

Automatic Synthesis of Virtual Wheelchair Training Scenarios

Wanwan Li*

Javier Talavera*

Amilcar Gomez Samayoa*

Jyh-Ming Lien

Lap-Fai Yu

George Mason University

*Equal Contributors



Figure 1: We have developed an optimization-based approach to automatically generate the scenarios (left) used for wheelchair training with different user-specified levels of difficulty. We test our generated scenario by developing a training program using VR trackers attached to the wheels of a stationary wheelchair to simulate motion in virtual space(right).

ABSTRACT

In this paper, we propose an optimization-based approach for automatically generating virtual scenarios for wheelchair training in virtual reality. To generate a virtual training scenario, our approach automatically generates a realistic furniture layout for a scene as well as a training path that the user needs to go through by controlling a simulated wheelchair. The training properties of the path, namely, its desired length, the extent of rotation, and narrowness, are optimized so as to deliver the desired training effects. We conducted an evaluation to validate the efficacy of the proposed virtual reality training approach. Users showed improvement in wheelchair control skills in terms of proficiency and precision after receiving the proposed virtual reality training.

Index Terms: Virtual Reality—Modeling and Simulation—Wheelchair Training Simulator

1 INTRODUCTION

Wheelchairs are used for those who have difficulty walking on their own. This can be due to illness, injury, or disability. Every year, there are nearly two million new wheelchair users in the U.S. alone [30]. Learning how to properly use a wheelchair is crucial because it will frequently become the primary source of transportation for these new wheelchair users. A skilled wheelchair user is able to properly maneuver around sharp turns and pass through narrow pathways in a timely manner without running into obstacles.

There are currently various methods [3, 8, 21] to learn how to use a wheelchair:

- **Human coach** - This is in-person guidance to teach users how to use a wheelchair. This method has shown success when taught in one-on-one training environments [3, 21].

- **Self-teaching through trial and error** - A user maneuvers a wheelchair on their own and rides the wheelchair with no guidance, expecting to improve. This method has not shown significant improvement within a short period of time [8].
- **Instruction video/manual** - User follows instructions via video or manual on the proper usage of a wheelchair by reading and watching. This is time-consuming and its efficacy depends on the user's patience.

All of these wheelchair training methods share a common flaw which is that they make it difficult to practice in multiple different environments. We attempt to solve this by using virtual reality simulation. As shown in Figure 1, our simulation allows users to train themselves in multiple environments with customized scenarios automatically generated through an optimization process.

During the VR training, the user requires no previous experience with wheelchair usage. Through the VR training scenarios synthesized by our technical approach, users can improve their proficiency in controlling the wheelchair in real-life. The goal of our approach is to release the users from the limitations of the current environments they are facing by automatically generating training scenarios in virtual reality. The major contributions of our work include the following:

- Proposing a novel optimization-based approach to automatically synthesize the training scenarios for improving the wheelchair skills with different training goals.
- Developing a physics-based wheelchair simulation that is highly immersive to enable users to maneuver the virtual wheelchair in the same way as they maneuver the real wheelchair.
- Conducting a user study to validate the effectiveness of the training scenarios synthesized through our optimization-based approach.

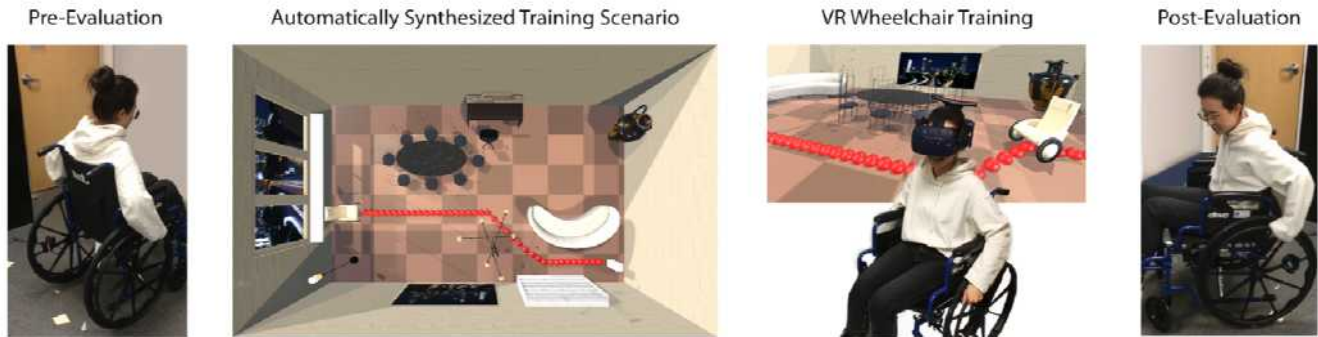


Figure 2: Overview of our approach. This figure shows the approach of our research. Pre-evaluation: before the VR training, the user completes a predefined real-world task in the laboratory during which the user navigate the wheelchair along a specified path with their performance recorded. Automatically synthesized training scenario: VR training scenarios that are automatically generated will be used for the VR training. VR wheelchair training: We have developed a physics-based simulation that is capable of mapping the motion of a real wheelchair into the virtual space by tracking the wheels’ rotation with two HTC VIVE VR trackers. With the optimized scenarios loaded into this wheelchair VR simulator, the users are placed in the VR headset and sat down in the raised wheelchair to keep the wheels off the ground. Post-evaluation: after the VR training, the user is asked to finish the same real-world task in the pre-evaluation stage and the performances are compared.

2 RELATED WORK

Traditional Wheelchair Training. Human coach training is the prevalent training method. When community-dwelling veterans with spinal cord injuries were individually trained in a home environment, they showed significant improvement to their wheelchair skills [21]. However, when novice subjects used a wheelchair without guidance, there was no significant improvement over time [8]. While the user may not receive explicit guidance, the VR training allows the user to encounter the same scenarios they would get with an instructor and more. Other training methods such as long term low-intensity wheelchair training [7] indicate that users’ mechanical efficiency increased and metabolic cost decreased significantly over time. However, this method lacks variation which could lead to boredom and decreased efficiency. Randomized clinical trials show that formal wheelchair training sessions result in significantly greater improvement than standard rehabilitation programs [26]. Our goal is to create an informal training program that is just as effective with more flexibility in terms of time, place and efficiency.

Virtual Reality Training and Simulation. There are many successful pieces of research about virtual reality training simulations that have been developed in the past in different fields of research areas including virtual reality surgical simulation and training [13, 15–17, 23, 37], virtual reality driving training programs [5, 10, 24] and task-driven virtual reality trainings such as industrial maintenance trainings [14], mining industry skills practicing [35] and shooting simulation [40] etc. Inspired by these successful studies which validate that VR training technologies can be a replacement for traditional training methods, our project explores how the virtual reality wheelchair training can be a substitute for traditional wheelchair training methods. As one important advantage of virtual reality training, trainees in the virtual world will be spared from any real-world injury in potentially dangerous scenarios such as earthquake safety training [25] and other natural disaster training [31]. This property of virtual reality allows the disabled to maneuver themselves in a wheelchair safely and injury-free throughout the training process. The virtual reality training interface also provides more flexible and comprehensive training scenarios. By allowing the user to specify the generated training scenarios, a user can target certain skills to work more efficiently. Human factors such as players’ driving habits [24] and physical movement in exergames [39, 41] can be emphasized by formulating an optimization problem to solve the parameters of a training scenario that best

matches the desired training goals. These facts drive our work to parameterize the wheelchair training scenario, to formulate it as an optimization problem, and to optimize the solution.

