

PERSONALIZING VR TRAINING EXPERIENCES THROUGH OPTIMIZATION

by

Wanwan Li

A Dissertation

Submitted to the

Graduate Faculty

of

George Mason University

In Partial fulfillment of

The Requirements for the Degree

of

Doctor of Philosophy

Computer Science

Committee:

| | |
|-------------|--|
| _____ | Dr. Lap-Fai Yu, Dissertation Director |
| _____ | Dr. Jyh-Ming Lien, Committee Member |
| _____ | Dr. Bo Han, Committee Member |
| _____ | Dr. Behzad Esmaili, Committee Member |
| _____ | Dr. David S. Rosenblum, Department Chair |
| _____ | Dr. Kenneth S. Ball, Dean, Volgenau School of Engineering |
| Date: _____ | Summer Semester 2021 George Mason University Fairfax, VA |

Personalizing VR Training Experiences through Optimization

A dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy at George Mason University

By

Wanwan Li
Master of Science
University of Central Florida, 2016
Bachelor of Science
Harbin Institute of Technology, 2014

Director: Dr. Lap-Fai Yu, Professor
Department of Computer Science

Summer Semester 2021
George Mason University
Fairfax, VA

Copyright © 2021 by Wanwan Li
All Rights Reserved

Dedication

I dedicate this dissertation to my academic advisor Dr. Lap-Fai(Craig) Yu, who has given me the greatest help and enlightened me the most during my journey of Ph.D. study.

Acknowledgments

I would like to thank with my warmest regard the following teammates who have collaborated to complete my research work, they are: Dr. Haikun (Quincy) Huang, Biao Xie (Ph.D. student), Yongqi Zhang (Ph.D. student), Javier Talavera (Senior), and Amilcar Gomez Samayoa (Senior). I would also thank my parents for their kind suggestions and encourages for my study with my genuine gratefulness.

Table of Contents

| | Page |
|--|------|
| List of Tables | viii |
| List of Figures | ix |
| Abstract | 0 |
| 1 Introduction | 1 |
| 1.1 Existing Work of Synthesizing Virtual Training Scenarios | 2 |
| 1.2 Existing Work of Synthesizing Exergame Level Designs | 4 |
| 1.3 Problem Statements, Thesis Synopsis and Organizations | 6 |
| 2 Automatic Synthesis of Virtual Wheelchair Training Scenarios | 8 |
| 2.1 Introduction | 8 |
| 2.2 Related Work | 10 |
| 2.3 Overview | 12 |
| 2.4 Training Scenario Generation | 13 |
| 2.4.1 Cost Functions | 14 |
| 2.4.2 Optimization | 17 |
| 2.5 Wheelchair Simulation | 20 |
| 2.6 Experiments | 22 |
| 2.6.1 Implementation | 22 |
| 2.6.2 Training and Evaluation | 23 |
| 2.7 Results and Discussion | 27 |
| 2.8 Summary | 29 |
| 3 Exertion-Aware Path Generation | 32 |
| 3.1 Introduction | 32 |
| 3.2 Related Work | 34 |
| 3.2.1 Procedural Modeling | 34 |
| 3.2.2 Exergaming | 35 |
| 3.2.3 Immersive Virtual Environments | 36 |
| 3.2.4 Pathfinding | 37 |
| 3.3 Overview | 38 |
| 3.4 Technical Approach | 38 |

| | | |
|-------|---|----|
| 3.4.1 | Problem Formulation | 38 |
| 3.4.2 | Cost Terms | 40 |
| 3.4.3 | Path Optimization | 46 |
| 3.4.4 | Road Synthesis | 51 |
| 3.5 | Experiments | 54 |
| 3.5.1 | Implementation | 54 |
| 3.5.2 | Different Terrains and Exertion Goals | 58 |
| 3.5.3 | Path Length | 58 |
| 3.5.4 | Landmarks | 59 |
| 3.5.5 | Forbidden Zones | 59 |
| 3.6 | User Study | 60 |
| 3.6.1 | Enjoyment Test | 61 |
| 3.6.2 | Exertion Test | 61 |
| 3.7 | Summary | 65 |
| 3.7.1 | Limitations and Future Work | 65 |
| 4 | Synthesizing Construction Safety Training Scenarios for VR Training | 68 |
| 4.1 | Introduction | 68 |
| 4.2 | Related Work | 70 |
| 4.3 | Overview | 71 |
| 4.4 | Training Scenario Synthesis | 72 |
| 4.4.1 | Cost Functions | 73 |
| 4.4.2 | Optimization | 78 |
| 4.5 | Experiments | 81 |
| 4.5.1 | Implementation | 81 |
| 4.5.2 | Training and Evaluation | 82 |
| 4.6 | Results and Discussion | 86 |
| 4.7 | Summary | 89 |
| 5 | Conclusion and Future Work | 92 |
| 5.1 | Summary | 92 |
| 5.2 | Limitation | 93 |
| 5.2.1 | Hardware-Related Limitations | 93 |
| 5.2.2 | Assessment-Related Limitations | 94 |
| 5.3 | Future Work | 94 |
| 5.3.1 | Personalized Public Safety Training Synthesis | 94 |
| 5.3.2 | Personalized Law Enforcement Training Synthesis | 95 |
| 5.3.3 | Personalized Exergame Level Synthesis | 96 |

| | | |
|-------|--|-----|
| A | Appendix | 97 |
| A.1 | Experiments | 97 |
| A.1.1 | Obstacles | 97 |
| A.2 | User Study | 97 |
| A.2.1 | Details of Enjoyment Test | 97 |
| A.2.2 | Conversion Between Expected Feedback Force and Expected Power Output Target | 99 |
| A.2.3 | Additional Details of Demographic Information | 99 |
| A.2.4 | Additional Analysis of User Study Results | 101 |
| B | Appendix | 104 |
| B.1 | Lecture-based Training | 104 |
| B.2 | Evaluation Tasks | 108 |
| B.3 | Evaluation Answers | 110 |
| B.4 | Users' Feedback Details | 111 |
| B.5 | Conditional Probability for Dynamic Hazard | 112 |
| B.6 | Synthesized Training Scenarios | 112 |
| | Bibliography | 115 |

List of Tables

| Table | Page |
|---|------|
| 3.1 Descriptive statistics for the Hard15 and Hard30 paths. The average completion duration, power output and energy expenditure of the participants are shown. | 65 |
| A.1 Demographic information of participants (Mean (SD)) in the enjoyment test. | 100 |
| A.2 Demographic information of participants (Mean (SD)) in the exertion test. | 100 |
| A.3 PACES items. Please rate how you feel at the moment about the physical activity you have been doing. | 103 |
| B.1 Answers for pre-evaluation(Control Group). | 108 |
| B.2 Answers for pre-evaluation(VR Group). | 108 |
| B.3 Answers for post-evaluation(Control Group). | 110 |
| B.4 Answers for post-evaluation(VR Group). | 110 |

List of Figures

| Figure | | Page |
|--------|---|------|
| 1.1 | This figure shows two examples of virtual training: (a) Public safety training program design in virtual reality using Virtual Drills to simulate earthquakes and (b) Synthesized personalized training programs for improving driving habits via virtual reality. | 3 |
| 1.2 | This figure shows two examples of synthesized exergames. | 5 |
| 2.1 | Overview of our approach | 9 |
| 2.2 | 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. | 15 |
| 2.3 | Distance. | 16 |
| 2.4 | Rotation. | 16 |
| 2.5 | Narrowness. | 17 |
| 2.6 | 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. | 18 |

| | | |
|------|---|----|
| 2.7 | 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 $\mathbf{p}'(t) = \mathbf{p}(t) + \alpha(t) \times \mathbf{r}(t)$ | 22 |
| 2.8 | 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. | 23 |
| 2.9 | 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. | 24 |
| 2.10 | 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. | 26 |
| 2.11 | 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. . . | 29 |
| 2.12 | 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. | 30 |
| 3.1 | Overview of our approach. | 35 |
| 3.2 | Problem representation. Black dots refer to curve \mathcal{R} 's control points. . . . | 39 |

| | | |
|------|--|----|
| 3.3 | Resistance force. A bike experiences a resistance force f which varies with the path elevation angle θ . The resistance force is composed of the gravitational force f_g and frictional force f_μ | 42 |
| 3.4 | Top: A path whose elevation angle changes smoothly from -40° to 40° . Bottom: the force feedback of the exercise bike used to approximate the resistance force experienced at the corresponding elevation angle. | 43 |
| 3.5 | Move strategies. Each move is applied to the control points of the 2D Hermite curve \mathcal{R} on the path grid. The arrows indicate the moving directions of the control points. Each move modifies curve \mathcal{R} , which in turn modifies path \mathcal{P} on the 3D terrain. | 44 |
| 3.6 | Elevations of the paths from the easy and hard groups. The paths correspond to real-world trails of the MTB Project. The elevation along each path is shown. The hard paths are generally bumpier than the easy paths. | 47 |
| 3.7 | Attaining high coplanarity of local road surfaces through our post-processing method. (a) Input terrain that is bumpy. (b) At the 50 th iteration, as the accumulated approximation error exceeds the threshold ϵ , the terrain between t_0 and $t_1 - \Delta t$ is approximated by a linear segment. t_0 is then reset to $t_1 - \Delta t$ to start searching for the next linear segment. (c) At the 80 th iteration, as the accumulated error exceeds the threshold ϵ again, the next linear segment is formed. (d) The output is an approximation of the input terrain by a series of linear segments. | 48 |
| 3.8 | A terrain is refined to create a horizontal road surface. | 52 |
| 3.9 | A tilted road. | 53 |
| 3.11 | Our custom-built bike. | 54 |
| 3.10 | Paths generated on different terrains with different exertion goals. In each row, the path difficulty target (e.g., easy) and total work target (e.g., 15kJ) are shown on the left. The path length target is 2,500m for all. The path difficulty, path length, total work, as well as the elevation and feedback force plots of each generated path are shown. The blue arrow and the red pin refer to the start and end. | 55 |
| 3.12 | A long path and a short path generated using long and short path length targets on the same input terrain. Both paths used the same total work target $\rho_W = 22.5\text{kJ}$. Our approach generates (a) a short uphill path which is more intensive; and (b) a long downhill path which is less intensive. | 56 |

| | | |
|------|--|----|
| 3.13 | Paths generated with specified landmarks (in blue). All paths were generated with the same path length target $\rho_L = 2,500\text{m}$ and total work target $\rho_L = 22.5\text{kJ}$. The generated paths pass the landmarks closely. | 57 |
| 3.14 | Paths generated with the avoidance of forbidden zones which refer to sea water in this example. (a) A path generated as the sea level is 150. (b) A path generated as the sea level rises to 250m. | 60 |
| 3.15 | Paths generated with the avoidance of manually specified obstacles which are the bunny, dragon, and teapot in this case. | 62 |
| 3.17 | Average energy expenditure in biking the Hard15 and Hard30 paths versus the targets. A significant difference was found with $p < 0.001$ | 62 |
| 3.16 | Three modes, delivered in random order, were used for the enjoyment test. (a) VR mode based on our VR bike system; (b) TV mode where the user biked while watching the game on a screen; (c) Gym mode where only biking information was displayed. | 63 |
| 3.18 | Boxplots of the participants' power output in biking the Hard15 and Hard30 paths. The red line show the average result of the 10 participants. The blue line shows the expected power output target. | 64 |
| 3.19 | Failure cases. Our approach failed to find (a) a hard path on a plain and (b) an easy path on a rocky mountain. | 67 |
| 4.1 | Overview of our approach. (a) We synthesize the tasks to pre-evaluate users' background. (b) According to the users' performance, we synthesize personalized training scenarios. (c) Users are trained through those synthesized training scenarios that emphasize their specific skills. (d) We post-evaluate the users' performance through the same sequence of tasks in pre-evaluation to measure their improvement. | 69 |
| 4.2 | Examples of common hazards. (a) The struck-by hazard caused by the swing of excavator. (b) The fall hazard caused by missing guardrail protections. (c) The caught-in/between hazard caused by the objects which are too close to the cave's edge. (d) The electrical hazard caused by electrical component such as light pole. | 73 |
| 4.3 | Dynamic Hazard. | 75 |
| 4.4 | Static Hazard. | 75 |

| | | |
|-----|--|----|
| 4.5 | Process of optimization. The green curves denote the navigation routes synthesized through optimization. (a) The route initialized at the entrance of the construction site. During the iterations, more vertices are randomly added, replaced, or removed to form new navigation routes. At the same time, different types of hazards (colored in red, blue, yellow, or green) will be randomly added, replaced, or removed simultaneously. Figure (b-d) shows a new frame in the optimization process based on the number of iterations completed at the time. Figure (d) shows the result of the final synthesized training scenario. | 76 |
| 4.6 | Sign. | 78 |
| 4.7 | Personalized training scenarios synthesized on the same construction site. Green curves are the navigation routes. The regions with fall hazards are highlighted in blue, caught-in/between hazards in yellow, electrical hazards in green, and the struck-by hazards in red. Six different scenarios are shown, including (a) a route passing through fewer hazardous regions; (b) a route passing through more regions with fall hazards; (c) a route passing through more regions with caught-in/between hazards; (d) a route passing through more regions with electrical hazards; (e) a route passing through more regions with struck-by hazards; and (f) a route passing through regions with all types of hazards. | 79 |
| 4.8 | Three virtual construction sites for user studies: (a) Construction Site 1 is used for pre-train and VR training. (b) Construction Site 2 is used for VR training. (c) Construction Site 3 is used for pre and post-evaluation. By differing the construction sites used for training and the ones used for evaluations, we avoid the biased result in the user study that users might repeat similar tasks in VR training that also appear in evaluations if both scenarios were synthesized on the same construction site. | 82 |

| | | |
|------|---|-----|
| 4.9 | Two operations on controllers: (a) the button on the left controller is used to control the navigation speed while the button on the right controller is used to click the correct answers. (b) Four answer buttons pop up after the user clicks on the "Select Hazard" button. When the front lightsaber on the right controller intersects with, for example, the "Caught-In" button, that button starts twinkling. Until the user clicks with the front trigger on the right controller, that answer will be submitted, and all the answer buttons will disappear. | 83 |
| 4.10 | Inspection errors. The number of mistakes users made on each type of hazard during the pre-evaluation and post-evaluation are plotted in this diagram. Darker colors represent the number of mistakes made in pre-evaluation while the lighter colors represent those in post-evaluation. Blue, yellow, and green colors denote the errors made on inspecting fall, caught-in/between, and electrical hazards, respectively. | 85 |
| 4.11 | Examples of common mistakes: (a) mistaken struck-by as fall hazard; (b) mistaken caught-in/between as fall hazard; (c) ignored caught-in/between hazard; (d) ignored electrical hazard. | 88 |
| 4.12 | Users' Feedback. | 91 |
| A.1 | Paths generated with the avoidance of manually specified obstacles which is an Armadillo in this case. | 98 |
| A.2 | Enjoyment test. PACES results for different modes. | 100 |
| A.3 | Illustration of the inertia effects: Blue curve is the targets power output $P(t)$ from Hard 30kJ. Yellow curve is the predicted power output $p(t)$ using Equation A.1 which is closer to the user study result. | 102 |
| B.1 | Lecture-based Training Slides(Part I). | 106 |
| B.2 | Lecture-based Training Slides(Part II). | 107 |
| B.3 | The 12 evaluation tasks. Including 5 fall hazards, 3 caught-in hazards, and 4 electrical hazards. Blue boundaries highlight the hazardous object in that region. Users from both groups are suggested to pay attention to the blue boundary where they are and only inspect the hazardous objects within the same boundary. | 109 |

| | | |
|-----|---|-----|
| B.4 | Personalized training scenarios synthesized on the construction site 1. Green curves are the navigation routes. The regions with fall hazards are highlighted in blue, caught-in hazards in yellow, electrical hazards in green, and the struck-by hazards in red. Here are four different training scenarios including (a) a route passing through fewer hazardous regions, this scenario is easy for the beginners to start with, and is used for the pre-train. (b) a route passing through more regions with fall hazards. (c) a route passing through more regions with caught-in hazards. (d) a route passing through more regions with electrical hazards. | 114 |
|-----|---|-----|

Abstract

PERSONALIZING VR TRAINING EXPERIENCES THROUGH OPTIMIZATION

Wanwan Li, PhD

George Mason University, 2021

Dissertation Director: Dr. Lap-Fai Yu

In view of the emerging technologies in the virtual reality industry, VR provides a practical platform to train people in a safe and time-saving manner through serious virtual training tasks. Recent researches validate the effectiveness of virtual training, especially, it has been proved that personalized training in virtual reality is efficient enough to emphasize users' specific skills. Inspired by the recent works that automatically synthesize different training scenarios to achieve different personalized training targets through optimization techniques that can be applied to substitute manual configuration.

My dissertation focuses on three interdisciplinary areas including personalized wheelchair training, exertion-aware path generation, and construction safety training. We first introduce an optimization-based approach for automatically generating virtual scenarios for wheelchair training in virtual reality. We then illustrate a novel approach for generating paths with desired exertion properties. After this, we will present a novel approach to synthesize construction safety training scenarios to train the users on how to proficiently inspect the potential hazards within the construction sites in virtual reality. In the end, we will draw a conclusion on existing research results and prospect the future trend of synthesizing personalized training scenarios in virtual reality.

