Lippert, T. Basics of modelling the pedestrian flow. *Phys. A Stat. Mech. Its Appl.* **2006**, *368*, 232–238. [CrossRef]
- 17. Henderson, L.F. On the fluid mechanics of human crowd motion. Transp. Res. 1974, 8, 509-515. [CrossRef]
- Kruse, T.; Pandey, A.K.; Alami, R.; Kirsch, A. Human-aware robot navigation: A survey. *Robot. Auton. Syst.* 2013, 61, 1726–1743. [CrossRef]
- 19. Cohen, J.D.; McClure, S.M.; Yu, A.J. Should I stay or should I go? How the human brain manages the trade-off between exploitation and exploration. *Philos. Trans. R. Soc. B Biol. Sci.* 2007, *362*, 933–942. [CrossRef] [PubMed]
- Martinez-Gil, F.; Lozano, M.; Fernández, F. Multi-agent reinforcement learning for simulating pedestrian navigation. In Proceedings of the International Workshop on Adaptive and Learning Agents, Taipei, Taiwan, 2 May 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 54–69._4. [CrossRef]
- Trinh, T.T.; Vu, D.M.; Kimura, M. A pedestrian path-planning model in accordance with obstacle's danger with reinforcement learning. In Proceedings of the 2020 the 3rd International Conference on Information Science and System, Cambridge, UK, 19–22 March 2020; pp. 115–120. [CrossRef]
- 22. Kretzschmar, H.; Spies, M.; Sprunk, C.; Burgard, W. Socially compliant mobile robot navigation via inverse reinforcement learning. *Int. J. Robot. Res.* **2016**, *35*, 1289–1307. [CrossRef]
- Møgelmose, A.; Trivedi, M.M.; Moeslund, T.B. Trajectory analysis and prediction for improved pedestrian safety: Integrated framework and evaluations. In Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV), Seoul, Korea, 28 June–1 July 2015; pp. 330–335. [CrossRef]
- Goto, K.; Kidono, K.; Kimura, Y.; Naito, T. Pedestrian detection and direction estimation by cascade detector with multi-classifiers utilizing feature interaction descriptor. In Proceedings of the 2011 IEEE Intelligent Vehicles Symposium (IV), Baden-Baden, Germany, 5–9 June 2011; pp. 224–229. [CrossRef]
- 25. Dominguez-Sanchez, A.; Cazorla, M.; Orts-Escolano, S. Pedestrian movement direction recognition using convolutional neural networks. *IEEE Trans. Intell. Transp. Syst.* 2017, *18*, 3540–3548. [CrossRef]
- Yi, S.; Li, H.; Wang, X. Pedestrian behavior understanding and prediction with deep neural networks. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp 263–279._16. [CrossRef]
- Quintero, R.; Almeida, J.; Llorca, D.F.; Sotelo, M.A. Pedestrian path prediction using body language traits. In Proceedings of the 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, USA, 8–11 June 2014; pp. 317–323. [CrossRef]
- Schneider, N.; Gavrila, D.M. Pedestrian path prediction with recursive bayesian filters: A comparative study. In Proceedings of the German Conference on Pattern Recognition, Saarbrücken, Germany, 3–6 September 2013; Springer: Berlin/Heidelberg, Germany, 2013; pp. 174–183._18. [CrossRef]

- Karasev, V.; Ayvaci, A.; Heisele, B.; Soatto, S. Intent-aware long-term prediction of pedestrian motion. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 2543–2549. [CrossRef]
- Ziebart, B.D.; Ratliff, N.; Gallagher, G.; Mertz, C.; Peterson, K.; Bagnell, J.A.; Hebert, M.; Dey, A.K.; Srinivasa, S. Planning-based prediction for pedestrians. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 3931–3936. [CrossRef]
- 31. Trinh, T.T.; Vu, D.M.; Kimura, M. Point-of-Conflict Prediction for Pedestrian Path-Planning. In Proceedings of the 12th International Conference on Computer Modeling and Simulation, Brisbane, Australia, 22–24 June 2020; pp. 88–92. [CrossRef]
- 32. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; Cambridge University Press: Cambridge, UK, 1999; doi:10.1017/S0263574799271172. [CrossRef]
- 33. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv* 2017, arXiv:1707.06347.
- 34. Ampofo-Boateng, K.; Thomson, J.A. Children's perception of safety and danger on the road. *Br. J. Psychol.* **1991**, *82*, 487–505. [CrossRef] [PubMed]
- Chung, W.; Kim, S.; Choi, M.; Choi, J.; Kim, H.; Moon, C.B.; Song, J.B. Safe navigation of a mobile robot considering visibility of environment. *IEEE Trans. Ind. Electron.* 2009, 56, 3941–3950. [CrossRef]
- ISO/IEC Guide 51: 2014. Safety Aspects—Guidelines for their Inclusion in Standards. 2014. Available online: https://www.iso.org/ standard/53940.html (accessed on 30 April 2021).
- Ikeda, T.; Chigodo, Y.; Rea, D.; Zanlungo, F.; Shiomi, M.; Kanda, T. Modeling and prediction of pedestrian behavior based on the sub-goal concept. *Robotics* 2013, 10, 137–144.
- 38. Bonneaud, S.; Warren, W.H. A behavioral dynamics approach to modeling realistic pedestrian behavior. In Proceedings of the 6th International Conference on Pedestrian and Evacuation Dynamics, Zurich, Switzerland, 6–8 June 2012; pp. 1–14.
- Asahara, A.; Maruyama, K.; Sato, A.; Seto, K. Pedestrian-movement prediction based on mixed Markov-chain model. In Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic, Information Systems, Chicago, IL, USA, 1–4 November 2011; pp. 25–33. [CrossRef]
- 40. Stoker, P.; Garfinkel-Castro, A.; Khayesi, M.; Odero, W.; Mwangi, M.N.; Peden, M.; Ewing, R. Pedestrian safety and the built environment: A review of the risk factors. *J. Plan. Lit.* **2015**, *30*, 377–392. [CrossRef]
- 41. Robin, T.; Antonini, G.; Bierlaire, M.; Cruz, J. Specification, estimation and validation of a pedestrian walking behavior model. *Transp. Res. Part B Methodol.* **2009**, 43, 36–56. [CrossRef]
- 42. Elliott, J.R.; Simms, C.K.; Wood, D.P. Pedestrian head translation, rotation and impact velocity: The influence of vehicle speed, pedestrian speed and pedestrian gait. *Accid. Anal. Prev.* **2012**, *45*, 342–353. [CrossRef] [PubMed]
- 43. Goh, P.K.; Lam, W.H. Pedestrian flows and walking speed: A problem at signalized crosswalks. ITE J. 2004, 74, 28.
- 44. Juliani, A.; Berges, V.P.; Teng, E.; Cohen, A.; Harper, J.; Elion, C.; Goy, C.; Gao, Y.; Henry, H.; Mattar, M.; et al. Unity: A general platform for intelligent agents. *arXiv* **2018**, arXiv:1809.02627.
- 45. Botvinick, M.; Weinstein, A.Model-based hierarchical reinforcement learning and human action control. *Philos. Trans. R. Soc. B Biol. Sci.* 2014, *369*, 20130480. [CrossRef]