Wheelchair Simulation in Virtual Reality. Virtual reality provides a convenient interface for training [38] or improving skills by allowing users to repeat tasks as long as they choose to [33]. Wheelchair simulation in VR can be used for either training or entertainment. Herrlich et. al. [19] successfully converts the driving characteristics of commonly used electric wheelchairs into the virtual physics system of a game engine and validates their system using a physics simulator. At the same time, multiple system modes for intelligent wheelchair behavior simulation are implemented by taking into account the kinematics of the wheelchair in the simulation [29]. Rodriguez et al. [33] develop a wheelchair simulator that is able to help disabled children familiarize themselves with the wheelchair. Harrison et al. [18] offer the conclusion that virtual environments are potentially useful to train inexperienced powered wheelchair users. However, these simulations [19, 29, 32, 33] only focus on electronic wheelchair control using the Joystick. In reality, many (if not most) new wheelchair users are using mechanical and not electrical wheelchairs. One alternative wheelchair simulation game [36], instead of using a joystick, has users maneuver themselves in a virtual wheelchair passing through multiple obstacles by pushing two SteamVR controllers forwards and backward. This game can be fun and replicates in some ways the movement of pushing a traditional mechanical wheelchair, but it lacks realistic mechanics and realistic physics. It serves essentially as an entertainment tool, whereas our goal is to develop a system that can be used to effectively train a wheelchair user. In our work, we present a novel approach to simulate a mechanical wheelchair in a virtual environment, automatically generating realistic scenarios that allow the user to improve driving skills while avoiding the risks inherent in real-world training. In addition to providing variable scenarios that mimic real-world situations, we are binding the VR trackers on the wheels of a wheelchair held in a stationary position so that users can maneuver themselves in the VR scene by pushing actual wheels as they would while actually driving a wheelchair.

3 OVERVIEW

In this project, we propose an optimization-based approach for automatically generating scenarios for wheelchair training in virtual reality. As shown in Figure 2, we develop a program that will au-

tomatically optimize a furniture arrangement to create a realistic indoor room and a path with predetermined characteristics such as length, rotations, and narrowness. Manually tuning the indoor room layouts and adjusting the paths to achieve the training goals could be nontrivial, inaccurate and time-consuming for the designers. Therefore, we optimize the training scenarios automatically using the Metropolis-Hastings Algorithm.

In our approach, we extract spatial relations between various furniture objects, set the path parameters, and combine them in a cost function. With these combined costs the optimizer can automatically generate a realistic furniture layout along with a traversable training path. As the realistic indoor room layout connects the VR training tasks more closely towards real-world navigation tasks, the training scenarios generated through our approach demonstrates high effectiveness. The different layouts are optimized using constraints that would normally be placed on furniture in a living room, office, a bedroom such as having a desk facing a chair or a bed along a wall. Furthermore, we can apply our approach to generate virtual training scenarios for rooms of different sizes, whose layouts will be optimized by our approach. In other words, both large-scale training scenarios (e.g., a conference hall) and small-scale training scenarios (e.g., a small apartment) can be synthesized automatically. Note that regardless of the size of the virtual room synthesized, the user can practice in a small room in the real world as the training is done through a VR setup and a stationary wheelchair.

The properties of the path, namely, its desired length, the extent of rotation, and narrowness, are optimized to deliver the desired training effects. A path with multiple rotations improves the skills of a user who struggles with maneuvering around objects while a narrow path helps a user who struggles with running into objects. These paths can be arranged in degrees of difficulty with a combination of all three of these characteristics. Figure 2 depicts an example of furniture and path. We initialize the objects in random positions and orientations in the room. Each initialized object has a specified pair that is relied on in the optimization process. An example of a pair might be a desk and chair. Since the chair should remain a certain distance from the desk as well as face it, the optimizer takes into account the position and rotation of the chair with respect to the desk to decide whether or not it is set up correctly. During the optimization process, the orientations and positions of the training path and objects in the room will be optimized to form a desired virtual training scenario.

4 TRAINING SCENARIO GENERATION

In order to synthesize training scenarios focusing on different skills of the users for maneuvering the wheelchair efficiently, our approach optimizes both a scene arrangement [12] and path, which help users achieve their training goals. The optimization process focuses on two parts, the first being the placement of the object and the second being the path generation. In this section, the details of the problem formulations are described.

4.1 Cost Functions

Overall Cost. A training scenario consists of a scene and a path. Let scene graph [11] $S = (V, \Theta, E)$ denote a scene consisting of M objects whose positions are denoted as $V = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_M\}$ and the rotations as $\Theta = \{\theta_1, \theta_2, \dots, \theta_M\}$. Where, $E = \{(i, j) | 1 \leq i \leq M, 1 \leq j \leq M, i \neq j\}$ are pairs of indices (i, j) with object i containing a pairwise relationship [34] with object j . Let path $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N\}$ denote the N nodes of a generated path for training. The overall cost of a training scenario is $C_{\text{total}}(P, S)$:

$$C_{\text{total}}(P, S) = C_S \mathbf{w}_S^T + C_P \mathbf{w}_P^T \quad (1)$$

where $C_S = [C_S^d, C_S^r]$ is a vector of scene costs and $\mathbf{w}_S = [w_S^d, w_S^r]$ is a vector of weights. The C_S^d is the pairwise distance cost of the

objects in the scene defined in Equation 2. The C_S^r is the pairwise rotation cost of the objects in the scene defined in Equation 3. $C_P = [C_P^d, C_P^r, C_P^n]$ is a vector of path costs and $\mathbf{w}_P = [w_P^d, w_P^r, w_P^n]$ is a vector of weights. The C_P^d is path distance cost defined in Equation 4. The C_P^r is the path rotation cost defined in Equation 5. The C_P^n is the path narrowness cost defined in the Equation 6.

Pairwise Distance Cost. Each object has a specified distance cost used to determine a distance from its pair. This cost allows for a reasonable distance between pairs. For example, it would be unreasonable to place a couch up against a TV nor would it be appealing to have a couch too far away from a TV. The pairwise distance cost avoids such scenarios by allowing a target pairwise distance to be specified by the user. Given the list of object positions as vector V and the pairs indices vector E , the pairwise distance cost $C_S^d(V, E)$ is denoted as:

$$C_S^d(V, E) = \sum_{\forall(i, j) \in E} (\|\mathbf{v}_i - \mathbf{v}_j\| - d_{i, j})^2 \quad (2)$$

where given any pair (i, j) , $\mathbf{v}_i$ is the position of object i and $\mathbf{v}_j$ is the position of object j and $d_{i, j}$ is the prespecified target distance between these paired objects.