Chapter 1: Introduction

As virtual training is gaining popularity and provide a highly immersive environment for users to practice their skills in virtual world, recent studies are focusing on investigating the possibility of VR training to substitute traditional training. For example, Li et al.[1] implements a VR training program within an immersive virtual indoor scene, simulate the virtual earthquake, and train people about how to survive in an earthquake. Through a series of user studies, it validates the effectiveness of their VR platform to improve the users' skills in avoiding physical injury. Lang et al.[2] simulates a driving system that captures and analyzes the users' gaze behavior during the training. After being validated through the user study, it has been proved to be an approach capable of substituting the traditional behind-the-wheel driving training. On one hand, a tremendous amount of successful works showing that VR training has the potential to substitute traditional training and improve training efficiency. On other hand, as another branch of virtual training, VR exercise gaming is gaining more attention from the VR industry. Combined with haptic feedback from interactive devices, or with considering the interactions between the users and their physical environment, VR platforms can provide immersive visual effects at the same time delivers the users' desired exertion training effects. For example, sensing technologies can be used to analyze user's exertion experiences and have led to research into how to design creative exertion cards as hands-on practice tools for exergame designers[3]. A well-programmed drone can be designed as an intelligent flying robot accompanying joggers for interactive and engaging exertion activities[4]. By mapping a suspension kit that can suggest a range of body postures for various exercise styles[5] into the virtual environment, the kinesthetic force feedback from Pull-Ups can be immersively applied into three physical activities including kitesurfing, paragliding, and space invader in virtual reality. Therefore, as more successful works suggest that exergaming is a promising way to motivate people to exercise, there is a

growing trend to use interactive devices and VR technologies within the exergames industry. However, designing levels for exergames or designing personalized training is challenging, because it is highly non-trivial to design game levels or personalized training scenarios that carry desired exertion properties or training targets. The traditional exergaming or training program design process is tedious as it involves tuning a level or training scenario to satisfy multiple non-trivial constraints such as helping the user achieve a certain amount of exercise, ensuring the visual realism and aesthetic quality of the training content, and adjusting the exergaming or virtual training duration. Therefore, designing plausible exergame levels or personalized training scenarios typically requires many rounds of trials-and-errors. Give these observations, my intended research area is to devise optimization-based approaches for exergame level designers or virtual training designers to achieve the designs in a fast, scalable, and automatic manner.

1.1 Existing Work of Synthesizing Virtual Training Scenarios

Virtual training, same as or short for training via virtual reality, is a training method in which a simulated virtual environment is used. All virtual training process occurs in a virtual space that has been visualized through a head-mounted device such Oculus Rift[6]. Through accurate interactive devices such as VR controllers, users can virtually manipulate and interact with virtual objects or complete virtual tasks in the virtual environment as if it is happening in reality. As the interactive technologies of virtual reality tend to be realistic and immersive, virtual training becomes more likely to substitute real-world hands-on. As shown in Figure 1.1(a), Li et al.[1] provided an immersive and novel virtual reality training approach, which is making use of realistic common indoor virtual environments equipped with furniture, to teach individuals how to survive earthquakes through the virtual drills during which virtual earthquakes are simulated and users navigate in the virtual environments to avoid getting hurt while learning the observation and self-protection skills to survive an earthquake. In this section, as some examples, several interesting topics that are relevant to virtual training will be presented.



(a) Earthquake Safety Training [1].



(b) Driving Habits Training [2].

Figure 1.1: This figure shows two examples of virtual training: (a) Public safety training program design in virtual reality using Virtual Drills to simulate earthquakes and (b) Synthesized personalized training programs for improving driving habits via virtual reality.

As personal skill training is becoming more diverse, personalized training elements become more fundamental for various kind of professionals or skilled labor. Lack of skilled labor will result in cost overruns and the loss of productivity. Also, the profit margin is highly dependent on the training quality and efficiency. Poorly conducted training programs will be likely to cause the schedule overruns and declines in productivity. To reduce such unnecessary costs, skilled laborers must be trained in an efficient way. As a matter of fact, the traditional one-cut-fits-all training strategies are not suitable for labor with various different backgrounds. Therefore, the personalized one-to-one training program designs are necessary. But on the other side, authoring tons of different personalized training programs will be non-trivial for the training programmers. As manually design plausible personalized training scenarios typically requires many rounds of trials-and-errors. Emerging optimization technologies such as optimization for navigation graph-based way-finding algorithms proposed by Huang et al.[7], optimization for personal preferences-driven functional workspace scene layout synthesis proposed by Liang et al.[8], and optimization for

realistic furniture arrangement in indoor scenes proposed by Yu et al.[9] prove that stochastic optimization-based approaches are effective to synthesize realistic virtual scenarios efficiently and accurately. Employing these optimization approaches, recent works successfully synthesize personalized training scenarios for users to emphasize specific skills such as improving driving habits[2], as shown in Figure 1.1(b), an optimization-based approach is proposed by first identifying the improper driving habits of a user when he drives in a virtual city and then synthesizing a pertinent training program to help improve the users driving skills based on the discovered improper habits of the user. As personalized skill training is an open and general reach topic, it can be easy to derive from one specific research topic into multiple different interdisciplinary research topics that are incorporated with different domain-specific knowledge.

1.2 Existing Work of Synthesizing Exergame Level Designs

Exergaming, also called fitness game, or gamercising, is a term used for video games that are served for exercise purpose and that are served for encouraging people to exercise while playing the game. Given the human-computer interaction devices such as a depth sensor (e.g. Microsoft Kinect [10], PlayStation Camera) and motion controllers (e.g. Wii Remote, PlayStation Move, Nintendo Joy-Con) used for capturing player body movement, exergaming has been proven to show the positive effects of rehabilitation and therapy such as weight control, balance enhancement, and cognitive-motor training. For example, the effectiveness of the exergames with respect to helping adolescents and adults to achieve their weight loss was studied and validated by Bohm-Morawitz et al. [11] and Staiano et al. [12]. As shown in Figure 1.2, Xie et al. [13] and Zhang et al. [14] synthesizes exergames that enable people to burned desired calories. Padala et al. [15] and Wuest et al. [16] discovered that exergaming can effectively improve the elder’s balance and movement performance. Ogawa et al. [17], and Schoene et al. [18] have found that exergaming can improve both cognitive functions and dual-task functions.

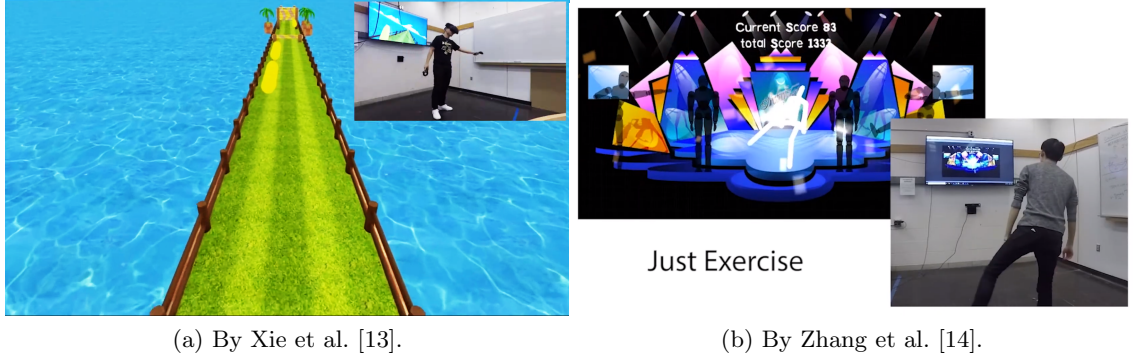


Figure 1.2: This figure shows two examples of synthesized exergames.

Synthesizing gaming content is typically achieved through procedure modeling techniques which can be employed to create 3D models and textures from sets of rules[19]. Procedure modeling techniques are widely applied to game design and development, from 2D platform game [20] to a first-person game[21], and now to motion-based game in human-computer interaction device [22]. Many research focuses on content quality, gameplay experience, and player skill as part of input parameters to generate game level automatically through algorithmic mean. For example, Shi et al. [23] combined learning based and rule-based method to dynamic adjust level difficulty in real-time. Along a similar direction, Jennings et al. [24] proposed a polymorphic, machine learning-based approach to adjust level difficulty based on player performance. Shaker et al. [25] propose a player-driven approach to create immersive game content.

All of those successful research studies suggest that it is possible to use a computational approach to synthesize personalized VR training scenarios or parameterized game content automatically, especially, when stochastic optimization-based approaches such as simulated annealing algorithms are employed, such synthesis can be efficient and accurate enough to satisfy users' expectations for the training programs or exertion games.

1.3 Problem Statements, Thesis Synopsis and Organizations

In this thesis, we aim at solving two main challenging problems through technical approaches, they are:

Problem 1. Is it possible to use a computational approach to synthesize personalized VR training scenarios automatically?

Problem 2. Are those personalized VR training programs synthesized with our approach better than traditional training programs?

Given these two problem statements, we have two corresponding hypotheses as theoretical supports for our thesis:

Hypothesis 1. Stochastic optimization-based approaches such as simulated annealing algorithms can be used to synthesize personalized VR training scenarios automatically.

Hypothesis 2. Statistic tests such as t-tests and ANOVA tests can tell us whether our synthesized personalized VR training programs are better than traditional training programs.

To summarize the thesis at hand, in view of the large amount of opportunities in novel applications in synthesizing virtual training scenarios and exergames, the dissertation thesis will focus on proposing optimization-based approaches to automatically synthesize the personalized training scenarios within several different interdisciplinary areas including:

1. Automatic Synthesis of Virtual Wheelchair Training Scenarios.
2. Exertion-Aware Path Generation.
3. Synthesizing Construction Safety Training Scenarios for VR Training.

Chapter 2 will introduce 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. The work presented in this chapter has been published as a conference paper [26] in 2020 IEEE Conference on Virtual Reality and 3D User Interfaces (IEEE VR 2020).

Chapter 3 will illustrate a novel approach for generating paths with desired exertion properties, which can be used for delivering highly realistic and immersive virtual reality applications that help users achieve exertion goals. Given a terrain as input, our optimization-based approach automatically generates feasible paths on the terrain which users can bike to perform body training in virtual reality. The approach considers exertion properties such as the total work and the perceived level of path difficulty in generating the paths. To verify our approach, we applied it to generate paths on a variety of terrains with different exertion targets and constraints. To conduct our user studies, we built an exercise bike whose force feedback was controlled by the elevation angle of the generated path over the terrain. The work presented in this chapter has been published as a conference paper [27] in ACM Transactions on Graphics (Proceeding of SIGGRAPH 2020).

Chapter 4 will present a novel approach to synthesize construction safety training scenarios to train the users on how to proficiently inspect the potential hazards within the construction sites in virtual reality. Given the specific training goals such as personalized preferences and estimated training time, we optimize the synthesized training scenarios through the simulated annealing approach. The work presented in this chapter will be submitted as a conference paper in 2022 IEEE Conference on Virtual Reality and 3D User Interfaces (IEEE VR 2022).

Chapter 5 will draw a conclusion on existing research results and prospect the future trend of personalized training in virtual reality and advanced technologies in optimizations of training designs. In this section, we will summarize the potential application area that might take advantage of optimizations-approaches to synthesize personalized training scenarios.

Chapter 2: Automatic Synthesis of Virtual Wheelchair Training Scenarios

2.1 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 [28]. 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 [29–31] 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 [29, 31].
- **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 [30].
- **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. Our simulation allows users to train themselves in multiple

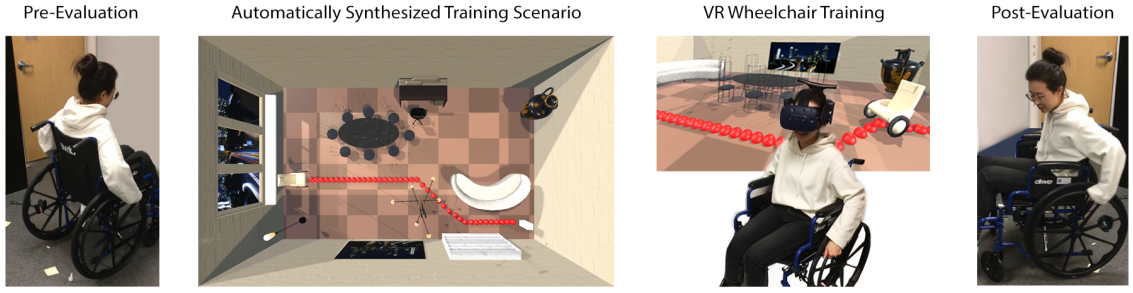


Figure 2.1: Overview of our approach. This figure shows the approach of our research. Pre-evaluation: before the VR training, the user completes a predesigned real-world task in the laboratory during which the user navigate the wheelchair along a specified path with their performance recorded. Automatically 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.

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.

2.2 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 [29]. However, when novice subjects used a wheelchair without guidance, there was no significant improvement over time [30]. 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 [32] 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 [33]. 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 [34–39], virtual reality driving training programs [2, 40, 41] and task-driven virtual reality trainings such as industrial maintenance trainings [42], mining industry skills practicing [43] and shooting simulation [44] 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 [1] and other natural disaster training [45]. 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 [2] and physical movement in exergames [14,46] 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 [47] or improving skills by allowing users to repeat tasks as long as they choose to [48]. Wheelchair simulation in VR can be used for either training or entertainment. Herrlich et. al. [49] 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 [50]. Rodriguez et al. [48] develop a wheelchair simulator that is able to help disabled children familiarize themselves with the wheelchair. Harrison et al. [51] offer the conclusion that virtual environments are potentially useful to train inexperienced powered wheelchair users. However, these simulations [48–50,52] 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, 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.

2.3 Overview

In this project, we propose an optimization-based approach for automatically generating scenarios for wheelchair training in virtual reality. As shown in Figure 2.1, we develop a program that will automatically 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 2.1 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.

2.4 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 [53] 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.

2.4.1 Cost Functions

Overall Cost. A training scenario consists of a scene and a path. Let scene graph [54] $S = (V, \Theta, E)$ 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 [55] 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 $C_{\text{total}}(P, S)$:

$$C_{\text{total}}(P, S) = \mathbf{C}_S \mathbf{w}_S^T + \mathbf{C}_P \mathbf{w}_P^T \quad (2.1)$$

where $\mathbf{C}_S = [C_S^d, C_S^r]$ 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 2.2. The C_S^r is the pairwise rotation cost of the objects in the scene defined in Equation 2.3. $\mathbf{C}_P = [C_P^d, C_P^r, C_P^n]$ 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 2.4. The C_P^r is the path rotation cost defined in Equation 2.5. The C_P^n is the path narrowness cost defined in the Equation 2.6.

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:

$$C_S^d(V, E) = \sum_{\forall (i, j) \in E} (\|\mathbf{v}_i - \mathbf{v}_j\| - d_{i, j})^2 \quad (2.2)$$

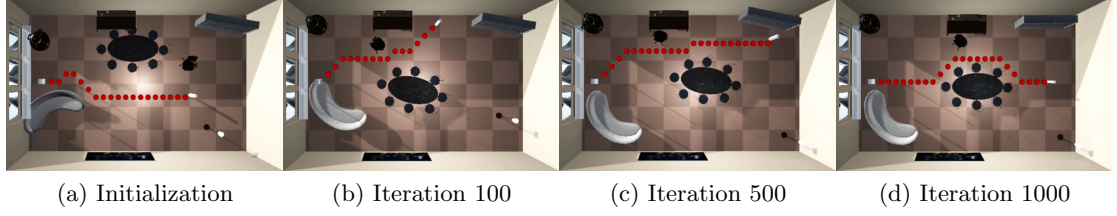


Figure 2.2: This figure shows an example of the optimization process. Figure (a) shows the initialization of the placement of the objects in the scene and a randomly generated path. Figure (b) (c) (d) shows a new frame in the optimization process based on the number of iterations completed at the time. 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.

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:

$$C_S^r(\Theta, E) = \sum_{\forall (i,j) \in E} (||\theta_i - \theta_j|| - \delta_{i,j})^2 \quad (2.3)$$

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

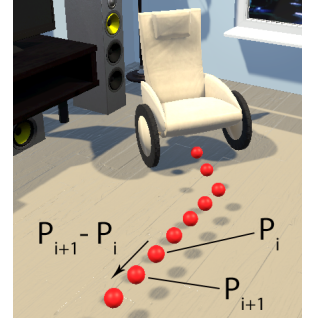


Figure 2.3: Distance.

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 $C_P^d(P)$:

$$C_P^d(P) = \left(\sum_{i=1}^{N-1} \|\mathbf{p}_{i+1} - \mathbf{p}_i\| - d_{\text{path}} \right)^2 \quad (2.4)$$

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} .

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

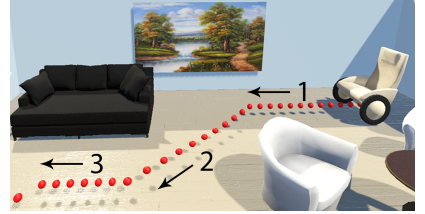


Figure 2.4: Rotation.

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 $C_P^r(P)$:

$$C_P^r(P) = \left(\sum_{i=2}^{N-1} \Gamma_i(P) - r_{\text{path}} \right)^2 \quad (2.5)$$

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 $C_P^n(P, S)$:

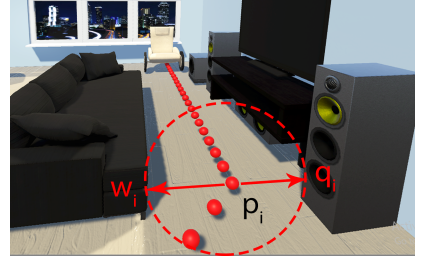


Figure 2.5: Narrowness.

$$C_P^n(P, S) = \frac{1}{N} \sum_{i=1}^N (\|\mathbf{p}_i - \mathbf{q}_i\| + \|\mathbf{p}_i - \mathbf{w}_i\| - n_{\text{path}})^2 \quad (2.6)$$

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 right side, \mathbf{w}_i is the closet object \mathbf{p}_i on the left side.

2.4.2 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 2.1. Positions of the objects in the scene are sampled randomly within the room to minimize the scene costs \mathbf{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 \mathbf{C}_P .

Path Computation. As illustrated in Figure 2.2, the optimization of the path is specified using three main costs, the distance of the path shown in Equation 2.4, the rotations of the path shown in Equation 2.5, and the narrowness shown in Equation 2.6. During the optimization, starting from the wheelchair, A* algorithm, a widely used pathfinding and graph traversing algorithm [56], is employed to generate a random path in each iteration.