Pairwise Rotation Cost. An object can specify a certain rotation with respect to its paired object. The orientation of certain objects is important for creating a realistic layout. Without this cost term, objects may face in random directions rather than towards their pairs. The pairwise rotation cost is calculated as the difference between the relative angles of the paired objects and their expected relative angles. Given the list of angle rotations of objects denoted as vector Θ and the pairs indices vector E , the pairwise rotation cost is denoted as:

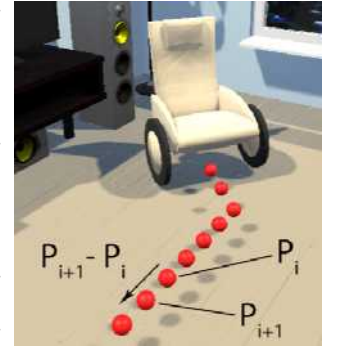
$$C_S^r(\Theta, E) = \sum_{\forall(i, j) \in E} (\|\theta_i - \theta_j\| - \delta_{i, j})^2 \quad (3)$$

where given any pair (i, j) , θ_i is the rotation of object i and θ_j is the rotation of object j and $\delta_{i, j}$ is the target relative angle between object i and j .

Path Distance Cost. A path begins at the wheelchair and ends at the target. The target position is randomly sampled during the optimization process. The path is optimized as the furniture attempt to place themselves around in the room. A distance, rotation, and narrowness for the path can be specified and used to raise or lower the difficulty of the path generated. A wide and shorter path with fewer rotations may constitute an easier path to navigate than one that is narrow, long and winding. Given a path with N nodes as $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N\}$, the path distance cost which measures the difference between path distances P and the user-specified target distance d_{path} is denoted as $C_P^d(P)$:

$$C_P^d(P) = \left(\sum_{i=1}^{N-1} \|\mathbf{p}_{i+1} - \mathbf{p}_i\| - d_{\text{path}} \right)^2 \quad (4)$$

As shown in the right figure, the total distance of a path can be calculated by summing up the distances between every two adjacent nodes in the path $\mathbf{p}_i$ and $\mathbf{p}_{i+1}$.



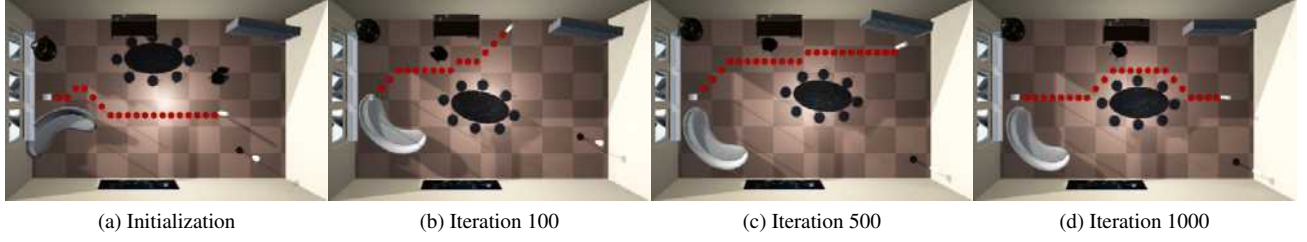


Figure 3: This figure shows an example of the optimization process. Figure (a) shows the initialization of the placement of the objects in the scene and a randomly generated path. Figure (b) (c) (d) shows a new frame in the optimization process based on the number of iterations completed at the time. Both the placement of objects and the path are randomly sampled in the scene to find the best scenario that matches the user’s specified training goal. Figure (d) shows the result of the final generated training scenario.

Path Rotation Cost. The customization of the path rotation is vital for adjusting the simulation difficulty for a user. It is important to have adjustable path difficulty for the user. For example, if a user cannot navigate turns well, adjustments can be made to have the path gradually increase rotations until the user feels comfortable with turning. The path rotation cost measures the difference between the rotation number of the path P and the user-specified target rotations r_{path} , denoted as $C_p^r(P)$:

$$C_p^r(P) = \left(\sum_{i=2}^{N-1} \Gamma_i(P) - r_{\text{path}} \right)^2 \quad (5)$$

where the rotation boolean function $\Gamma_i(P)$ returns 1 when $\mathbf{p}_{i+1} - \mathbf{p}_i \neq \mathbf{p}_i - \mathbf{p}_{i-1}$, otherwise the function returns 0. As shown in the right figure, the rotation number increases only when the adjacent two nodes have different directions.

Path Narrowness Cost. A path narrowness cost was introduced in order to manage the training difficulty with respect to how narrow a path is. Narrower paths require more precise control which mimics training scenarios within a tight virtual space. The path narrowness cost measures the difference between the average narrowness of path P and the user-specified path narrowness n_{path} and is denoted as $C_p^n(P, S)$:

$$C_p^n(P, S) = \frac{1}{N} \sum_{i=1}^N (||\mathbf{p}_i - \mathbf{q}_i|| + ||\mathbf{p}_i - \mathbf{w}_i|| - n_{\text{path}})^2 \quad (6)$$

where $\mathbf{q}_i$ is the position of the object in the scene S which is closet to the path node $\mathbf{p}_i$ on the left side, $\mathbf{w}_i$ is the closet object $\mathbf{p}_i$ on the right side.

4.2 Optimization

The goal is to synthesize a realistic setup of a room and an expected training path that a user can navigate with a wheelchair. This is done by minimizing the total cost of the training scenario C_{total} in Equation 1. Positions of the objects in the scene are sampled randomly within the room to minimize the scene costs C_S based on the parameters specified for each pair of objects. At the same time,

the path is randomly sampled between the objects in the scene to minimize the path cost C_p .

Path Computation. As illustrated in Figure 3, the optimization of the path is specified using three main costs, the distance of the path shown in Equation 4, the rotations of the path shown in Equation 5, and the narrowness shown in Equation 6. During the optimization, starting from the wheelchair, A* algorithm, a widely used pathfinding and graph traversing algorithm [6], is employed to generate a random path in each iteration. Randomly sampled paths are accepted when they are closer to the target path than the previous path. Until the best path is found given the currently synthesized layout of the scene.

Optimization Steps. Each optimization step has three types of moves:

- An object moves to a random position.
- An object rotates to a random orientation.
- The path target moves to a random position.