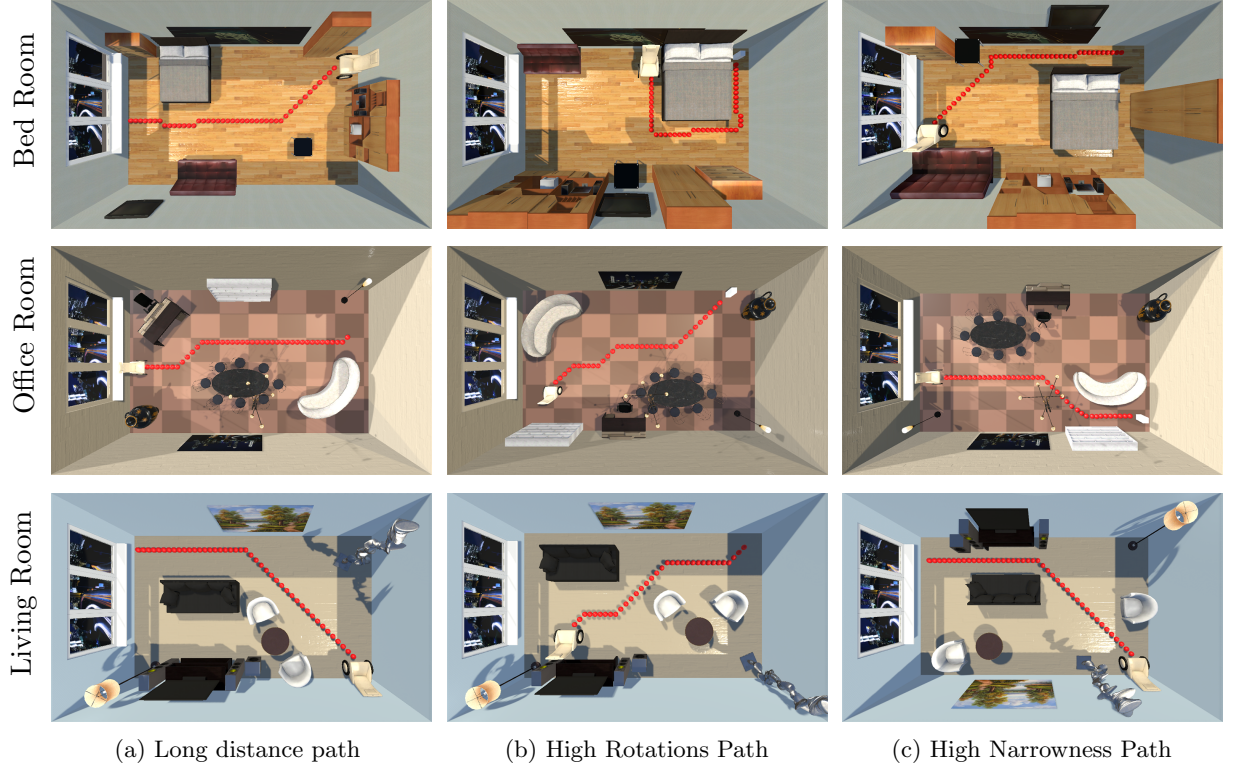


Figure 2.6: We have 9 different training scenarios including Column (a): long path training scenarios. Column (b): high rotations path training scenarios and Column (c): high narrowness training scenarios with different scene settings: First row: the bedrooms, second row: the office rooms and the third row: the living rooms.

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 [57] 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 [58] $f(t)$:

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

where t is the temperature parameter of the Metropolis-Hastings Algorithm [59]. 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 [59] 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 [60] 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 B.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 emphasizing different training skills. These terms are defined in Equation 2.1:

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

$r_{\text{path}} = 1$ with a weaker rotation weight $w_{\text{p}}^{\text{r}} = 0.2$.

- **Path Narrowness:** Column (c) has a smaller narrowness target $n_{\text{path}} = 10$ with a stronger narrowness weight $w_{\text{p}}^{\text{n}} = 0.6$. While Column (a) and (b) have a wider narrowness target $n_{\text{path}} = 20$ with a weaker narrowness weight $w_{\text{p}}^{\text{n}} = 0.2$.

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 B.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 3.6.

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 2.7, we apply the physics-based simulation [61] to simulate the different motions of the manual wheelchair [62] 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 :

$$\omega_*(t) = \cos^{-1} \frac{\mathbf{r}_*(t) \cdot \mathbf{r}_*(t - \Delta t)}{|\mathbf{r}_*(t)| \cdot |\mathbf{r}_*(t - \Delta t)|} \delta_*(t) \quad (2.7)$$

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:

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

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.

Given the wheels' angular speed in equation(2.7). Let R denotes the radius of the wheels, and, $\mathbf{s}(t)$, the linear speed of the scene is calculated as:

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

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

$$\alpha(t) = \begin{cases} \sin^{-1} \left(\frac{v_l(t)}{|c_l - c_r|} \right) \text{sgn}(\omega_l) & v_r \leq v_l \\ -\sin^{-1} \left(\frac{v_r(t)}{|c_l - c_r|} \right) \text{sgn}(\omega_r) & v_r > v_l \end{cases} \quad (2.8)$$

However, Equation 2.8 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:

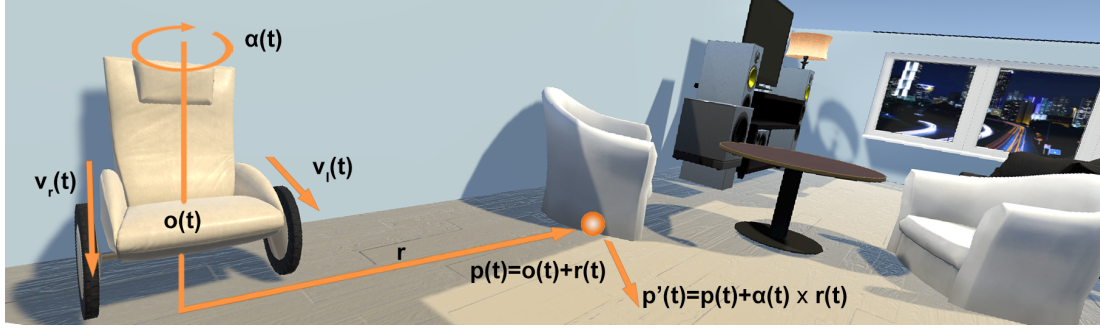


Figure 2.7: Physics-based wheelchair simulation: This figure shows the rotation of a scene relative to a stationary VR wheelchair according to the physics simulation. Two trackers are rotating as the user is pushing or pulling the wheels of the physical wheelchair. Let $\mathbf{o}(t)$ be the center of 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 $\mathbf{p}'(t) = \mathbf{p}(t) + \alpha(t) \times \mathbf{r}(t)$.

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

2.6 Experiments

2.6.1 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.

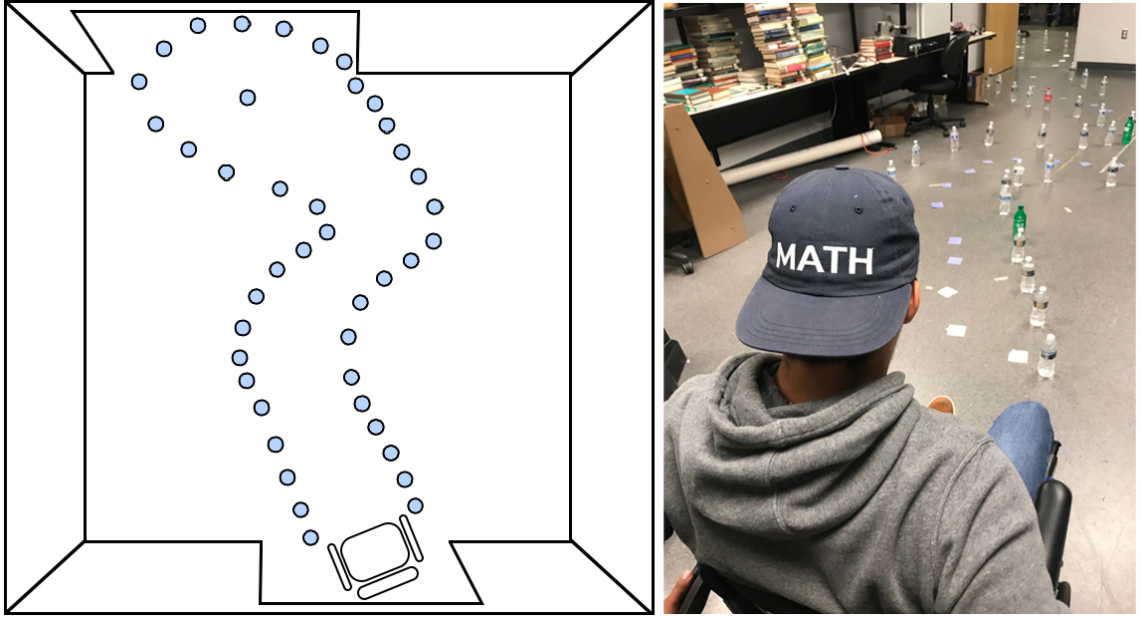


Figure 2.8: The real-world scenario used for evaluation. Left: Top-down view of the scenario with the path shown. Right: Photo of the scenario; water bottles were placed on the ground to measure collision.

2.6.2 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 2.4.2.
- **Post-evaluation:** The user ran through the same course as the pre-evaluation.

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.

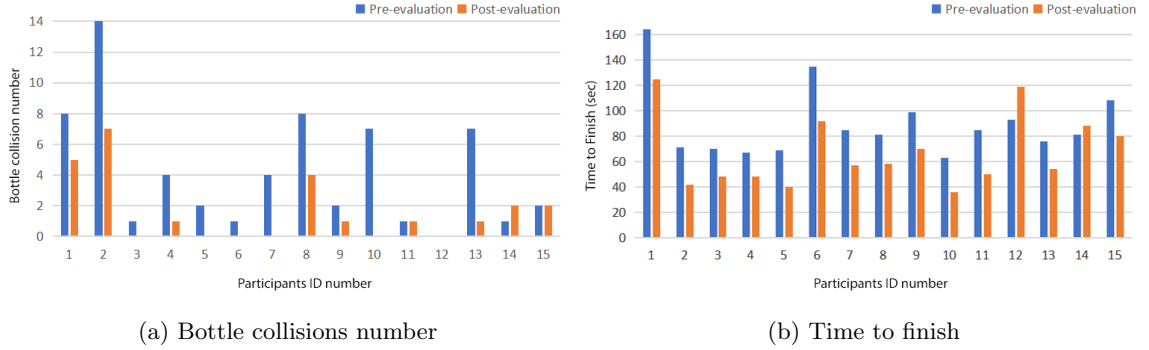


Figure 2.9: This figure shows the bar plots of pre-evaluation(blue) and post-evaluation(orange). There were 15 users in total. Figure (a) is the number of bottle collisions. Figure (b) is the time to finish the task for each user.

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 2.8. 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 2.4.2. The scenarios encompassed three different types of rooms with three path settings for each type of room as shown in Figure B.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 B.4, different types of paths varied in the number of rotations, path length, and path narrowness allowing for 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 2.5. 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

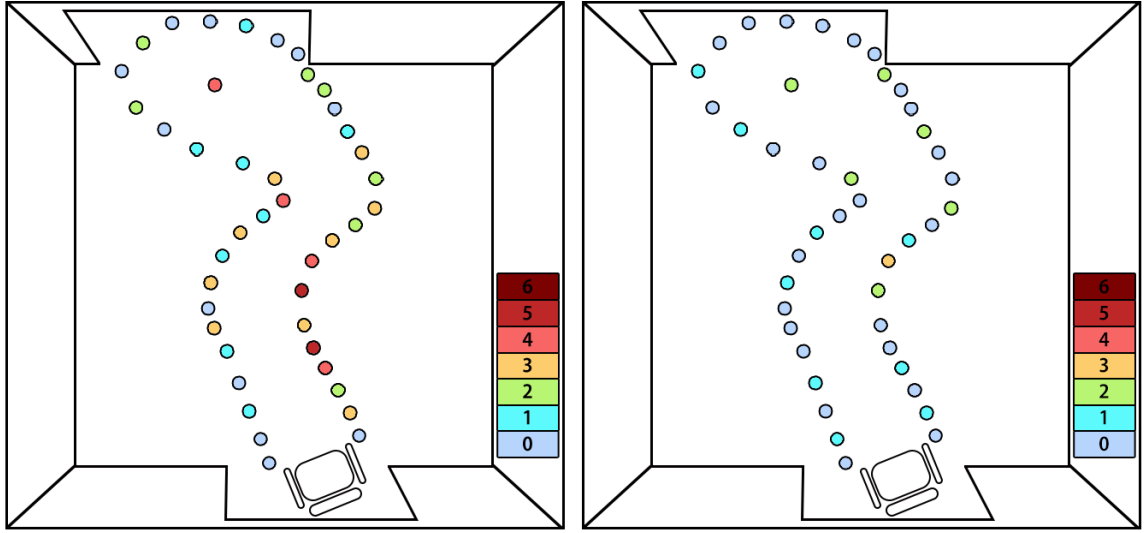


Figure 2.10: Visualizations of the number of the bottles knocked down during pre-evaluation (Left) and post-evaluation (Right). Colored circles are the places where the bottles have been knocked down. Colors show the frequency of that position where collisions happen. As the color goes from blue to red, the number of collisions increases.

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.

2.7 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 2.4.2, 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 2.9. 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.

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 [63, 64] 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 .

Bottle collision factor measures how precisely the user can control the wheelchair along a given path. As shown in Figure 2.10, 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 2.11, 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 2.12. 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.

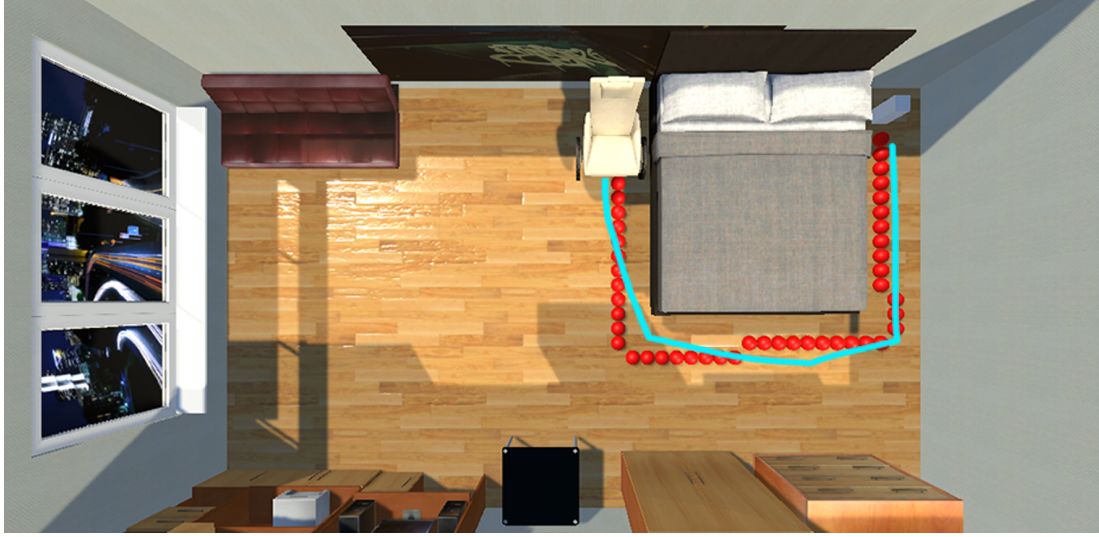


Figure 2.11: User’s path compared with the generated path: This figure shows that during the VR training, users are trying to follow the generated path, where the blue curve is the user’s path automatically recorded by the computer, the red dotted curve is the generated path. The significant consistency 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.

2.8 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 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

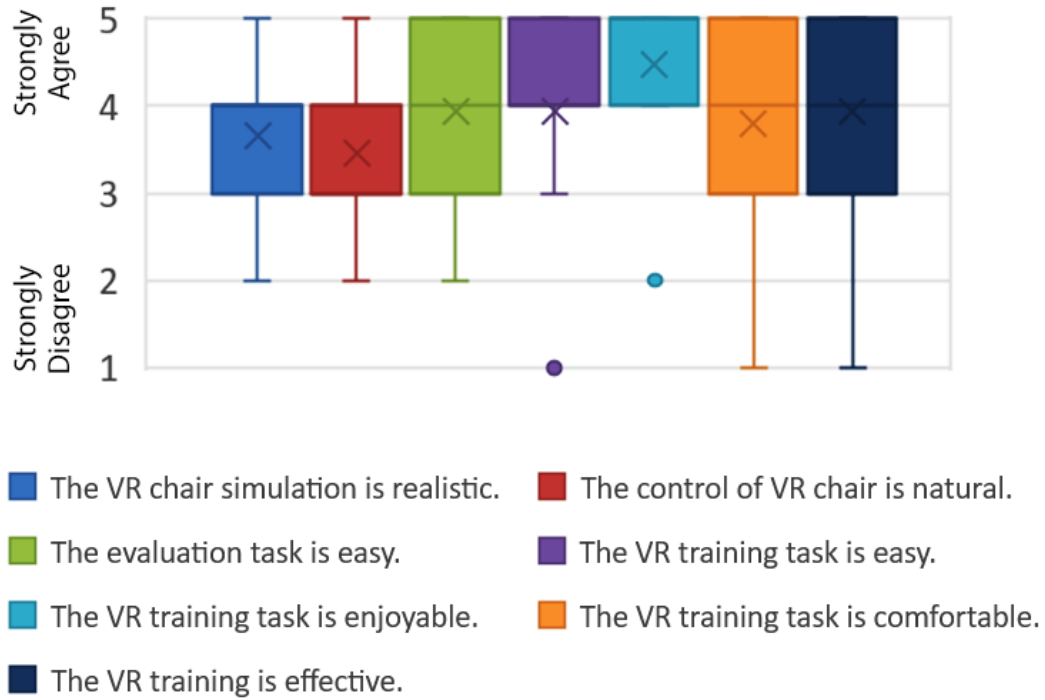


Figure 2.12: This figure shows the feedback of 15 users after the nine different VR training scenarios. Different colors represent different questions in the questionnaire. Different numbers indicate the extent to which the users agree. 1 indicates strongly disagrees while 5 indicates strongly agrees. The results draw a positive conclusion that most users agree that their performance in maneuver a wheelchair improved after the VR training.

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.

Chapter 3: Exertion-Aware Path Generation

3.1 Introduction

“The ultimate display would, of course, be a room within which the computer can control the existence of matter. A chair displayed in such a room would be good enough to sit in. Handcuffs displayed in such a room would be confining, and a bullet displayed in such a room would be fatal.”

— Ivan Sutherland, 1965

Ivan Sutherland, inventor of the first virtual reality (VR) head-mounted display, envisioned that the ultimate display would deliver virtual contents that are highly realistic in terms of visuals and haptics. The tremendous efforts of computer graphics researchers have led to the invention of generative models and procedural modeling algorithms capable of synthesizing visually stunning virtual contents such as streets [65], roads [66–68], terrains [69, 70], cities [71, 72], and worlds [73]. However, as these virtual contents are mainly employed for visualization purposes, haptic properties such as the force feedback afforded by these virtual contents are typically not considered in their synthesis.

In this chapter, we propose to generate virtual content with the consideration of haptic properties afforded by the content. More specifically, we propose an optimization-based approach to generate paths on a terrain that are both visually realistic and associated with desired exertion properties. Users can experience the generated paths in virtual reality through riding an exercise bike whose resistance is adjusted according to the elevation angle of the paths. The users’ haptic feelings of the paths match with their visual perception, lead to enhanced realism and immersiveness. For example, they would feel that it is physically more demanding to ride uphill than downhill as the bike’s resistance increases with the elevation angle of the generated paths. After riding the generated paths in virtual reality,

users will also accomplish their exertion goals (in terms of total work done). Leveraging our exertion-aware paths and setup, biking in a virtual world resembles biking in the real world.

Our approach is motivated by the widespread popularity of exergames: a genre of games that prompt players to exercise. Successful exergame titles include Just Dance, Your Shape Fitness Evolved, and EA Sports Active, etc. With the growing popularity of consumer-grade virtual reality devices, a recent trend is to devise VR-based exergames as epitomized by recent VR exergame titles such as BoxVR, Ninja Legends, and Beat Saber. Following this trend, NordicTrack, a major gym equipment manufacturer, has introduced a VR bike that allows users to work out while biking in a virtual environment.

One major challenge in developing exergames is game level design, which is typically done manually by level designers under current practice. The design process is tedious as it involves tuning a level to satisfy multiple non-trivial goals and constraints such as helping the player achieve a certain amount of exercise, ensuring the visual realism and aesthetic quality of the game level, and constraining the game duration. Creating a plausible level typically requires many rounds of trials-and-errors. By solving the exergame level design problem as an optimization problem, our approach enables game level designers to generate exergame levels, paths with desirable exertion effects in our case, in a fast, scalable, and automatic manner. Our computational design framework also provides ample and explicit controls to designers to generate paths satisfying certain design preferences by specifying constraints. The major contributions of our work include:

- Proposing a novel problem statement of generating paths with desired exertion properties over a terrain, which can be employed for immersive virtual reality applications;
- Devising an optimization-based approach to automatically generate exertion-aware paths that help users achieve exertion goals. A generated path can be applied to control the force feedback of an exercise bike such that a user’s haptic experience matches with his visual perception as he bikes along the path;

- Conducting user studies to validate the enhanced user experience and exertion effects brought by the generated paths.

3.2 Related Work

We briefly review literature relevant to our problem statement of generating paths with the consideration of force feedback properties.

3.2.1 Procedural Modeling

Procedural modeling is widely applied to generate virtual world content such as terrains, rivers, roads, buildings, and cities in a scalable manner, reducing the cost of production by manual design. We refer our readers to surveys of procedural virtual world modeling [74] and terrain modeling [75].

For road generation, Galin et al. [68] encoded the slope of a specified terrain as well as natural obstacles as part of their objective cost functions that were procedurally minimized. Along a similar direction, Galin et al. [76] procedurally generated hierarchical road networks that connected cities, villages, and towns given a large terrain; while Beneš et al. [66] applied traffic simulations in the neighborhood to generate major roads in an urban setting. Moreover, Nishida et al. [67] proposed an example-based approach to incorporate user input with its underlying terrain to generate and adapt roads in existing road networks. These research efforts mainly focus on the interaction among the urban and natural components on the given terrain instead of the human experience in navigating the generated roads. Our work focuses on the player’s physical experience of the generated paths as they ride along the paths in virtual reality with a device (e.g., an exercise bike) capable of providing force feedback.

Procedure generation techniques have been applied for game design and development, from 2D platform games [20] to first-person games [77], and recently also to a ski-slalom game played on Wii [22] for physical rehabilitation.

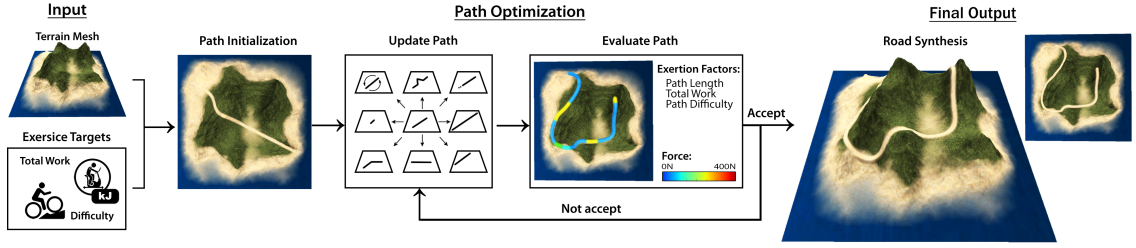


Figure 3.1: Overview of our approach.

While some research efforts focus on generating virtual worlds for a game, others consider gameplay experience and the player’s skills for adjusting game levels automatically through algorithmic means. For example, Shi et al. [23] and Jennings-Teats et al. [24] proposed rule-based methods to dynamically adjust game level difficulty based on the real-time performance of the player. Similarly, Hooshyar et al. [78] applied a data-driven approach to consider an individual’s skills in generating an educational game. For a comprehensive review on game content generation, please refer to the survey by Hendrikx et al. [79]. In contrast, our work focuses on considering the physical exertion and immersive experience of the player in procedural game generation.

3.2.2 Exergaming

Exergames refer to a genre of games that encourage people to exercise while playing a game. Exergames use motion sensing devices such as a depth sensor (e.g. Microsoft Kinect, PlayStation Camera) or a motion controller (e.g. Wii Remote, PlayStation Move, Nintendo Joy-Con) to capture body movement. The genre has gradually evolved from console exergames such as Microsoft Fitness and Nintendo Wii Sports to VR platform games and games played on sport equipment (e.g., VirZOOM bike). The use of motion sensing devices for game design enables the tracking and incorporation of the player’s motion and physical experience in the game mechanics, making exergames engaging as an alternative form of exercises. For example, ICAROS [80] enables players to train different muscle groups to

improve their reflex, balance, and coordination using fitness equipment while wearing a VR headset. VirZoom and CSE Entertainment developed exergaming systems that allow players to bike in virtual scenes by wearing a VR headset when using an exercise bike.

Furthermore, research has shown the positive effects of exergaming for rehabilitation and therapy such as weight control, balance enhancement, and cognitive-motor training. Bohm-Morawitz et al. [11] and Staiano et al. [81] studied the use of exergames to help adolescents and adults to achieve their weight loss. Padala et al. [15] and Wüest et al. [16] discovered that exergaming can effectively improve an elder’s balance and movement performance. Ogawa et al. [17] and Schoene et al. [18] found that exergaming can improve cognitive and dual-task functions. Mueller et al. [82,83] provided guidelines of designing exergames. One of the important remarks for the success of previous research is integrating exercises with virtual content to motivate players to achieve exercise goals.

However, most of the existing exergame content and settings are manually constructed and tested. Such a trial-and-error approach is time-consuming and expensive. Inspired by previous works [14, 46] on assembling exergaming levels by considering human poses, we propose a computational design framework that enables path-finding on a terrain to achieve desired exertion goals. In addition, we create an engaging exertion experience that allows players to bike the generated paths in virtual reality while training their lower body strength.

3.2.3 Immersive Virtual Environments

Immersiveness is an important factor to consider in designing virtual reality content. Previous research [84] found that presence, a form of immersion, is highly connected to a user’s enjoyment level in a virtual reality setting. Therefore, to create a realistic virtual environment, researchers have investigated the visual [85], sound [86, 87], and locomotion [88, 89] aspects of a virtual environment to enhance the sense of presence for virtual experience. In particular, research on creating immersive biking experience has focused on the cycling aspect of speed [90], force feedback control [91], and road surface texture [92]. However,

the virtual environments of these works were either manually created or did not consider the physical aspects of a player in the virtual biking experience. In contrast, our approach considered the exertion and perception aspects of the virtual training experience such as the total work done and the perceived difficulty of the path as optimization terms in the path generation. Considering the physical interactions such as the force feedback that the virtual content affords in VR will make the virtual training experience more realistic, immersive, and hence more enjoyable and engaging.

3.2.4 Pathfinding

Tremendous amount of research on pathfinding has been conducted to overcome challenges in applications such as robotic simulation, traffic simulation and games. Algorithms based on graph search (e.g., A*), sampling (e.g., RRT), agents, and optimization, etc. have been devised for pathfinding and motion planning.

For example, Wilkie et al. [93] used the first-in-first-out property of traffic to enable A* search algorithm to plan for large-scale vehicle routes. Karaman et al. [94] used the anytime property of RRT* to improve efficiency during online motion planning. Best et al. [95] proposed AutonoVi, a optimization-based maneuver planning for autonomous vehicle navigation that supports dynamic maneuvers. Huang et al. [96] proposed an agent-based approach to consider human perception and wayfinding for generating paths. Nathan et al. [97] proposed and applied a Partial-Refinement A* (PRA*) approach for pathfinding in real-time strategy games. Refer to the comprehensive reviews by Algfoor et al. [98] and Gonzalez et al. [99] on pathfinding and motion planning.

Pathfinding in an open terrain is often assumed to be isotropic; the cost of the path is independent of the traveling direction. There are some research works that investigate anisotropic pathfinding. Inspired by Galin et al. [68], we uniformly discretize the terrain and assume our path generation problem is anisotropic such that the costs associated with the generated path depend on the traveling direction. Such costs associate physical exertion effects with geometric information such as the elevation of the path in the optimization.

3.3 Overview

Figure 3.1 shows an overview of our approach. Given a terrain, a total work target, and a target level of perceived path difficulty as input, the goal is to generate an exertion-aware path over the terrain. The user can ride an exercise bike along the generated path whose elevation angle will be used to adjust the force feedback of the bike based on a physical model, such that the user’s haptic feeling matches with his visual perception of the path. Upon completing the path, the user will have also achieved the target amount of exertion (i.e., achieved the total work target).

The core of our approach is an optimization framework. The total work target and the target level of perceived path difficulty are encoded as optimization goals. The optimization proceeds iteratively. At each iteration, the generated path will be evaluated in terms of the exertion it will induce to the user biking along the path. It will also be evaluated in terms of the perceived difficulty of the path predicted by a classifier trained with a dataset of real bike paths annotated with difficulty ratings. The optimization proceeds until a path that satisfies all the desired properties is generated. Our approach then synthesizes a road surface along the generated path and slightly refines the terrain to make the cross-section of the road surface horizontal for a natural riding experience.

The user can then bike along the generated path to achieve his exertion goal, using a VR bike whose force feedback is controlled by the elevation angle of the generated path.

3.4 Technical Approach

3.4.1 Problem Formulation

Representation.

Figure 3.2 depicts the representation of our problem. The goal of our approach is to generate a path \mathcal{P} on the input 3D terrain by optimization to satisfy the desired properties.

Akin to Galin et al. [68], on the xz -plane we construct a 2D Hermite curve \mathcal{R} which is mapped to the 3D terrain to generate the path \mathcal{P} . The xz -plane is overlaid with an $N \times N$ grid called path grid \mathbf{M} . The 2D Hermite curve \mathcal{R} is constructed by a sequence of control points. Each control point is located at a grid intersection as shown in Figure 3.2. For notation convenience, we also use \mathbf{M} to denote an $N \times N$ occupancy matrix of the path grid.

An entry of \mathbf{M} stores the index of a control point that is located at the entry's corresponding grid intersection on the xz -plane; otherwise it stores a value of -1 to indicate that no control point is located there. We use $N = 20$ by default in our implementation.

The 2D Hermite curve \mathcal{R} is C^1 continuous. We parameterize curve \mathcal{R} such that a point on curve \mathcal{R} is denoted by $\mathbf{r}(t)$, where $t \in [0, 1]$. $\mathbf{r}(0)$ and $\mathbf{r}(1)$ refer to the start point and end point of curve \mathcal{R} .

The 2D curve \mathcal{R} on the xz -plane is mapped to the path \mathcal{P} on the surface of the 3D terrain using a height map function H of the terrain. Every 2D point $(\mathbf{r}_x(t), \mathbf{r}_z(t))$ on curve \mathcal{R} on the xz -plane has its corresponding elevation $H(\mathbf{r}_x(t), \mathbf{r}_z(t))$ on the 3D terrain. Upon mapping and parameterization, the path \mathcal{P} on the 3D terrain is given by $\mathbf{p}(t) = (\mathbf{r}_x(t), H(\mathbf{r}_x(t), \mathbf{r}_z(t)), \mathbf{r}_z(t))$, where $t \in [0, 1]$. $\mathbf{p}(0)$ and $\mathbf{p}(1)$ refer to the start point and end point of path \mathcal{P} .

Optimization Objective.

The goal of the optimization is to find a path \mathcal{P} that minimizes the total cost function $C_{\text{total}}(\mathcal{P})$ which evaluates the overall error of the path \mathcal{P} generated through path grid \mathbf{M} . The total cost function $C_{\text{total}}(\mathcal{P})$ is defined as:

$$C_{\text{total}}(\mathcal{P}) = w_L C_L(\mathcal{P}) + w_W C_W(\mathcal{P}) + w_D C_D(\mathcal{P}), \quad (3.1)$$

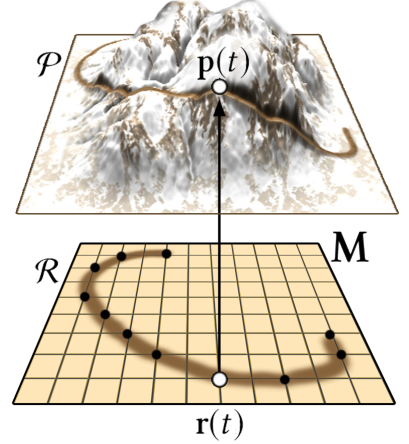


Figure 3.2: Problem representation. Black dots refer to curve \mathcal{R} 's control points.

where the path length cost $C_L(\mathcal{P})$ encodes the prior path length; the total work cost $C_W(\mathcal{P})$ encodes the target amount of total work induced on the user for completing path \mathcal{P} on a VR bike; and the path difficulty cost $C_D(\mathcal{P})$ encodes the target level of perceived path difficulty with respect to the variation of the path elevation throughout the biking experience. The w_L , w_W , and w_D represent the respective blending weights of these three cost terms. We provide details of the costs in the following section.

3.4.2 Cost Terms

Our optimization framework considers a number of cost terms to guide the generation of a path to carry certain exertion-related properties. While our framework is extensible to incorporate additional considerations and constraints, we focus on three cost terms related to the exercise duration, how much total work the generated path induces on the user, and the perceived level of difficulty during the biking experience. Interestingly, people usually consider factors like these when choosing a bike path in the real world to exercise.

Path Length Cost.

Different players may prefer different exercise duration for achieving different training goals. Given a certain biking speed, the duration of the exercise depends on the total length of the path. To adjust the length of the path, we introduce the path length cost which measures the difference between the length $L(\mathcal{P})$ of the generated path and a path length target ρ_L . Path length cost $C_L(\mathcal{P})$ is defined as:

$$C_L(\mathcal{P}) = 1 - \exp\left(-\left(\frac{L(\mathcal{P}) - \rho_L}{\sigma_L}\right)^2\right), \quad (3.2)$$

where we empirically set $\sigma_L = 2\rho_L$. $L(\mathcal{P})$ is the length of path \mathcal{P} calculated through line integral:

$$L(\mathcal{P}) = \int_{\mathcal{P}} d\mathbf{p} = \int_0^1 |\mathbf{p}'(t)| dt \quad (3.3)$$

Total Work Cost.

Another important consideration is the total work done for completing path \mathcal{P} by riding on a VR bike. Different types of exercises induce different total works. For example, cardiovascular exercises typically expend more energy than weight-training exercises do per training session [100]. It is challenging for a level designer to manually tune a level (a path in our case) to associate it with an arbitrary total work target as it involves non-trivial physics considerations of the force and movement of the user during the exergame.

To incorporate such a consideration, we formulate the expected total work $W(\mathcal{P})$ for a generated path \mathcal{P} according to a physics model and compare it with the desired total work target ρ_W . The total work cost $C_W(\mathcal{P})$ is defined as:

$$C_W(\mathcal{P}) = 1 - \exp\left(-\left(\frac{W(\mathcal{P}) - \rho_W}{\sigma_W}\right)^2\right), \quad (3.4)$$

where we empirically set $\sigma_W = 2\rho_W$. $W(\mathcal{P})$ is the total work of completing path \mathcal{P} calculated through line integral:

$$W(\mathcal{P}) = \lambda \int_{\mathcal{P}} f(t) d\mathbf{p} = \lambda \int_0^1 f(t) |\mathbf{p}'(t)| dt, \quad (3.5)$$

where $f(t)$ is the resistance force experienced by the user along the path $\mathbf{p}(t)$ as the path elevation angle $\theta(t)$ changes. From the directional derivative of the height map function H , we have $\theta(t) = \tan^{-1}\left(\frac{d\mathbf{r}(t)}{dt} \cdot \nabla H(\mathbf{r}(\mathbf{t}))\right)$.