In our optimization approach, the Metropolis-Hastings Algorithm [27] is used to find the optimal solutions. During each iteration, whether or not to accept a new move is decided by an acceptance probability. This acceptance probability is determined by the total cost of the current status (P, S) and the proposed status (P', S') . Bounding the range of the total cost function $C_{\text{total}}(P, S)$ to the interval $[0, 1]$, we define a Boltzmann-like function [1] $f(t)$:

$$f(t) = \exp \left(-\frac{1}{t} C_{\text{total}}(P, S) \right)$$

where t is the temperature parameter of the Metropolis-Hastings Algorithm [22]. As t decreases, the optimizer is less likely to accept a worse solution. The acceptance probability function is denoted as $Pr(P', S' | P, S)$:

$$Pr(P', S' | P, S) = \min \left(1, \frac{f(P', S')}{f(P, S)} \right)$$

The acceptance probability function [22] is dependent on the number of iterations. At the start of the optimization, the temperature is set to 1.0, giving the optimizer a higher probability of accepting incorrect moves. As the iterations increase and the temperature decreases, the algorithm becomes more greedy [9] and less likely to accept worse moves than before. When the optimizer reaches its final stages all bad moves are discouraged and minimized to a probability close to zero.

Changing Parameters. As shown in Figure 4, different path parameters, and weights are applied for generating different training scenarios. Higher priority is placed on the terms with higher weights. Consequently, adjusting the settings of the weights generates paths

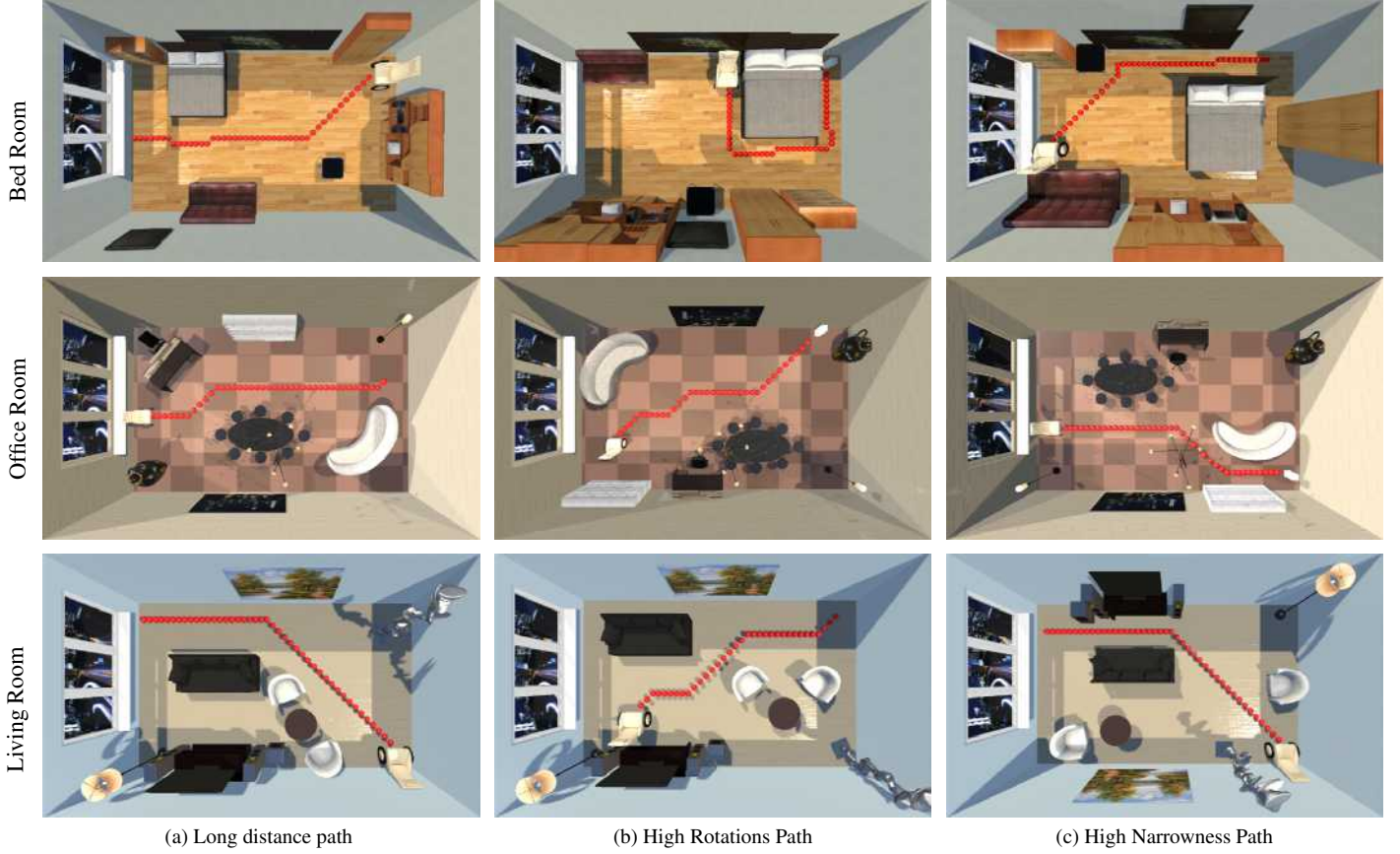


Figure 4: We have 9 different training scenarios including Column (a): long path training scenarios. Column (b): high rotations path training scenarios and Column (c): high narrowness training scenarios with different scene settings: First row: the bedrooms, second row: the office rooms and the third row: the living rooms.

emphasizing different training skills. These terms are defined in Equation 1:

- **Path Distance:** Column (a) has a longer distance target $d_{\text{path}} = 100$ with a stronger distance weight $w_p^d = 0.6$, while Column (b) and (c) have shorter distance targets $d_{\text{path}} = 50$ with weaker distance weights $w_p^d = 0.2$.
- **Path Rotation:** Column (b) has a higher rotation target $r_{\text{path}} = 4$ with a stronger rotation weight $w_p^r = 0.6$. While Column (a) and (c) have a lower rotation target $r_{\text{path}} = 1$ with a weaker rotation weight $w_p^r = 0.2$.
- **Path Narrowness:** Column (c) has a smaller narrowness target $n_{\text{path}} = 10$ with a stronger narrowness weight $w_p^n = 0.6$. While Column (a) and (b) have a wider narrowness target $n_{\text{path}} = 20$ with a weaker narrowness weight $w_p^n = 0.2$.

The weights specified above allowed for the prioritization of certain parameters over others. The higher the weight the more emphasis is placed on that characteristic. For example, in Column(a) of Figure 4, priority is placed on the path distance and lower weights are put on the path rotations and narrowness. This makes the path focus more on reaching the desired length rather than the rotation or narrowness. These settings are used for generating the nine different training scenarios for the user study in Section 6.2.

5 WHEELCHAIR SIMULATION

We simulate the motion of the wheelchair in a randomly generated training scenario by attaching VR trackers onto a real wheelchair. As shown in Figure 5, we apply the physics-based simulation [20] to simulate the different motions of the manual wheelchair [2] such as moving forward, backward and rotating.