λ is a parameter for scaling the distance of the path in the virtual world down to the real moving distance of a pedal of the exercise bike that the user rides. Suppose a path in the virtual world is 1,000m long, with $\lambda = 0.1$ set by default, the user will just need to ride

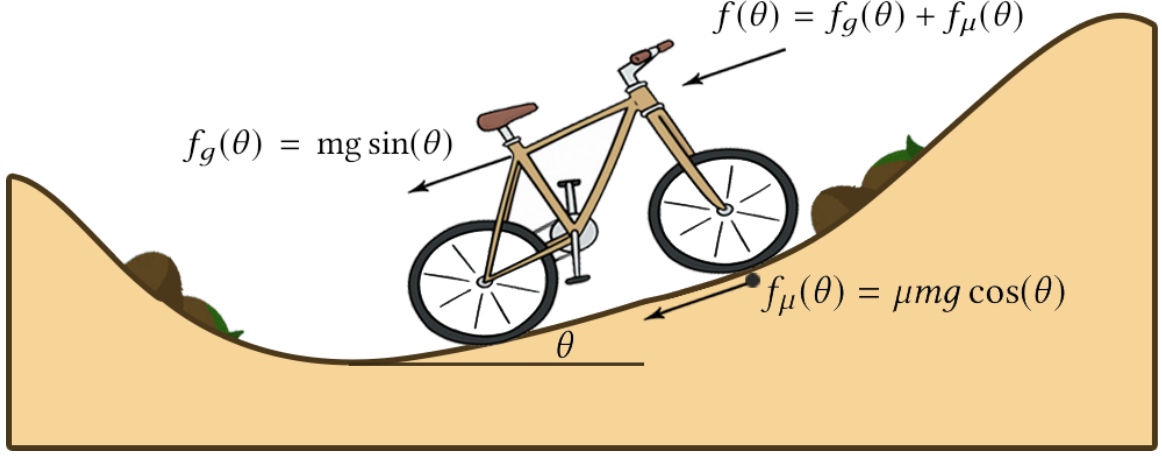


Figure 3.3: Resistance force. A bike experiences a resistance force f which varies with the path elevation angle θ . The resistance force is composed of the gravitational force f_g and frictional force f_μ .

100m on the exercise bike to complete the path.

According to the physics model shown in Figure 3.3, a bike traveling on a slope would experience a resistance force f composed of the gravitational force f_g and the frictional force f_μ . The resistance force varies with the path elevation angle θ . By decomposition of forces, it is calculated as:

$$f(\theta) = mg \sin(\theta) + \mu mg \cos(\theta), \quad (3.6)$$

where m is the mass; g is the gravitational constant; and μ is the friction coefficient of the terrain surface.

In practice, the resistance force $f(t)$ in (3.5) is simulated by the feedback force of an exercise bike that the user rides. The exercise bike can give a discrete set of feedback forces $F = \{f_i\}$ through its pedals. Figure 3.4 (bottom) shows the feedback forces $\{f_1, \dots, f_6\}$ of the exercise bike used in our experiments, which are designed by the bike's manufacturer.

For simplicity, we assume mass $m = 70\text{kg}$ which is slightly above the average global body mass [101]. We want to associate biking on a horizontal path (i.e., $\theta = 0$) with a small feedback force (f_2) of the exercise bike. In this case, the user only experiences the frictional

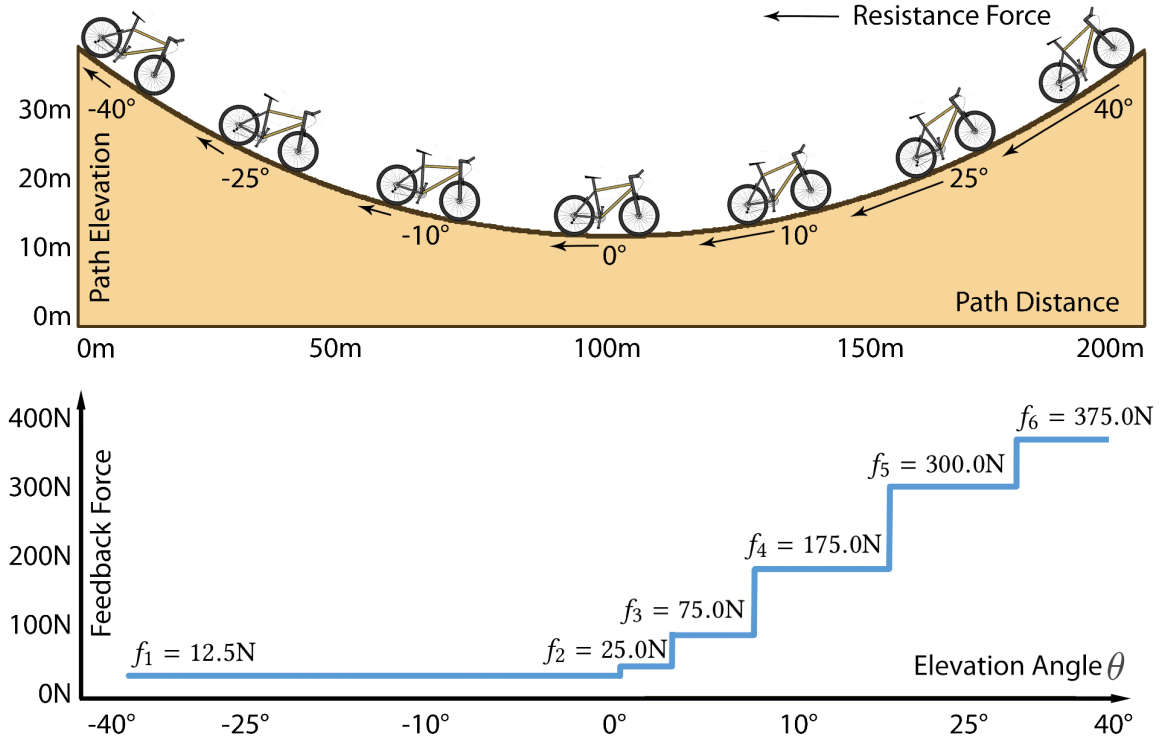


Figure 3.4: Top: A path whose elevation angle changes smoothly from -40° to 40° . Bottom: the force feedback of the exercise bike used to approximate the resistance force experienced at the corresponding elevation angle.

force as the resistance force according to Figure 3.3. Using (3.6), we calculate the friction coefficient $\mu = \frac{f_2}{mg}$.

As the user bikes along the path $\mathbf{p}(t)$, we compute the resistance force $f(\theta)$ from (3.6) that he should experience in reality according to the physics model. Then a feedback force $f_i \in F$ that is closest to $f(\theta)$ is given by the exercise bike to approximate the resistance force $f(\theta)$. Essentially, the user experiences a feedback force f_i that increases with the path elevation angle realistically.

As the resistance force $f(t)$ is simulated by the feedback force f_i of the exercise bike, for computing the total work cost $C_W(\mathcal{P})$, we set $f(t) = f_i$ to calculate the total work $W(\mathcal{P})$ in (3.5).

Figure 3.4 depicts the feedback force assigned according to the path elevation angle.

For going uphill, we consider elevation angle up to 40° that corresponds to a 84% gradient and is very challenging [102]. For going downhill ($\theta < 0^\circ$), the user experiences a net force pushing him downwards according to (3.6). We assume that the user only rides using a very small force in this case that corresponds to the minimum feedback force f_1 of the exercise bike.

Note that we could also associate biking on a horizontal path with a feedback force other than f_2 of the exercise bike. For example, if we want to simulate biking on a rough surface (e.g., a muddy road) with a high friction, we could associate it with a higher feedback force level (e.g., f_3). In this case, we will obtain a larger μ and the user will experience a larger simulated frictional force.

Path Difficulty Cost.

The bumpiness of a path is related to the perceived level of biking difficulty. For example, a trail on a bumpy mountain is usually perceived as more difficult to bike than a trail on a plain. Websites such as the MTB Project collect people’s ratings of the biking difficulty of different trails.

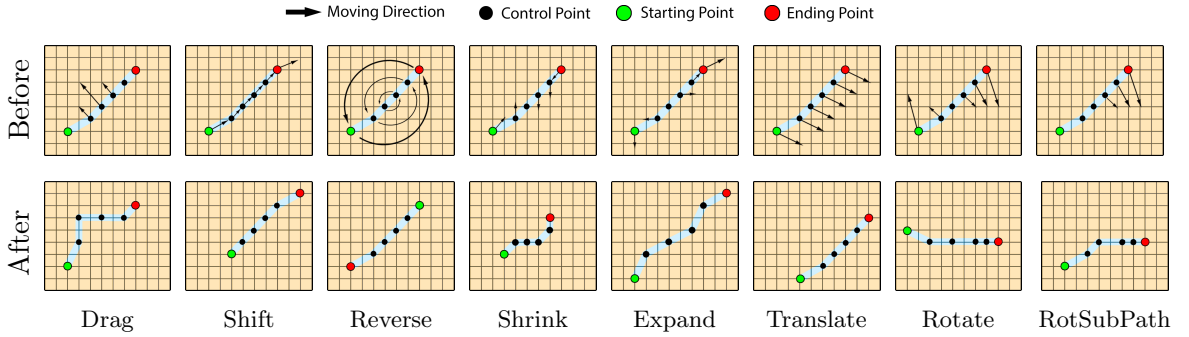


Figure 3.5: Move strategies. Each move is applied to the control points of the 2D Hermite curve \mathcal{R} on the path grid. The arrows indicate the moving directions of the control points. Each move modifies curve \mathcal{R} , which in turn modifies path \mathcal{P} on the 3D terrain.

As the perception of biking difficulty is subjective, we formulate the path difficulty cost by using a classifier trained with difficulty ratings of real-world trails given by bikers. This cost measures the difference between the perceived difficulty $D(\mathcal{P})$ of the generated path

and the desired path difficulty target ρ_D . It is defined as:

$$C_D(\mathcal{P}) = 1 - \exp\left(-\left(\frac{D(\mathcal{P}) - \rho_D}{\sigma_D}\right)^2\right), \quad (3.7)$$

where we empirically set $\sigma_D = 0.25$ in our implementation. $D(\mathcal{P}) \in [0, 1]$ is the perceived difficulty of path \mathcal{P} evaluated using the classifier, with a larger value referring to a more difficult path.

To train a classifier to predict the difficulty of a path, we collected more than 30,000 paths from the MTB Project contributed by passionate bikers excited to share their knowledge of local biking trails with others. The difficulty of a path is voted by MTB users. The difficulty ratings range from 1 to 6, with 1 meaning easy and 6 meaning hard. The number of votes for a path varies from 1 to 100. We dropped the paths with fewer than 10 votes.

We computed the average difficulty of each path using its votes. We then grouped the paths into two difficulty groups: easy and hard. The easy group contains paths with an average difficulty rating lower than 3; the hard group contains paths with an average difficulty rating higher than 4. We ignored paths with an average difficulty rating between 3 and 4 as their difficulty levels are ambiguous. Finally, we obtained 1,000 easy paths and 1,000 hard paths.

Each collected path is encoded as a sequence of sample points denoting the elevation along the path. The number of sample points of a path varies from 100 to 1,000. A sample point is taken every 0.1 mile approximately. Figure 3.6 visualizes the two groups of paths. Visually, the hard paths tend to be bumpier than the easy paths.

As the paths have different lengths, to align the paths for training, we extracted 11 features from each path:

- Path length;
- Highest and lowest elevations of the path;
- Sum of ascent (elevation increase) along the path;

- Sum of descent (elevation decrease) along the path;
- Minimum and maximum curvature;
- Accumulative curvature;
- Number of peaks and valleys;
- Variance in the elevations of the sample points of the path.

We trained a feedforward artificial neural network for predicting the difficulty of a path based on its features. Each feature was normalized to $[0, 1]$. The neural network has 1 hidden and fully-connected layer with 15 hidden units. The neural network has 11 trainable parameters. It uses RELU as the activation function and SGD as the loss function. We randomly selected 500 paths from each difficulty group to construct the training dataset, and used the remaining paths as the testing data. The prediction accuracy of our classifier is about 82%. Each prediction also gives a probability that indicates how likely the input path is classified into the hard group. For example, if the probability of a given path is 0.2, it means that the path has a 20% chance to be classified as a hard path and a 80% chance to be classified as an easy path. We use this probability as a metric $D(\mathcal{P})$ to evaluate the difficulty of a generated path \mathcal{P} . Our classifier is able to correctly classify 86% of the easy paths and 78% of the hard paths. We used a neural network as our classifier it achieved a higher prediction accuracy and robustness compared to alternative classifiers such as random forests and SVM in our preliminary tests.

3.4.3 Path Optimization

Initialization.

We initialize the path grid \mathbf{M} to generate a random line, which contains a sequence of control points of a 2D Hermite curve, near the center of the grid. To ensure that the random line stays within the terrain boundary, we consider both the target path length ρ_L and the terrain width ω . The coordinates of the center (x_c, y_c) of the line are randomly initialized in the

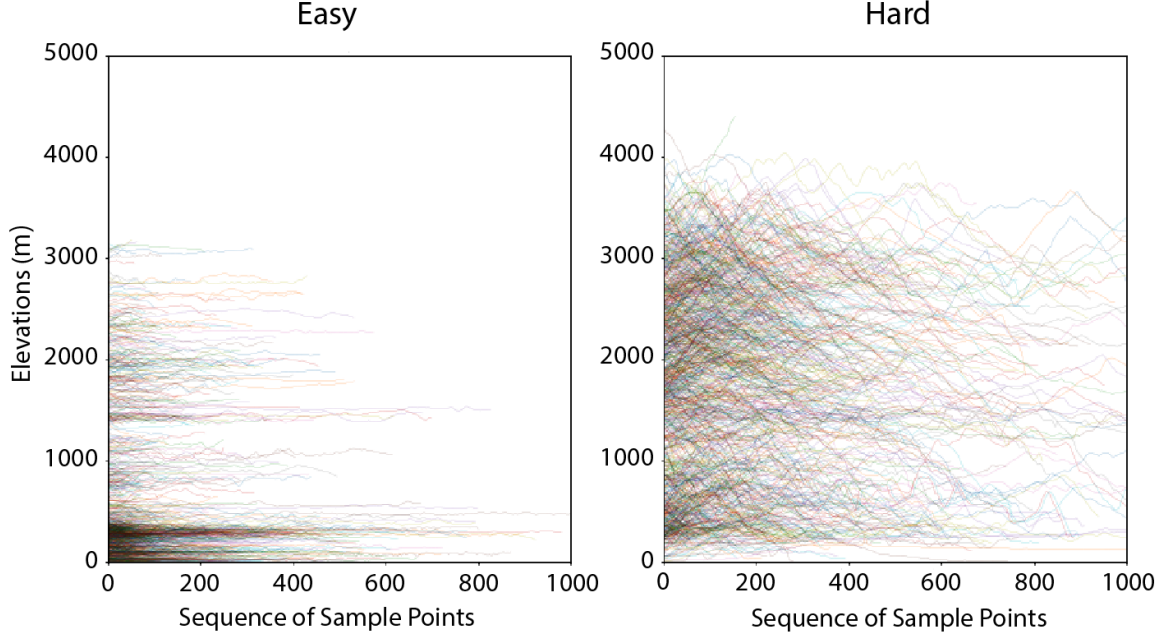


Figure 3.6: Elevations of the paths from the easy and hard groups. The paths correspond to real-world trails of the MTB Project. The elevation along each path is shown. The hard paths are generally bumpier than the easy paths.

range $[\frac{\rho_L}{2}, \omega - \frac{\rho_L}{2}]$. By selecting a random angle $\phi \in [0, \pi]$, we generate the start point of the line as $(x_c - \frac{\rho_L}{2} \cos(\phi), x_c - \frac{\rho_L}{2} \sin(\phi))$ and the end point as $(x_c + \frac{\rho_L}{2} \cos(\phi), x_c + \frac{\rho_L}{2} \sin(\phi))$. The control points on this line are then used to construct a 2D Hermite curve \mathcal{R} that is mapped to the 3D terrain to generate an initial path \mathcal{P} as described in Section 3.4.1.

Process.

The optimization proceeds iteratively. At each iteration, our approach applies a move to modify the control points of the 2D Hermite curve \mathcal{R} so as to modify curve \mathcal{R} and the path \mathcal{P} on the 3D terrain. The move is randomly selected from one of the eight different types of moves shown in Figure 3.5:

- **Drag:** Some control points in the middle are dragged aside by a random amount so that the curve is dragged like a spring.

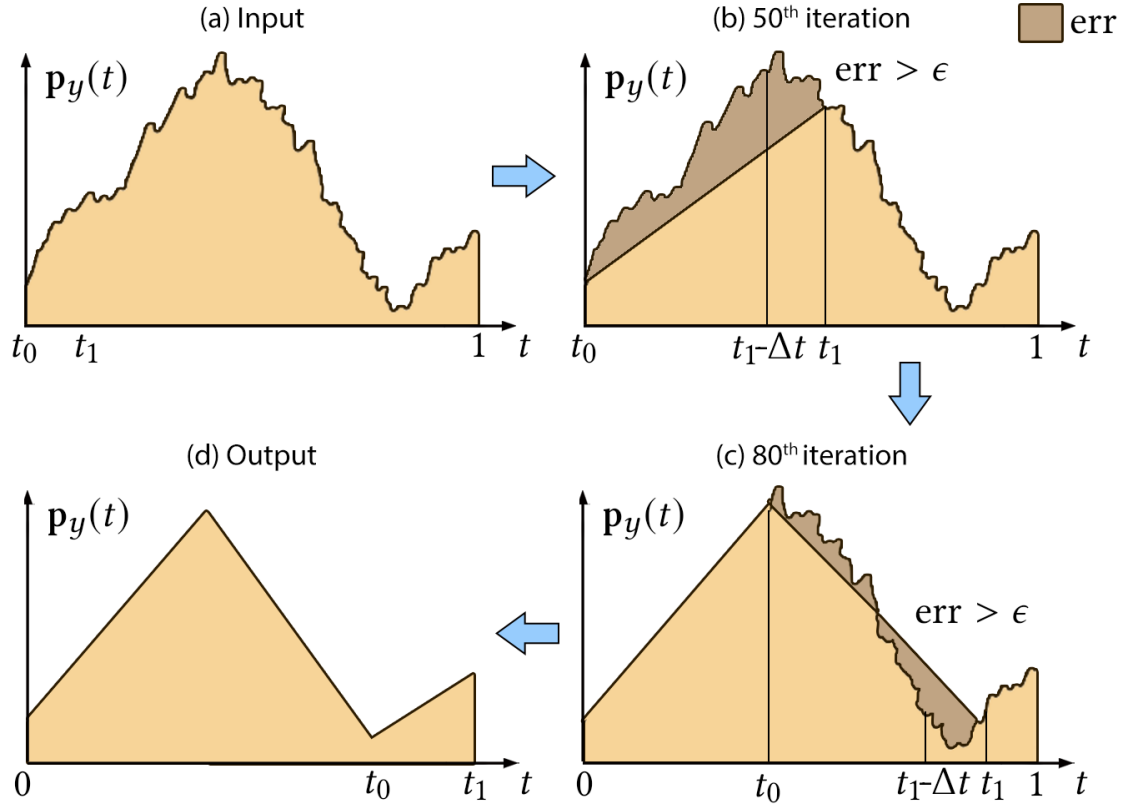


Figure 3.7: Attaining high coplanarity of local road surfaces through our post-processing method. (a) Input terrain that is bumpy. (b) At the 50th iteration, as the accumulated approximation error exceeds the threshold ϵ , the terrain between t_0 and $t_1 - \Delta t$ is approximated by a linear segment. t_0 is then reset to $t_1 - \Delta t$ to start searching for the next linear segment. (c) At the 80th iteration, as the accumulated error exceeds the threshold ϵ again, the next linear segment is formed. (d) The output is an approximation of the input terrain by a series of linear segments.