With the HTC VIVE trackers, we can track the precise position of a point on the wheel. Let symbol $*$ denotes either left l or right r , then by defining p_* as a tracker's position and c_* as a wheel's center position, we can calculate the angular speed of both sides of the wheels $\omega_*(t)$ as a function with respect to time t :

$$\omega_*(t) = \cos^{-1} \frac{\mathbf{r}_*(t) \cdot \mathbf{r}_*(t - \Delta t)}{|\mathbf{r}_*(t)| \cdot |\mathbf{r}_*(t - \Delta t)|} \delta_*(t) \quad (7)$$

where $\mathbf{r}_*(t)$ is the radius vector of a wheel, which is defined as:

$$\mathbf{r}_*(t) = \frac{\mathbf{p}_*(t) - \mathbf{c}_*}{|\mathbf{p}_*(t) - \mathbf{c}_*|}$$

and the $\delta_*(t)$ is the rotation direction of a wheel, defined as:

$$\delta_*(t) = \text{sgn}(\mathbf{r}_*(t) \times \mathbf{r}_*(t - \Delta t) \cdot \mathbf{x})$$

In our simulation approach, the wheelchair is always facing along the z axis. Wheelchair movement in the simulation is done by translated and rotating the scene rather than the virtual wheelchair. For example, when the wheelchair seems to move forward in the simulation, it actually remains static while the scene moves past it.

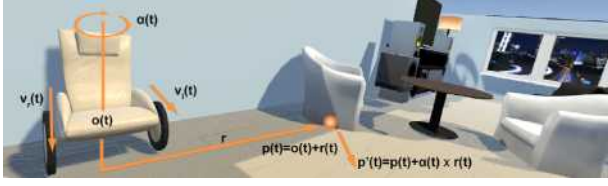


Figure 5: Physics-based wheelchair simulation: This figure shows the rotation of a scene relative to a stationary VR wheelchair according to the physics simulation. Two trackers are rotating as the user is pushing or pulling the wheels of the physical wheelchair. Let $\mathbf{o}(t)$ be the center of the wheelchair and $\mathbf{v}_l(t)$ and $\mathbf{v}_r(t)$ be the linear speed of the left and right wheels. Any point $\mathbf{p}(t)$ in the scene will rotate along the center $\mathbf{o}(t)$ with angular speed $\alpha(t)$ which is decided by $\mathbf{v}_l(t)$ and $\mathbf{v}_r(t)$. Given an arbitrary point $\mathbf{p}(t)$ whose distance to the center $\mathbf{o}(t)$ is $\mathbf{r}(t)$, after the rotation of the scene, the new position of that point will be $\mathbf{p}'(t) = \mathbf{p}(t) + \alpha(t) \times \mathbf{r}(t)$.

Given the wheels' angular speed in equation (7). Let R denotes the radius of the wheels, and, $\mathbf{s}(t)$, the linear speed of the scene is calculated as:

$$\mathbf{s}(t) = \text{sgn}(\omega_l(t)) \min(|\omega_l(t)|, |\omega_r(t)|) R$$

Let $v_*(t) = R \cdot (|\omega_*(t)| - \min(|\omega_l(t)|, |\omega_r(t)|))$ denotes the relative linear speed of the left wheel or right wheel, as shown in Figure 5, the rotation speed of the scene $\alpha(t)$ is calculated as:

$$\alpha(t) = \begin{cases} \sin^{-1} \left(\frac{v_l(t)}{|c_l - c_r|} \right) \text{sgn}(\omega_l) & v_r \leq v_l \\ -\sin^{-1} \left(\frac{v_r(t)}{|c_l - c_r|} \right) \text{sgn}(\omega_r) & v_r > v_l \end{cases} \quad (8)$$

However, Equation 8 works well when the two wheels are rotating in the same direction. If the rotation directions of two wheels are opposite to each other, then the wheelchair will only rotate along the axis at the center of $\mathbf{o}(t)$ which is calculated as:

$$\mathbf{o}(t) = \text{lerp} \left(\mathbf{c}_l, \mathbf{c}_r, \frac{|\omega_l(t)|}{|\omega_l(t) + \omega_r(t)|} \right)$$

in this case, the rotation speed of the scene $\alpha(t)$ is calculated as:

$$\alpha(t) = \sin^{-1} \left(\frac{|\omega_l(t)R|}{|c_l - \mathbf{o}(t)|} \right) \text{sgn}(\omega_l)$$

6 EXPERIMENTS

6.1 Implementation

We implemented our approach using C# and Unity 2019.2.0f1. The pre-evaluation and post-evaluation were conducted using a physical wheelchair. The VR training was run using a PC with 32.0 GB RAM, a 3.60Hz Intel(R) Core i7-9700K CPU processor, a Fresco Logic IDDCX Adapter graphics card, and a physical wheelchair with HTC VIVE trackers attached to the wheels.

6.2 Training and Evaluation

To verify the effectiveness of our VR wheelchair training, we let users go through a three-stage study. The three stages are:

- **Pre-evaluation:** The user used a physical wheelchair in a room where their wheelchair control skills were tested.
- **Training:** The VR wheelchair training included nine unique scenarios with different target settings described in Section 4.2.
- **Post-evaluation:** The user ran through the same course as the pre-evaluation.

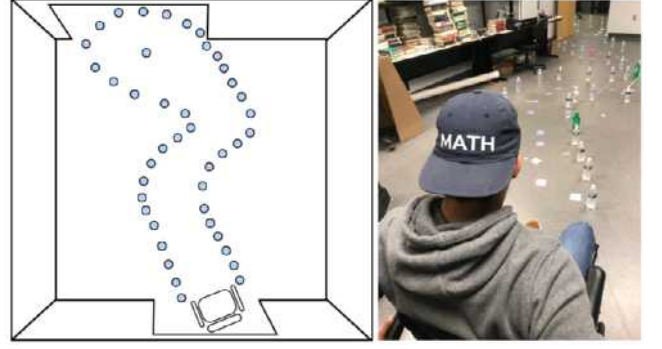


Figure 6: The real-world scenario used for evaluation. Left: Top-down view of the scenario with the path shown. Right: Photo of the scenario; water bottles were placed on the ground to measure collision.

Users. We recruited 15 users among whom two have used a wheelchair before while others have not. Each of them was put into the three-stage study in which different performances were recorded. The goal of this user study was to determine the effects of wheelchair training on the user's ability to maneuver a wheelchair.

In order to measure whether or not any improvement occurred, two metrics were used to determine the amount of improvement. The first metric was the number of obstacles a user collided with; in this case, the obstacles were bottles. This metric determined whether or not there was a significant increase in the user's wheelchair control skill from the number of bottles collided with.

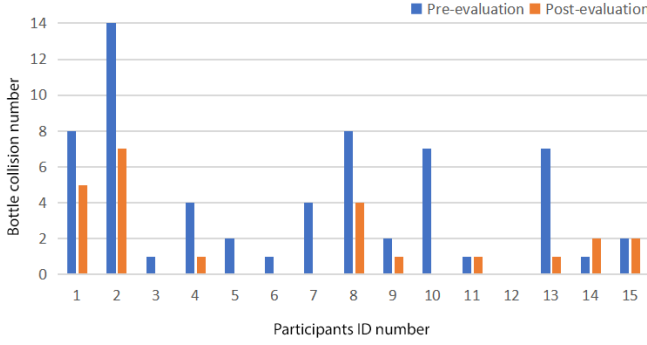
The second metric measured during the evaluations was the time it took a user to complete the task. A user completing tasks more quickly after the VR training indicated an improvement in the user's ability to use a wheelchair more efficiently.