- **Shift:** The first and last control points move forward or backward by a random amount so that the curve shifts like a snake.
- **Reverse:** The first and last points of the curve swap with each other, reversing the travel direction of the path.
- **Shrink:** The first and last control points move towards each other to shrink the curve.
- **Expand:** The first and last control points move away from each other to elongate the curve.

- **Translate:** All control points translate by the same amount along a random direction so that the curve translates.
- **Rotate:** The whole curve is rotated with a random angle.
- **Rotate Sub-Path:** One side of the curve is rotated with a random angle while the other side stays.

To constrain the solution to be valid, we drop a move if it results in a 2D Hermite curve that is out of the boundary of the path grid or is self-colliding. To regularize the resulting curve to be smooth, a move is also dropped if the resulting curve makes a sharp corner, i.e., there exists control points (x_{k-1}, y_{k-1}) , (x_k, y_k) , and (x_{k+1}, y_{k+1}) such that $[x_k - x_{k-1}, y_k - y_{k-1}] \cdot [x_{k+1} - x_k, y_{k+1} - y_k] > 0$.

We use the Metropolis Hastings algorithm [103] with a simulated annealing [59] state-searching step to optimize the path. At each iteration, a new path \mathcal{P}' is proposed by applying a randomly-selected move on the current path \mathcal{P} as described above. The new path \mathcal{P}' is accepted with a probability defined based on the Metropolis criterion:

$$Pr(\mathcal{P}'|\mathcal{P}) = \min\left(1, \frac{\gamma(\mathcal{P}')}{\gamma(\mathcal{P})}\right), \quad (3.8)$$

where $\gamma(\mathcal{P})$ is a Boltzmann-like function comprising the total cost:

$$\gamma(\mathcal{P}) = \exp\left(-\frac{1}{t}C_{\text{total}}(\mathcal{P})\right), \quad (3.9)$$

and t is the temperature parameter of simulated annealing, which drops from an initial high value (we set it to 1 empirically) to a low value near zero over the iterations. Essentially, the initial high temperature allows the optimizer to explore the solution space more aggressively with a high acceptance probability. As the temperature decreases over iterations, the optimizer becomes more greedy and it is less likely to accept worse paths than before. The

Algorithm 1 Turning Points Extraction

```
1: function LINEARAPPROXIMATIONERROR( $\mathbf{p}_y(t), t_0, t_1$ )
2:   totalError      =  $\int_{t_0}^{t_1} \left| \mathbf{p}_y(t) - \left( \mathbf{p}_y(t_0) + (t - t_0) \frac{\mathbf{p}_y(t_1) - \mathbf{p}_y(t_0)}{t_1 - t_0} \right) \right| dt$    return
   totalError/( $t_1 - t_0$ )
3: procedure LINEAR APPROXIMATION( $\mathbf{p}_y(t), \epsilon$ )                                 $\triangleright \epsilon$ : max error
4:    $t_0 \leftarrow 0$                                                              $\triangleright 0$ : time stamp of start point
5:    $t_1 \leftarrow \Delta t$                                                      $\triangleright \Delta t$ : a small time step
6:    $T \leftarrow \{0\}$                                                           $\triangleright T$ : a set of time stamps of turning points
7:   while  $t_1 < 1$  do
8:     err  $\leftarrow$  LINEARAPPROXIMATIONERROR( $\mathbf{p}_y(t), t_0, t_1$ )
9:     while  $t_1 < 1 \wedge \text{err} < \epsilon$  do
10:       $t_1 \leftarrow t_1 + \Delta t$                                             $\triangleright t_0$  stays,  $t_1$  moves forward
11:      err  $\leftarrow$  LINEARAPPROXIMATIONERROR( $\mathbf{p}_y(t), t_0, t_1$ )
12:       $t_0 \leftarrow t_1 - \Delta t, T \leftarrow T \cup \{t_0\}$                 $\triangleright$  add new turning point
   return  $T \cup \{1\}$                                                           $\triangleright$  add time stamp of end point
```

optimization terminates if the absolute change in $C_{\text{total}}(\mathcal{P})$ is smaller than 3% over the past 50 iterations.

Parameter Settings.

By default, we set the weights as $w_L = 0.5$, $w_W = 0.3$, and $w_D = 0.2$. The targets ρ_L , ρ_W , and ρ_D are set according to the specific goals of the experiments.

Note that we attempted solving the optimization using CMA-ES similar to some existing works [104, 105]. However, we found that due to the complex optimization landscape and the curse of dimensionality (there were about 30 2D control points), the optimization was sensitive to the initialization of the first population in CMA-ES and was also memory-intensive. The optimizer tended to get trapped at a poor local optimum if the initialization was poor. Therefore, we discretize the solution space to reduce the problem complexity and apply simulated annealing with flexible moves to extensively explore the solution space. A path is optimized in a few minutes (5,000 iterations) typically.

3.4.4 Road Synthesis

With a generated path \mathcal{P} , our approach generates a road along this path on the terrain. The user bikes on this road in virtual reality. To make this road visually appealing and to deliver a natural and comfortable biking experience, we incorporate design concepts, namely, coplanarity and horizontality, from road engineering [106, 107], in constructing this road. The terrain is also slightly refined in forming this road.

Coplanarity of Local Road Surfaces.

Figure 3.7 shows the effects of maintaining the coplanarity of local road surfaces. While path \mathcal{P} is generated on the terrain whose surface could be bumpy, the road built on top of the path should be locally linear to prevent the user from biking up and down too frequently, which may result in dizziness. Such local linearity also suits the VR exercise bike hardware as it prevents the bike’s force feedback from changing too frequently, which may wear out the motor otherwise.

The key idea for achieving high coplanarity of local road surfaces is to approximate the path with a series of linear segments, each of which is associated with a coplanar road surface. To this end, we devise a post-processing method to refine the elevation of the original path. A new path \mathcal{P}' is formed which approximates the original path \mathcal{P} by a number of linear segments. More specifically, the new path’s elevation $\mathbf{p}'_y(t)$ is an approximation of the original path’s elevation $\mathbf{p}_y(t)$. The new path resembles the original path by keeping the key features of the original path such as its major turning points. The road is built on top of the new path \mathcal{P}' .

The post-processing method consists of two major steps: (1) extracting the major turning points from the original path following Algorithm 1; (2) and linearly interpolating the extracted turning points to form a new path \mathcal{P}' to approximate the original path \mathcal{P} . Figure 3.7 illustrates this process.

Note that while Section 3.4.2 uses the original path \mathcal{P} for deriving the costs, we use the new path \mathcal{P}' instead of the original path \mathcal{P} to evaluate the path length cost in (3.2) and

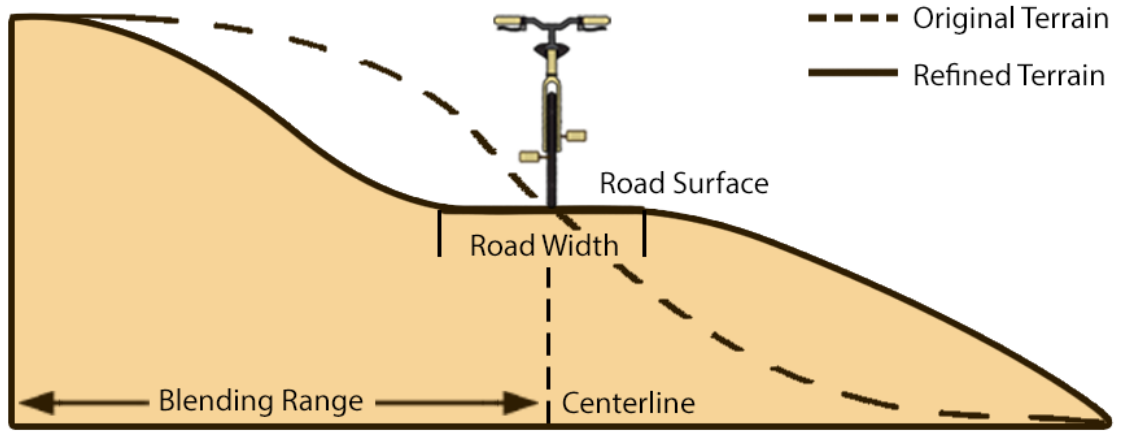


Figure 3.8: A terrain is refined to create a horizontal road surface.

the total work cost in (3.4). That is, we evaluate $C_L(\mathcal{P}')$ and $C_W(\mathcal{P}')$ instead of $C_L(\mathcal{P})$ and $C_W(\mathcal{P})$ in optimizing the paths for our experiments. This is because we had our user bike along the new path \mathcal{P}' rather than the original path \mathcal{P} in our experiments to avoid the high bumpiness of the input terrain. Using \mathcal{P}' for evaluating these costs is then more accurate. If the designer would rather generate a road closely following the bumpy terrain and did not apply the post-processing method for linear approximation, for example, for simulating a vigorous mountain biking experience, the original path \mathcal{P} could have been used for evaluating the costs.

Horizontality of Road Cross Slope.

As shown in Figure 3.9, a tilted road would cause a bike to slide or even flip on its side due to gravity. The cross slope of a road refers to the road's slope taken across the road's centerline. It is important to keep the cross slope of a road horizontal. Inspired by the geometric design of roads, we incorporate such a consideration in creating a realistic road along the path for delivering a natural biking experience.

To create a road with a horizontal cross slope, we slightly refine the terrain along the road by a blending operation (Figure 3.8). More specifically, we design an α -map blending function for refining the input terrain's height map $H(x, z)$ so that the terrain is flat along the cross slope of the road; and the sides (within a certain blending range) of the road blend smoothly into the natural landscape.

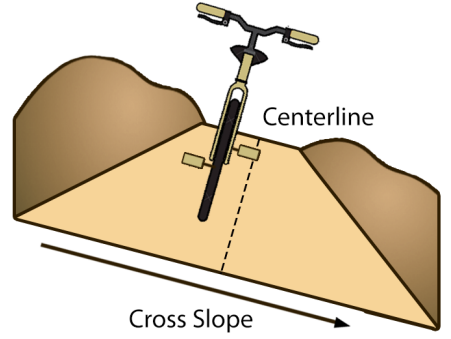


Figure 3.9: A tilted road.

We make the road's cross slope horizontal by setting the terrain near the center of the road to have the same height as the elevation of the path \mathcal{P}' . As the elevation of path \mathcal{P}' is $\mathbf{p}'_y(t)$, the new height map $H'(x, z)$ of the terrain after blending is:

$$H'(x, z) = \alpha(x, z)\mathbf{p}'_y(t^*(x, z)) + (1 - \alpha(x, z))H(x, z), \quad (3.10)$$

where $t^*(x, z) = \operatorname{argmin}_t |\mathbf{r}(t) - (x, z)|$ corresponds to the point $\mathbf{r}(t)$ on the 2D Hermite curve \mathcal{R} that is closest to the point (x, z) on the xz -plane. $d(x, z) = \min(|\mathbf{r}(t) - (x, z)|)$ denotes the minimal distance from pixel (x, z) to the 2D Hermite curve \mathcal{R} . The α -map blending function $\alpha(x, z)$ is defined as:

$$\alpha(x, z) = \begin{cases} 1 & d(x, z) < w \\ 0 & d(x, z) > \kappa \\ \frac{1}{2} + \frac{1}{2} \cos\left(\frac{d(x, z) - w}{\kappa - w} \pi\right) & w \leq d(x, z) \leq \kappa, \end{cases} \quad (3.11)$$

where w is the road's half width and κ denotes a blending range. The blending decays from 1 to 0 as $d(x, z)$ increases from w to κ .

3.5 Experiments

3.5.1 Implementation

We implemented our approach using an Alienware machine equipped with an Intel Core i7-9700 CPU, an NVIDIA GeForce RTX 2070 graphics card, and 32GM of RAM. The implementation was done in Python. A path is generally optimized in about 1,000 to 5,000 iterations, depending on input terrain and the specified targets which affect the optimization difficulty. Optimizing a path generally takes about 1 to 5 minutes on our machine.

Custom-built Bike. Figure 3.11 shows the custom-built bike used in our user study. It was built based on a Schwinn Evolution SR Indoor Cycle Bike. We built an Arduino MEGA2560 R3 with a motor to control the resistance and sensors to monitor the cadence. The resistance level of the original bike could be adjusted by turning the resistance control knob. We replaced the manufacturer’s knob with our own stepper motor that was controlled by the Arduino. Then

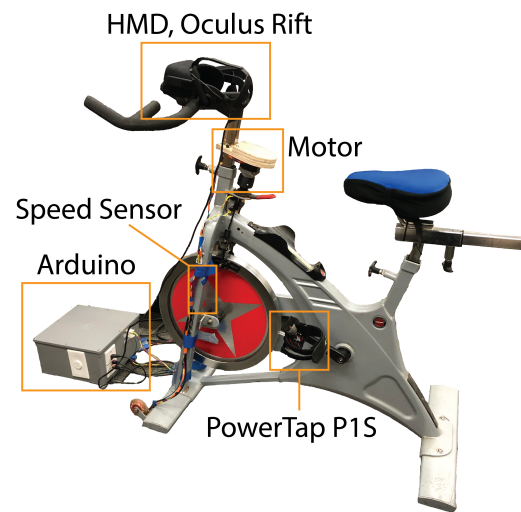


Figure 3.11: Our custom-built bike.

the Arduino could adjust the resistance according to the elevation angle of the generated path to give the desired feedback force. As the force feedback is tuned by a resistance knob controlled by a stepper motor, the motor speed determines the speed of change of force feedback. We tuned our motor to avoid sudden changes. Using more discrete levels used can also smoothen the transition of force feedback between different levels.

A reed switch was mounted on the bike’s frame near the wheel to measure the speed. Two magnets were attached to the wheel, triggering the reed switch when they passed. This allowed speed measurement by the Arduino with a high degree of accuracy. The Arduino sent the speed readings to the computer through serial ports so that our program could

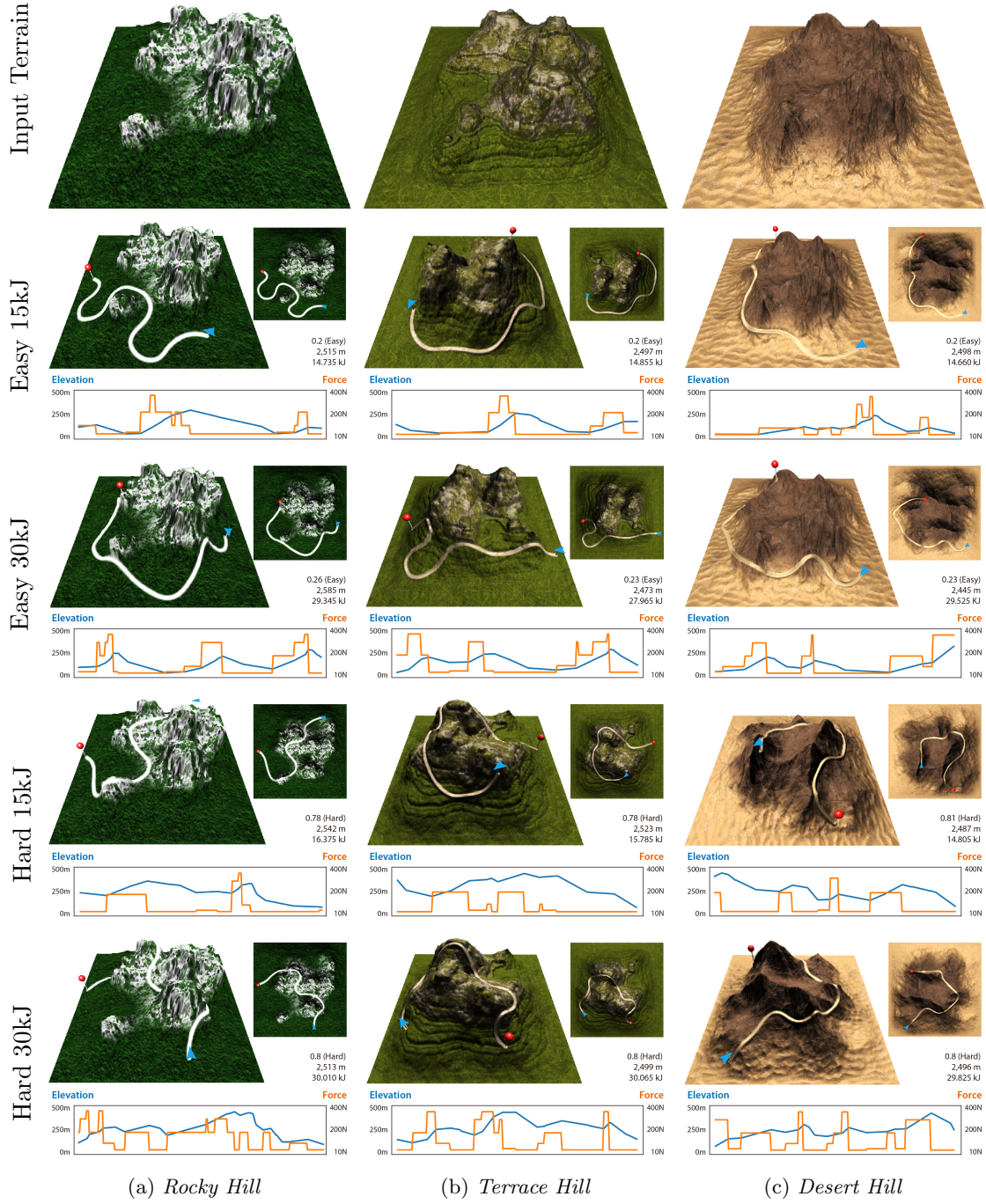


Figure 3.10: Paths generated on different terrains with different exertion goals. In each row, the path difficulty target (e.g., easy) and total work target (e.g., 15kJ) are shown on the left. The path length target is 2,500m for all. The path difficulty, path length, total work, as well as the elevation and feedback force plots of each generated path are shown. The blue arrow and the red pin refer to the start and end.