Pre-evaluation. The pre-evaluation took place in a physical room where the users used a real wheelchair to navigate through a specified path labeled by paper markers as shown in Figure 6. Each user sitting in the wheelchair was put in a room with multiple obstacles, in our case, water bottles. He was asked to go through the room following the pre-determined path on the floor marked by the papers and then to return back to the start point. This phase took the user an average of one to two minutes to complete. His performance was tracked with the number of water bottles he collided with and the amount of time it took him to complete the entire course.

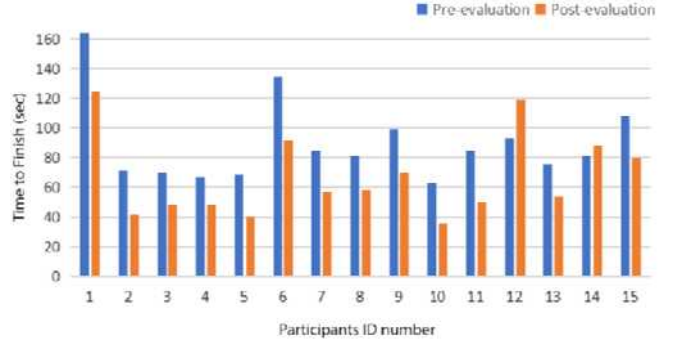
VR Training. After having completed the pre-evaluation in the physical room, the user performed the virtual reality training. During the VR training, the user was put into nine training scenarios, all of which were pre-optimized with the technical approach described in Section 4.2. The scenarios encompassed three different types of rooms with three path settings for each type of room as shown in Figure 4. The three different rooms included a bedroom, an office and a living room. In each type of room, the user went through training scenarios with three different specified path targets, namely, long-distance path, highly rotating path, and narrow path.

- **Long-distance path:** The path was straight and long for the user to practice controlling the wheelchair to go straight.
- **Highly-rotating path:** The path had more rotations so the user practiced navigating the wheelchair in environments involving a large number of turns.
- **Narrow path:** A narrow path was used to enhance the user's skill of controlling a wheelchair precisely.

As shown in the Figure 4, different types of paths varied in the number of rotations, path length, and path narrowness allowing for



(a) Bottle collisions number



(b) Time to finish

Figure 7: This figure shows the bar plots of pre-evaluation(blue) and post-evaluation(orange). There were 15 users in total. Figure (a) is the number of bottle collisions. Figure (b) is the time to finish the task for each user.

different aspects of VR training. The objective of the users during the VR training was to follow the automatically generated path as closely as possible.

During the VR training, the user was asked to follow the path of red spheres in the VR room. During the training, the user sat in a real wheelchair, which was raised to keep the wheels off the floor. To move the wheelchair in virtual space the physical wheelchair wheels were pushed. The trackers attached to the wheel picked up the movement and moved the virtual wheelchair through the simulation approach described in Section 5. For each training scenario, it took about 45-90 seconds to complete the task. Overall, it took about 10-15 minutes to complete the whole VR training with nine different scenarios.

Post-evaluation. In order to validate that after the nine different VR training scenarios, the user's proficiency and the precision of the wheelchair control improved significantly, the same real-world task was assigned to the user again in order to compare his performance before and after. In the post-evaluation, the user reran the same course and completed the same tasks that were specified in the pre-evaluation. The measurement of the time it took to complete the tasks as well as the number of bottles that the user collided with the second run were recorded and analyzed to determine whether or not the user benefited from the training.

After having the user completed all the assigned training tasks, the user was prompted to complete a questionnaire that offered different questions for feedback on the effectiveness of the VR training in their ability to use a wheelchair, and whether or not the simulation was realistic compared with a real wheelchair.

7 RESULTS AND DISCUSSION

We discussed the results of the different training sessions and analyzed users' improvement in terms of how well the wheelchair was controlled before the VR training and after. In order to prove that the user's proficiency and the precision of the wheelchair control improve after completed the VR training scenarios automatically generated through our technical approach described in Section 4.2, we compared the users' performance during the pre-evaluation and the post-evaluation. The comparison includes bottle collisions during the evaluation and total time to complete the evaluation.

As shown in the Figure 7. there are two columns for each of the 15 users, illustrating the number of the bottle collisions (a) and amount of time to finish the task in seconds (b). The blue columns are from the pre-evaluation records and the orange columns are from the post-evaluation records.

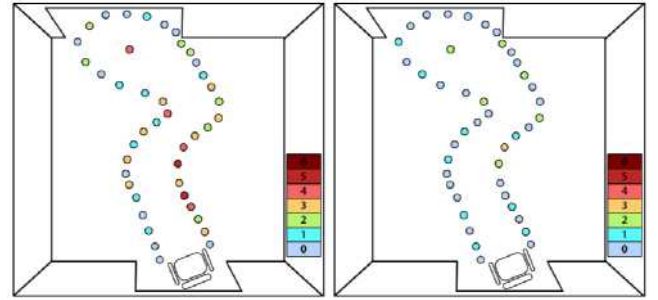


Figure 8: Visualizations of the number of the bottles knocked down during pre-evaluation (Left) and post-evaluation (Right). Colored circles are the places where the bottles have been knocked down. Colors show the frequency of that position where collisions happen. As the color goes from blue to red, the number of collisions increases.

In order to measure whether the improvement of the user's skills in controlling a wheelchair is significant enough. We apply the ANOVA test to compare the pre-evaluation data and post-evaluation data. In our case, the factors are time duration and bottle number. Therefore, we have applied two separate one-way ANOVA tests [4, 28] respectively. The one-way analysis of variance (ANOVA) is used to determine whether there are any statistically significant differences between the means of two or more independent groups. ANOVA test is the most widely used statistical test for hypothesis testing in factorial experiments. To prove that both the proficiency and the precision of the wheelchair control increase after the VR training, we apply two ANOVA tests separately for both the time duration factor and bottle collision factor.

The time duration factor measures how efficiently the user can control the wheelchair. The H_0 hypothesis (null hypothesis) assumes that there is no significant difference between the mean values of the time to finish the pre-evaluation and the mean values of the time to finish the post-evaluation. However, we can prove that there is a statistically significant difference between the two. As calculated, $P_{value} = 0.003627 < 0.05$, which means there is a statistically significant difference in the time duration to complete the task before VR training and after the VR training. The mean finishing time for the pre-evaluation is 89.9 seconds and the mean finishing time for the post-evaluation is 67.133 seconds. Therefore, we reject H_0 .