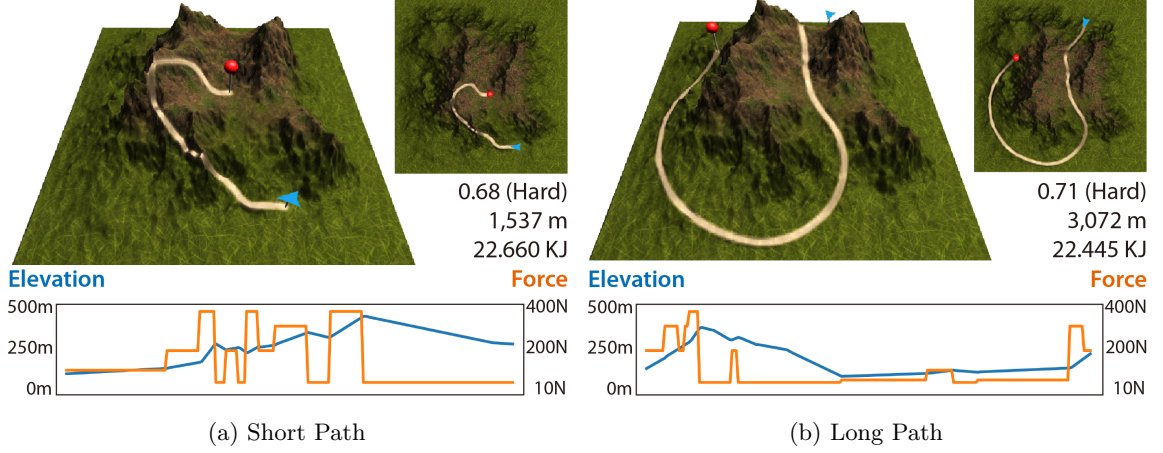


Figure 3.12: A long path and a short path generated using long and short path length targets on the same input terrain. Both paths used the same total work target $\rho_W = 22.5 \text{ kJ}$. Our approach generates (a) a short uphill path which is more intensive; and (b) a long downhill path which is less intensive.

move the user's bike in virtual reality according to the measured speed.

Since we wanted to validate the effectiveness of our approach, we replaced the original pedals with PowerTap P1S power meter pedals. Per the manufacturer's instructions, the left pedal was used for measuring the power output of a user as he was biking. The original bike allowed a total of 10 levels of feedback forces. However, we dropped the first three levels as the forces were very small and almost identical. We also dropped the last level as it was very heavy to ride. Figure 3.4 shows the magnitudes of the levels of feedback forces used.

Terrain Generation.

To conduct our experiments, we generated different terrains including a *Rocky Hill*, a *Terrace Hill*, and a *Desert Hill*. We used a terrain generator called World Creator, which can generate, design, blend, mix, paint sculpt, erode and simulate realistic terrains, to generate the height maps of the terrains. As one feature of the World Creator, different types of terrains can be generated through different filters that carry specific geological properties.

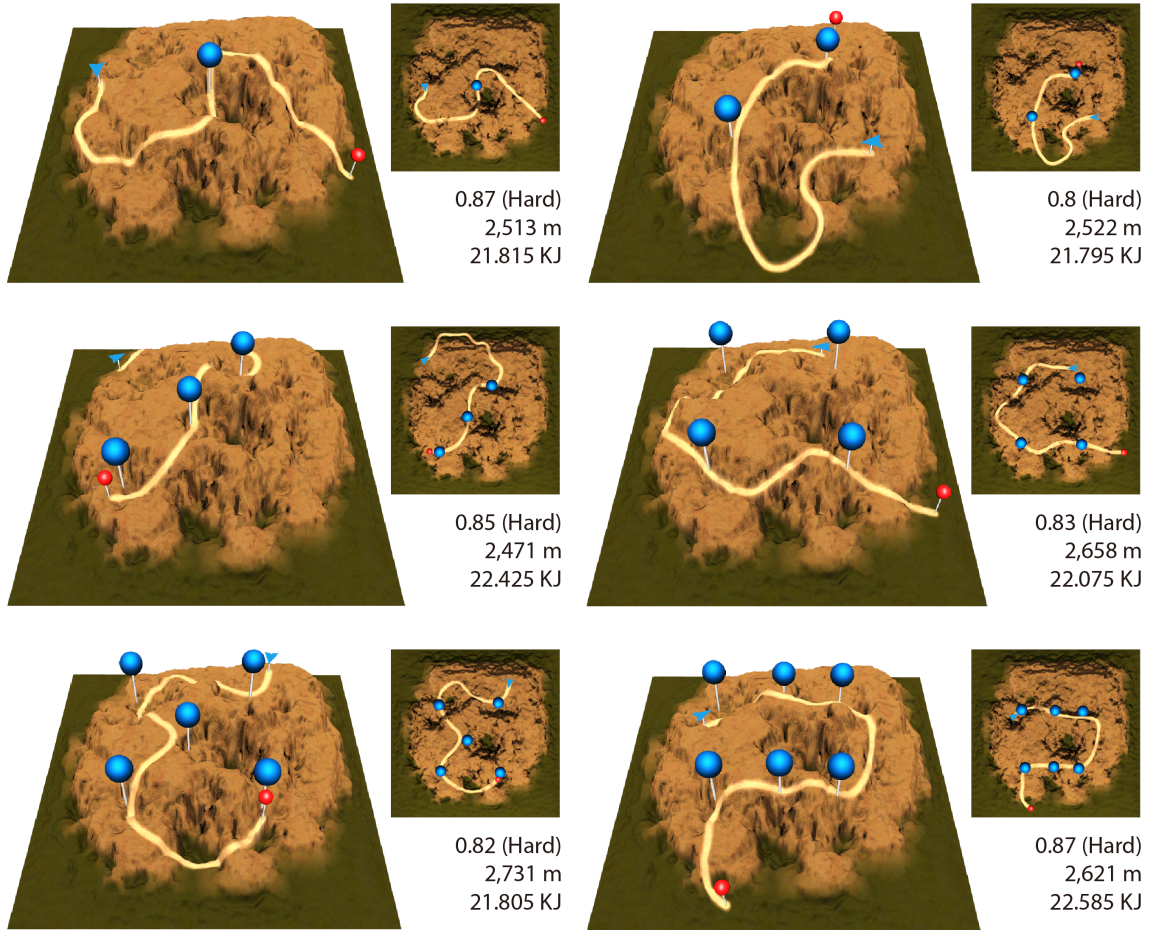


Figure 3.13: Paths generated with specified landmarks (in blue). All paths were generated with the same path length target $\rho_L = 2,500\text{m}$ and total work target $\rho_L = 22.5\text{kJ}$. The generated paths pass the landmarks closely.

For example, the *Rocky Hill* was generated using the canyon filter, rocky-hills filter, smooth slope filter, and the zero-edge filter.

For realistic rendering, we simulated the appearance of the terrain using three different levels of textures, namely, the road texture, the plain texture, and the mountain texture. We used an alpha blending function similar to that described in Section 3.4.4 to blend the road texture smoothly into the texture of the natural landscape (i.e., mountain or plain) near the sides of the road.

The path \mathcal{P}' generated in Section 3.4.4 that approximates the original path \mathcal{P} by a number of linear segments may not be smooth, resulting in corners between linear segments.

While we use the elevation angles from path \mathcal{P}' to control the discrete feedback force of the bike following Section 3.4.2, for visual appeal only, we smooth the corners by fitting small quadratic Bézier curves at the corners.

3.5.2 Different Terrains and Exertion Goals

We experimented our approach with optimizing paths on three different terrains. Each generation used the same path length target $\rho_L = 2,500\text{m}$. The paths were generated using four combinations of exertion goals:

1. Easy and Low Work ($\rho_D = 0.2$, $\rho_W = 15\text{kJ}$);
2. Easy and High Work ($\rho_D = 0.2$, $\rho_W = 30\text{kJ}$);
3. Hard and Low Work ($\rho_D = 0.8$, $\rho_W = 15\text{kJ}$);
4. Hard and High Work ($\rho_D = 0.8$, $\rho_W = 30\text{kJ}$);

Figure B.4 shows the terrains and the generated paths. All generated paths have a length which is close to the path length target of 2,500m. As the plots below each sub-figure show, the feedback force pattern changes with the path elevation correspondingly. The paths generated with larger total work targets tend to have larger average forces than the ones generated with smaller targets. On the other hand, given the same total work target, paths generated with different path difficulties tend to have different distributions on the same terrain: Easy paths tend to lie on the plain area where the elevations vary in smaller scales; hard paths tend to lie on the mountainsides where the elevations vary in larger scales. We conducted user studies based on some of these generated paths. We discuss the results in Section 3.6.

3.5.3 Path Length

Figure 3.12 shows a short path and a long path generated using a short and long path length target respectively. The input terrain is called *Corner Hill*. The short path and long

path were generated with a target of $\rho_L = 1,500\text{m}$ and $\rho_L = 3,000\text{m}$ respectively. The same total work target of $\rho_W = 22.5\text{kJ}$ was used for both paths. As the results show, to achieve the same amount of total work, the short path is more intensive in terms of force requirement as it goes uphill while the long path is less intensive as it goes downhill mostly.

3.5.4 Landmarks

Figure 3.13 shows paths generated by specifying landmarks on the terrain that the paths should pass. The input terrain is called *Sandy Hill*. The landmarks specified by the designer follow the dot patterns on the six faces of a dice. The paths were generated with the same path length target $\rho_L = 2,500\text{m}$ and the same total work target $\rho_L = 22.5\text{kJ}$.

We extend the formulation by adding a landmark cost that encourages a generated path to pass the landmarks. Denote the positions of the landmarks on the 3D terrain as $\{(x_i, y_i, z_i)\}$. The landmark cost $C_K(\mathbf{p})$ is defined as:

$$C_K(\mathbf{p}) = 1 - \exp\left(-\left(\frac{1}{\sigma_K \omega K} \sum_{i=1}^K d(x_i, z_i)\right)^2\right), \quad (3.12)$$

where $d(x_i, z_i)$ denotes the minimal distance, defined in (3.11), between landmark i and the 2D Hermite curve \mathcal{R} on the xz -plane. K is the number of landmarks used. ω is the width of the terrain. $\sigma_K = 0.25$ is set empirically. As Figure 3.12 shows, the generated paths pass the landmarks either exactly or very closely.

3.5.5 Forbidden Zones

Figure 3.14 shows paths generated with the avoidance of forbidden zones referring to sea water. The input terrain is called *Island Hill*. The sea water covers the land according to the sea level. The paths were generated with the same path length target $\rho_L = 2,500\text{m}$ and the same total work target $\rho_L = 22.5\text{kJ}$. We enabled such a consideration by adding a

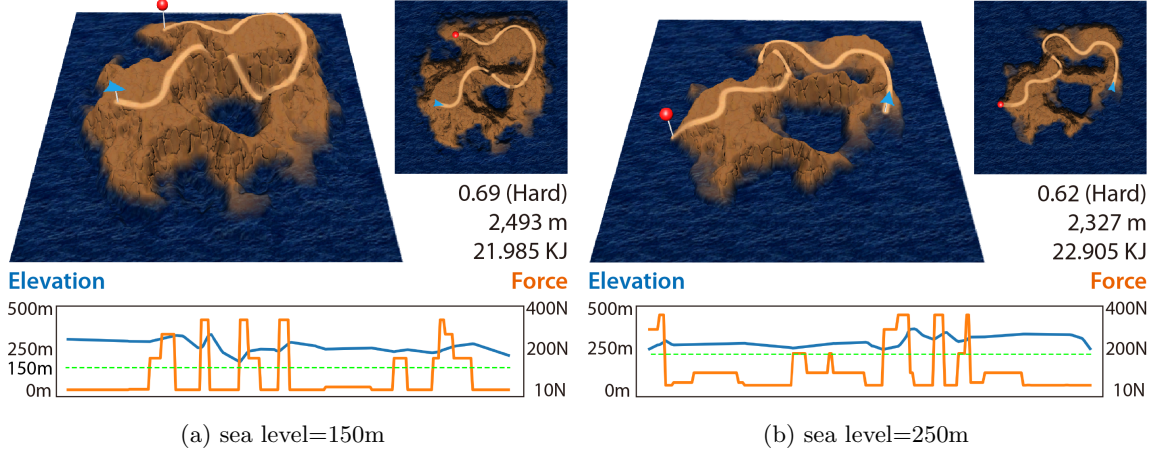


Figure 3.14: Paths generated with the avoidance of forbidden zones which refer to sea water in this example. (a) A path generated as the sea level is 150. (b) A path generated as the sea level rises to 250m.

forbidden zone cost $C_F(\mathbf{p})$ defined as:

$$C_F(\mathbf{p}) = 1 - \exp\left(-\left(\frac{1}{\sigma_F} \frac{\int_{\mathcal{P}} \Gamma(\mathbf{p}) d\mathbf{p}}{\int_{\mathcal{P}} d\mathbf{p}}\right)^2\right), \quad (3.13)$$

where $\Gamma(\mathbf{p}) = 1$ if path \mathbf{p} is below the sea level and $\Gamma(\mathbf{p}) = 0$ otherwise. $\sigma_F = 0.25$ is set empirically. The cost penalizes if path \mathbf{p} enters the forbidden zones. As Figure 3.14 shows, the generated paths avoid getting into the sea water. Alternatively, the designer could manually specify obstacles on a terrain to avoid similarly. Figure 3.15 shows an example using the same path length and total work targets.

3.6 User Study

We conducted user study experiments to test the enjoyment and exertion induced on users by our generated paths. The experiments were done with a head-mounted display (HMD) VR headset and a custom-built exercise bike, which were connected to an Alienware laptop showing the generated paths in a game-like setting. The laptop was equipped with an Intel

Core i7-8750H CPU, an NVIDIA GeForce GTX 1070 graphics card, and 16GB of memory. The VR headset was the Oculus Rift, which used an OLED screen with $2,160 \times 1,200$ resolution at 90Hz. Our custom-built exercise bike allowed our program to change its feedback force and monitor its cadence and power output while a user was biking.

3.6.1 Enjoyment Test

To test the enjoyment of biking experiences using our VR setup, we recruited 23 participants to bike our generated paths (from Section 3.5.2) under three different exercise modes (VR mode, TV mode and Gym mode) as shown in Figure 3.16. We examined descriptive statistics on our participants’ performance using a Friedman Test to detect if there was an overall difference across the three different modes. We also conducted Wilcoxon signed-rank tests to pinpoint where the difference was located in each pair of modes. (e.g., VR-TV pair). From the statistical analysis, we concluded that the participants generally preferred exercising under the VR mode. For more details of the analysis, please refer to our supplementary material.

3.6.2 Exertion Test

To verify the exertion effectiveness of the generated paths, we used the Hard 15kJ and Hard 30kJ paths generated for the *Rocky Hill* as shown in Figure B.4 for an exertion test. We call these paths Hard15 and Hard30 for brevity. We wanted to validate if the participants achieved the total work targets (e.g., 15kJ and 30kJ) specified for optimizing these paths.

Participants. We recruited 10 participants, who were university students and staff, to bike the Hard15 and Hard30 paths. They were 26 years old on average and their average body mass index was 21kgm^{-2} . Refer to the supplementary material for more details.

Procedure. Our evaluation procedure was IRB-approved. Each participant was briefed about the game control and given a warm-up session to get familiar with the game. Then we asked the participant to bike the two paths given in a randomized order. A path was

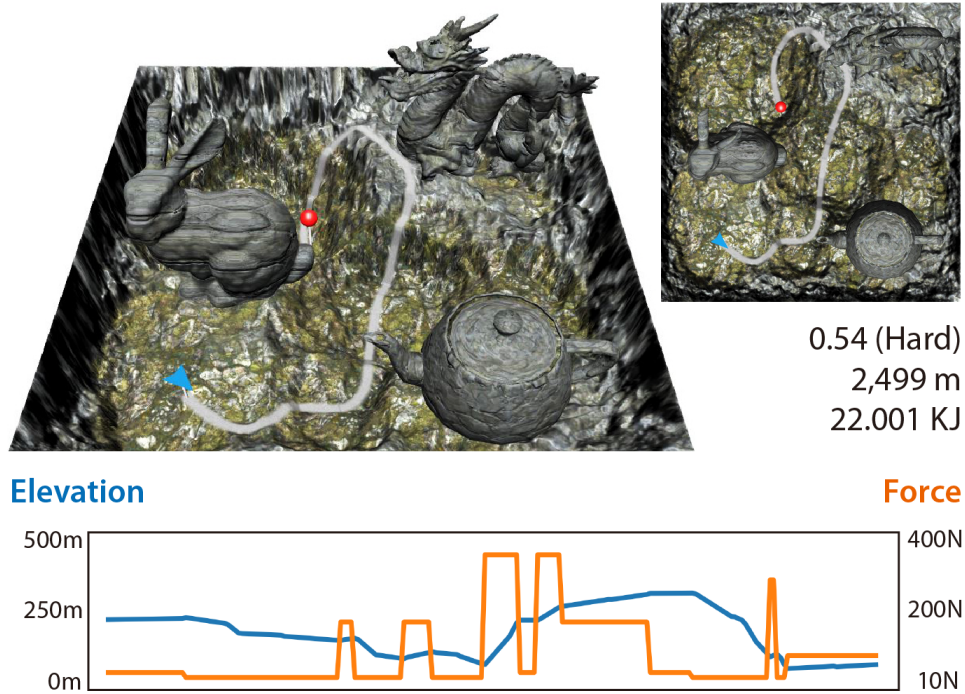


Figure 3.15: Paths generated with the avoidance of manually specified obstacles which are the bunny, dragon, and teapot in this case.

completed when the participant reached the end point. There was a 5-minute break between biking the two paths.