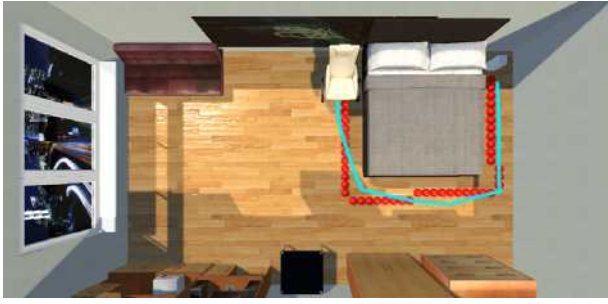


Figure 9: User's path compared with the generated path: This figure shows that during the VR training, users are trying to follow the generated path, where the blue curve is the user's path automatically recorded by the computer, the red dotted curve is the generated path. The significant consistency between the user's recorded trajectory and the prespecified path verifies that the user's improvement of performance is due to the generated training scenarios.

Bottle collision factor measures how precisely the user can control the wheelchair along a given path. As shown in Figure 8, the left figure shows the number of bottle collisions during the pre-evaluation and the right figure shows the number of bottle collisions during the post-evaluation. We apply another ANOVA test to the pre-evaluation data and the post-evaluation data. The H_0 hypothesis (null hypothesis) suggests that there is no significant difference between the mean values of the number of bottle collisions during the pre-evaluation and the mean values of the number of bottle collisions during the post-evaluation. However, we can prove that there is a statistically significant difference between the two. As calculated, $P_{value} = 0.003588 < 0.05$, which means there is a statistically significant difference in how precisely the users control the wheelchair path before VR training and after. The mean number of bottle collisions for the pre-evaluation is 4.13 and the mean number of bottle collisions for the post-evaluation is 1.6. Therefore, we reject H_0 .

As shown in the two ANOVA test results, we can conclude that both the proficiency and the precision of the wheelchair control reject H_0 and strongly validate our VR training effects. In other words, the user has significant improvement in wheelchair control reflected in two aspects: speed and accuracy. The program will automatically record the path taken by the users in the virtual room. As the user follows the specified path in the VR training scenario, as shown in Figure 9, their improvement validates our approach that through these automatically generated training scenarios, the training effects are strong enough to be a substitute for other effective training methods.

Besides the performance analysis, we designed a questionnaire to ask the user how they felt during the study. The feedback is shown in Figure 10. Most of the users thought that the wheelchair simulation was realistic; the control of the VR wheelchair was natural; and the VR training was comfortable, enjoyable, and effective. This feedback is consistent with the statistical analysis results of the users' performance during the user study.

8 SUMMARY

We introduced an optimization-based approach for training wheelchair skills, with emphasis on different aspects of the wheelchair skills such as moving long straight distances, rotating around objects, and passing through a narrow path. We developed the optimization approaches to generate scenarios to achieve user-specified training goals. In order to make the VR training more immersive, we designed a physics-based wheelchair simulation algorithm to map the motion of the real wheelchair onto the VR wheelchair whose control is proved to be natural. In the end, we

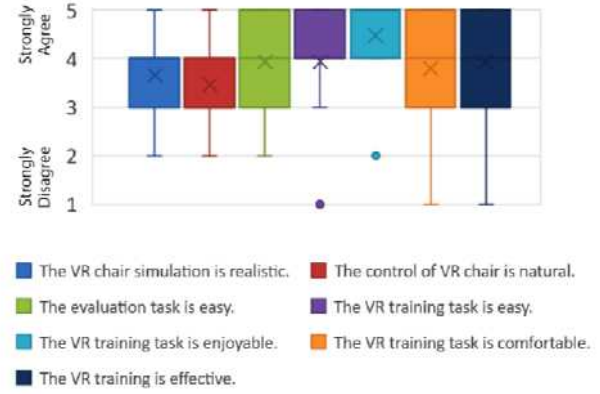


Figure 10: This figure shows the feedback of 15 users after the nine different VR training scenarios. Different colors represent different questions in the questionnaire. Different numbers indicate the extent to which the users agree. 1 indicates strongly disagrees while 5 indicates strongly agrees. The results draw a positive conclusion that most users agree that their performance in maneuver a wheelchair improved after the VR training.

have designed a user study to validate our approaches. From the analysis results, we conclude that our approach improved the user's ability to finish the wheelchair tasks efficiently.

Limitations. Due to the VR hardware constraints, long-time training tasks will result in the users' visual fatigue and possible loss of concentration. However, this is hard to solve at this moment because there are gaps between virtual reality and reality that are physically existing and not easy to avoid.

Future Work. To eliminate the impacts which are not from VR training, in future user studies, we could include a control group that did not perform the VR training. The control group should only have the pre-evaluation and post-evaluation, without the training part. Then we can validate the effectiveness of VR training by comparing the VR group and the control group.

In order to increase the user's interest during the whole training process, we may introduce some penalty-reward mechanisms to increase the level of entertainment. This will introduce some positive effects with respect to the training efficiency, however, there are risks of some unexpected negative effects as well, such as the user paying attention to the game part more than the training process. This can be an element of work to explore and study in future work.

REFERENCES

- [1] E. Aarts and J. Korst. Simulated annealing and boltzmann machines. 1988.
- [2] K. T. Asato, R. A. Cooper, R. N. Robertson, and J. Ster. Smart/sup wheels: development and testing of a system for measuring manual wheelchair propulsion dynamics. *IEEE Transactions on Biomedical Engineering*, 40(12):1320–1324, 1993.
- [3] K. L. Best, R. L. Kirby, C. Smith, and D. A. MacLeod. Wheelchair skills training for community-based manual wheelchair users: a randomized controlled trial. *Archives of Physical Medicine and Rehabilitation*, 86(12):2316–2323, 2005.
- [4] G. E. Box et al. Some theorems on quadratic forms applied in the study of analysis of variance problems, i. effect of inequality of variance in the one-way classification. *The annals of mathematical statistics*, 25(2):290–302, 1954.
- [5] J. Cremer, J. Kearney, and Y. Papelis. Driving simulation: challenges for vr technology. *IEEE Computer Graphics and Applications*, 16(5):16–20, 1996.

- [6] X. Cui and H. Shi. A*-based pathfinding in modern computer games. *International Journal of Computer Science and Network Security*, 11(1):125–130, 2011.
- [7] S. De Groot, M. De Bruin, S. Noomen, and L. Van der Woude. Mechanical efficiency and propulsion technique after 7 weeks of low-intensity wheelchair training. *Clinical biomechanics*, 23(4):434–441, 2008.
- [8] S. De Groot, H. Veeger, A. Hollander, and L. Van der Woude. Adaptations in physiology and propulsion techniques during the initial phase of learning manual wheelchair propulsion. *American journal of physical medicine & rehabilitation*, 82(7):504–510, 2003.
- [9] R. A. DeVore and V. N. Temlyakov. Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187, 1996.
- [10] J. F. Dols, J. Molina, F. J. Camacho, J. Marín-Morales, A. M. Pérez-Zuriaga, and A. Garcia. Design and development of driving simulator scenarios for road validation studies. *Transportation research procedia*, 18:289–296, 2016.
- [11] C. Erikson and D. Manocha. Simplification culling of static and dynamic scene graphs. *UNC-Chapel Hill Computer Science TR98-009*, 1998.
- [12] M. Fisher, D. Ritchie, M. Savva, T. Funkhouser, and P. Hanrahan. Example-based synthesis of 3d object arrangements. *ACM Transactions on Graphics (TOG)*, 31(6):135, 2012.
- [13] A. Gallagher, N. McClure, J. McGuigan, I. Crothers, and J. Browning. Virtual reality training in laparoscopic surgery: a preliminary assessment of minimally invasive surgical trainer virtual reality (mist vr). *Endoscopy*, 31(04):310–313, 1999.
- [14] N. Gavish, T. Gutiérrez, S. Webel, J. Rodríguez, M. Peveri, U. Bockholt, and F. Tecchia. Evaluating virtual reality and augmented reality training for industrial maintenance and assembly tasks. *Interactive Learning Environments*, 23(6):778–798, 2015.
- [15] T. P. Grantcharov, V. B. Kristiansen, J. Bendix, L. Bardram, J. Rosenberg, and P. Funch-Jensen. Randomized clinical trial of virtual reality simulation for laparoscopic skills training. *British journal of surgery*, 91(2):146–150, 2004.
- [16] K. Gurusamy, R. Aggarwal, L. Palanivelu, and B. Davidson. Systematic review of randomized controlled trials on the effectiveness of virtual reality training for laparoscopic surgery. *British Journal of Surgery*, 95(9).
- [17] K. S. Gurusamy, R. Aggarwal, L. Palanivelu, and B. R. Davidson. Virtual reality training for surgical trainees in laparoscopic surgery. *Cochrane database of systematic reviews*, (1), 2009.
- [18] A. Harrison, G. Derwent, A. Enticknap, F. Rose, and E. Attree. The role of virtual reality technology in the assessment and training of inexperienced powered wheelchair users. *Disability and rehabilitation*, 24(11-12):599–606, 2002.
- [19] M. Herrlich, R. Meyer, R. Malaka, and H. Heck. Development of a virtual electric wheelchair-simulation and assessment of physical fidelity using the unreal engine 3. In *International Conference on Entertainment Computing*, pp. 286–293. Springer, 2010.
- [20] F. Jiarang and Y. Jianqiao. An exact solution for the statics and dynamics of laminated thick plates with orthotropic layers. *International Journal of Solids and Structures*, 26(5-6):655–662, 1990.
- [21] R. L. Kirby, D. Mitchell, S. Sabharwal, M. McCranie, and A. L. Nelson. Manual wheelchair skills training for community-dwelling veterans with spinal cord injury: a randomized controlled trial. *PloS one*, 11(12):e0168330, 2016.
- [22] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.
- [23] U. Kühnapfel, H. K. Cakmak, and H. Maaß. Endoscopic surgery training using virtual reality and deformable tissue simulation. *Computers & graphics*, 24(5):671–682, 2000.
- [24] Y. Lang, L. Wei, F. Xu, Y. Zhao, and L.-F. Yu. Synthesizing personalized training programs for improving driving habits via virtual reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 297–304. IEEE, 2018.
- [25] C. Li, W. Liang, C. Quigley, Y. Zhao, and L.-F. Yu. Earthquake safety training through virtual drills. *IEEE transactions on visualization and computer graphics*, 23(4):1275–1284, 2017.
- [26] A. H. MacPhee, R. L. Kirby, A. L. Coolen, C. Smith, D. A. MacLeod, and D. J. Dupuis. Wheelchair skills training program: A randomized clinical trial of wheelchair users undergoing initial rehabilitation. *Archives of Physical Medicine and Rehabilitation*, 85(1):41–50, 2004.
- [27] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.
- [28] P. Moran and C. Smith. The correlation between relatives on the supposition of mendelian inheritance. *Transactions of the Royal Society of Edinburgh*, 52:899–438, 1918.
- [29] H. Niniss and A. Nadif. Simulation of the behaviour of a powered wheelchair using virtual reality. In *3rd International Conference on Disability, Virtual Reality and Associated Technologies*, pp. 9–14, 2000.
- [30] Pants Up Easy. U.S. Wheelchair User Statistics. <https://www.pantsupeasy.com/u-s-wheelchair-user-statistics/>, April 2016. Accessed November 24, 2019.
- [31] R. Querrec, C. Buche, E. Maffre, and P. Chevallier. Sécurévi: virtual environments for fire-fighting training. In *5th virtual reality international conference (VRIC’03)*, pp. 169–175, 2003.
- [32] P. D. Ritsos and W. Nigel. A cost-effective virtual environment for simulating and training powered wheelchairs manoeuvres. In *Proc. Med. Meets Virtual Reality NextMed/MMV*, vol. 220, p. 134, 2016.
- [33] N. Rodriguez. Development of a wheelchair simulator for children with multiple disabilities. In *2015 3rd IEEE VR International Workshop on Virtual and Augmented Assistive Technology (VAAT)*, pp. 19–21. IEEE, 2015.
- [34] T. Shao, W. Xu, K. Zhou, J. Wang, D. Li, and B. Guo. An interactive approach to semantic modeling of indoor scenes with an rgbd camera. *ACM Transactions on Graphics (TOG)*, 31(6):136, 2012.
- [35] E. Van Wyk and R. De Villiers. Virtual reality training applications for the mining industry. In *Proceedings of the 6th international conference on computer graphics, virtual reality, visualisation and interaction in Africa*, pp. 53–63. ACM, 2009.
- [36] ViRa Games. Wheelchair Simulator. https://store.steampowered.com/app/871510/Wheelchair_Simulator/, June 2018.
- [37] J. Westwood, H. Hoffman, D. Stredney, and S. Weghorst. Validation of virtual reality to teach and assess psychomotor skills in laparoscopic surgery: results from randomised controlled studies using the mist vr laparoscopic simulator. *Medicine Meets Virtual Reality: art, science, technology: healthcare and evolution*, p. 124, 1998.
- [38] R. Williams and J. Dattilo. Using wheelchair simulations to teach about inclusion. *Schole: A Journal of Leisure Studies and Recreation Education*, 20(1):140–145, 2005.
- [39] B. Xie, Y. Zhang, H. Huang, E. Ogawa, T. You, and L.-F. Yu. Exercise intensity-driven level design. *IEEE transactions on visualization and computer graphics*, 24(4):1661–1670, 2018.
- [40] W. G. Zaenglein Jr. Shooting simulating process and training device using a virtual reality display screen, June 24 1997. US Patent 5,641,288.
- [41] Y. Zhang, B. Xie, H. Huang, E. Ogawa, T. You, and L.-F. Yu. Pose-guided level design. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 554. ACM, 2019.