Measurement. We used the PowerTap P1S power meter installed on our bike to record the power output and duration during the experiments for analysis. Before the evaluation, we calibrated the power meter. The captured data included duration in seconds, energy burned in kilojoules, average power in watts, and average cadence in round-per-minutes. The power meter was comfortable to use and did not require the participant to wear any other sensor.

Results. Figure 3.17 compared the average energy

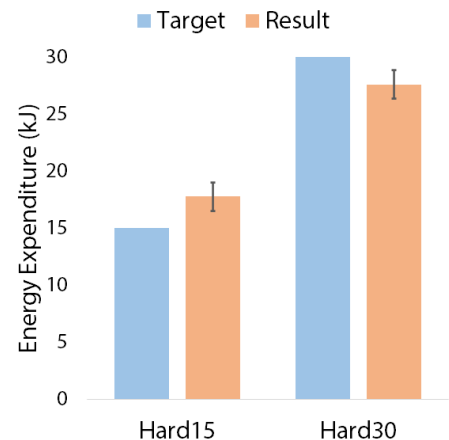


Figure 3.17: Average energy expenditure in biking the Hard15 and Hard30 paths versus the targets. A significant difference was found with $p < 0.001$.

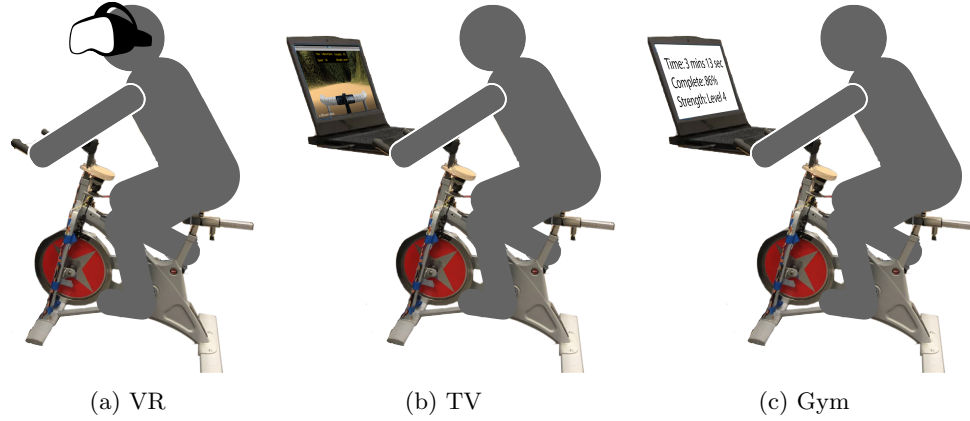
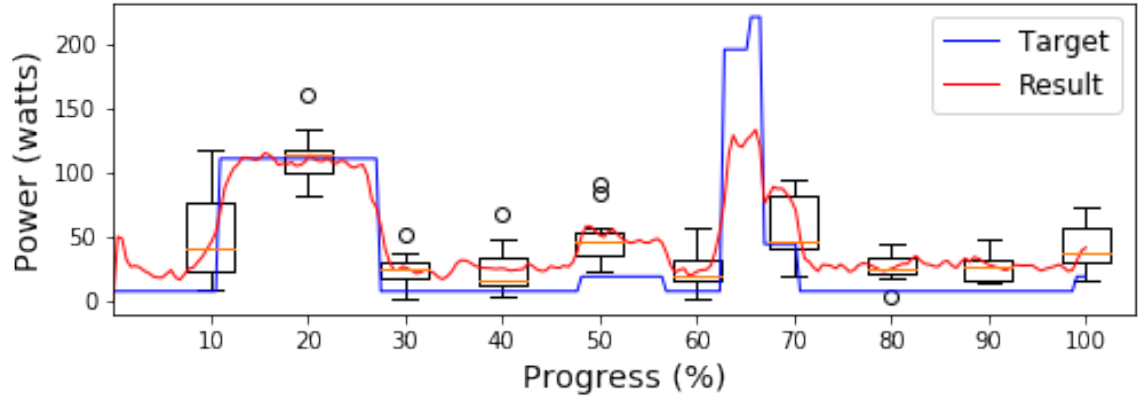


Figure 3.16: Three modes, delivered in random order, were used for the enjoyment test. (a) VR mode based on our VR bike system; (b) TV mode where the user biked while watching the game on a screen; (c) Gym mode where only biking information was displayed.

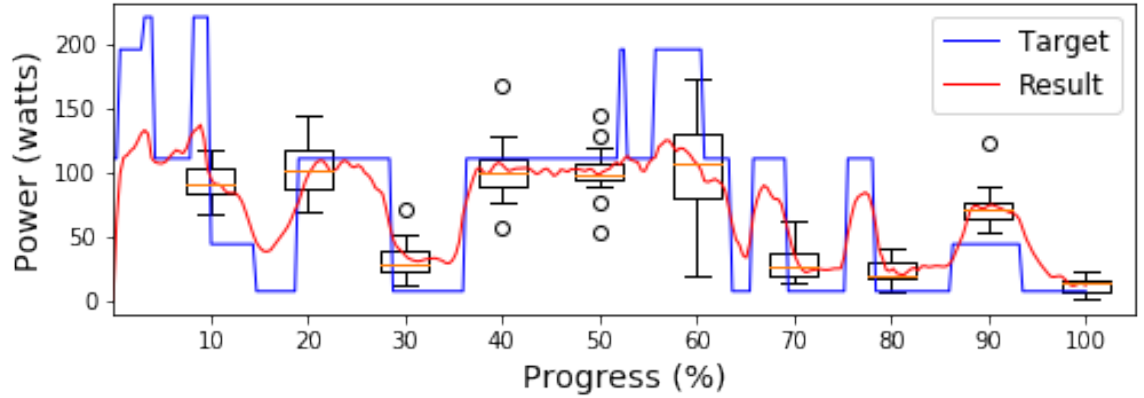
expenditure with the total work targets specified for optimizing the paths. We observed that overall the results were close to the targets. On average, the energy expenditure for the Hard15 path was 17.8kJ and that for the Hard30 path was 27.6kJ. Table 3.1 shows the descriptive statistics. We provided a possible explanation for the slight deviations from the targets in our supplementary material.

We examined descriptive statistics on our participants' energy expenditure using a paired t-test to detect whether there are differences in biking the Hard15 and Hard30 paths. A significant difference ($p < 0.01$) was found.

To observe the participants' energy expenditure during the experiment, we compared their power output results during the entire gameplay with the expected power output targets of the generated paths. Note that we convert the expected feedback forces of our generated paths into the expected power output targets for this comparison because the power meter measured power, not forces. We describe the conversion in our supplementary material. As Figure 3.18 shows, the participants' power output results follow the targets closely. Overall, we conclude that the participants' energy expenditure matched with the total work targets specified for optimizing the paths reasonably well, hence the generated



(a) Hard15 results.



(b) Hard30 results.

Figure 3.18: Boxplots of the participants’ power output in biking the Hard15 and Hard30 paths. The red line show the average result of the 10 participants. The blue line shows the expected power output target.

paths help the participants achieve the exertion goals.

User Feedback. We asked our participants to answer some open response questions at the end of our evaluation. Most participants felt that our VR bike was fun to play since the biking experience was realistic and they enjoyed biking up and down. Some of them preferred using our VR bike setup as a replacement for a typical exercise bike. One complained that the resolution of our game was not high enough. One wished that the hardware could be improved with a brake. Our supplementary material contains more user comments.

Table 3.1: Descriptive statistics for the Hard15 and Hard30 paths. The average completion duration, power output and energy expenditure of the participants are shown.

| Path | Average Power(watt) | Time(second) | Energy(kJ) |
|-------------|----------------------------|---------------------|-------------------|
| Hard15 | 110.4 ± 16.5 | 162.7 ± 23.3 | 17.8 ± 2.4 |
| Hard30 | 157.0 ± 23.6 | 178.1 ± 17.3 | 27.6 ± 1.6 |

3.7 Summary

We proposed a novel problem of generating paths over a terrain with desirable force feedback properties. We devised an optimization-based approach for generating such exertion-aware paths, which can be employed for virtual training experiences such as exergaming. By applying the generated paths to control the force feedback of an exercise bike, the immersiveness and realism of the virtual reality experience are enhanced as the user’s haptic experience matches with his visual perception. We verified our approach for generating exertion-aware paths for a variety of scenarios and also through a user study consisting of an enjoyment test and an exertion test.

3.7.1 Limitations and Future Work

We discuss some limitations and possible extensions of our approach. We measured the energy expenditure in joules instead of calories because the amount of calories burned could vary among people; different people could burn different amounts of calories when doing the same exercise (biking the same path), influenced by their gender, age, height, weight, and other factors. In future work, we could obtain basic body information of a user and employ that to generate a more personalized level that uses calories burned as an optimization objective to deliver more personalized training.

For our proof-of-concept experiments, we built our bike based on a common indoor cycling bike that supports 10 levels of feedback force in its original setting. The feedback force levels we used covered elevation angles up to 40° that corresponds to a 84% gradient, which was comprehensive in terms of angle coverage. However, a high-end exercise bike

could offer more feedback force levels (e.g., 20 levels) which could provide more realistic and high-resolution haptic experience. Our approach could be easily extended to support more feedback force levels. We note that some high-end commercial exercise bikes are equipped with a live coach feature, allowing the user to receive personal coaching. These services usually required paid membership and could be expensive. Our approach provides an alternative, low-cost solution to generate and deliver exercising programs with desired exertion goals.

As shown in Section 3.4.2, our classifier achieved a 82% accuracy in classifying perceived difficulties of paths based on path features. Evaluating perceived difficulties in a user study was difficult because there are a number of factors which could affect the perceived difficulty of a biking experience but could be difficult to simulate. For example, we could not simulate some real-world factors (e.g., road textures, windiness, humidity) which are not related to paths but could have influenced the perceived difficulty ratings in our dataset. Therefore, in our approach, we consider the difficulty mainly based on the elevation features of a path. Moreover, we assume that the terrain’s surface material is the same throughout the experience, and hence the friction coefficient is constant. In future work, we could include a variety of surface materials (e.g., mud, sand, concrete) carrying different physical properties such as friction properties for a terrain. We could also consider weather effects on the friction properties of the road. For example, a path on a rainy day should be more slippery than that on a sunny day, which could be simulated by decreasing the friction coefficient.

As far as the scope of our user study is concerned, the paths generated for our user evaluation are not compatible with a regular biking training session which is typically at least 30-minute long. For a real training, we could generate the desired path by using a longer path length target or a larger total work target in our optimization.

Moreover, we assume that the input terrain consists of a wide range of elevation which allows our optimizer to find a path that satisfies the exertion goals. If the terrain does not show desired variations in elevation, finding a satisfactory path may not be feasible. Figure 3.19 shows two failure cases, where our approach failed to find a hard path on the

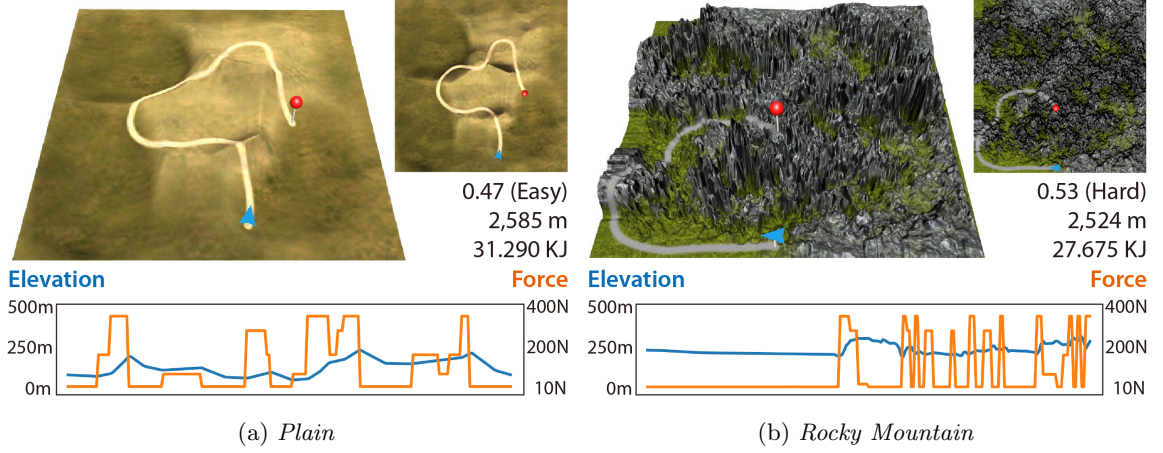


Figure 3.19: Failure cases. Our approach failed to find (a) a hard path on a plain and (b) an easy path on a rocky mountain.

Plain terrain and an easy path on the *Rocky Mountain* terrain although the total work target of 30kJ and the path length target of 2,500m were roughly achieved. To address this problem, future extensions might analyze a terrain to suggest possible exertion goals in a pre-processing step. Another extension is to explore optimizing both the terrain and the path concurrently to generate a solution that satisfies the exertion goals.

To demonstrate the novelty of our approach for generating paths with desired force feedback properties, we customized a bike which allows users to achieve exertion goals in virtual reality training. Our approach could be extended for generating virtual training content for other exercise machines such as a treadmill or an elliptical while leveraging the enjoyment brought about by virtual reality.

It would be interesting to extend our bike with a fan whose speed changes according to the speed of the bike. For example, the fan speeds up when the bike is going downhill so that the user can feel the wind. Such haptic feedback may create an even more engaging, realistic, and immersive VR exertion experience.

Chapter 4: Synthesizing Construction Safety Training Scenarios for VR Training

4.1 Introduction

In the construction industry, the labor costs constitute 30-50% of total project costs[108]. Since construction companies typically have a 2-3% profit margin, managing labor costs in terms of safety and productivity are critical[109]. In fact, lack of skilled labor is the main reason for cost overruns[110], loss of productivity[111], more frequent safety incidents[112], schedule overruns[113], and decline in quality performance[114]. To improve their skills, construction workers must be trained; however, the traditional one-size-fits-all training approach has failed to fully prepare workers for current worksite challenges[115]. Therefore, developing personalized and cost-efficient training programs for construction workers will have a significant impact on the bottom line of construction companies. It becomes pertinent to find efficient approaches and safe training programs to speed up the construction industry's human supply.

On the other side, given the emerging technologies in the virtual reality industry, VR provides a practical platform to train people in a safe and time-saving manner through serious virtual training tasks. Lots of recent research indicates that virtual reality training can be a substitute for real-world tasks training to some extent[2,116–119]. Therefore, given this observation, it is possible to take advantage of virtual reality to achieve the training goals not easy to be achieved in reality. As less prior knowledge is required, VR training saves a considerable amount of time for the trainee to start practicing. Besides, virtual reality training is injury-free and less-stressful for the trainee to step forward easily and improve their skills efficiently.

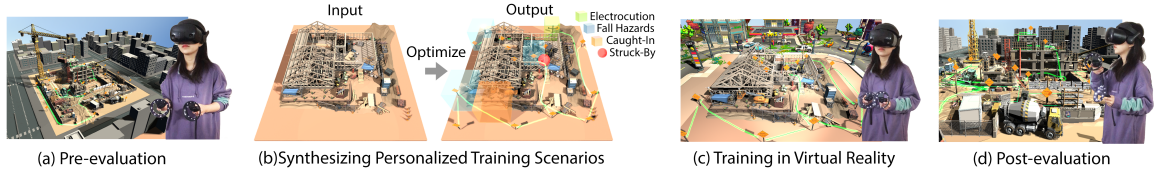


Figure 4.1: Overview of our approach. (a) We synthesize the tasks to pre-evaluate users’ background. (b) According to the users’ performance, we synthesize personalized training scenarios. (c) Users are trained through those synthesized training scenarios that emphasize their specific skills. (d) We post-evaluate the users’ performance through the same sequence of tasks in pre-evaluation to measure their improvement.

As motivated by the facts that most of those existing works that are successful in building up immersive virtual scenarios for simulating the virtual experience in a construction site[120–124] are lack of emphasizing the effectiveness of personalized construction safety training, we devised an optimization-based approach to automatically synthesize personalized training scenarios to improve the users’ skills in inspecting potential hazards within the construction sites in virtual reality. Given the specific training targets such as personalized preferences and estimated training time, we optimize the synthesized training scenarios through the simulated annealing approach. In order to prove the efficacy of our approach, we developed several virtual construction sites and conduct a series of user studies to analyze the users’ performances before the VR training and after. The major contributions of our work include:

- Building construction sites with different types of common hazards integrated and implementing a user-friendly interactive interface for construction safety training in virtual reality.
- Devising an optimization-based approach to synthesize personalized training scenarios on our developed construction sites, including the navigation routes and hazard inspection tasks, to achieve user-specified training targets.
- Validating the effectiveness of our synthesized training scenarios through user studies and statistical analysis.

4.2 Related Work

Virtual Construction Safety Training. As the traditional construction safety training industries share a common drawback of the lack of trainees’ engagement [125], more innovative approaches and benchmarks [126–128] are presented to improve the construction safety training approaches. To enable trainees to have control over their progress and achieve personal instructions in a safe environment, more attention is given to immersive virtual construction safety training [125, 129–133]. In recent years, as virtual reality technologies become widespread, Sacks et al. [120] indicate that virtual construction safety training is feasible and effective, especially in terms of workers’ learning, identifying, and inspecting the construction safety risks. Aati et al. [121] developed an immersive virtual platform for enhancing the training for work zone inspectors, which is capable of replicating the experience of an inspector driving through a work zone in a safer, cheaper, and quicker way. Through the simulation of VR-based safety training programs developed by Zhao et al. [134], users can effectively promote their abilities for hazard inspection. At the same time, augmented reality and augmented panoramas of reality technologies are applied to virtual construction safety training to achieve immersive training experiences [133]. Furthermore, Eiris et al. [122] developed and compared two immersive virtual training platforms for hazard inspection among which one is based on VR while another is based on the 360-degree panorama. Those works plot a trend that immersive virtual construction safety training is gaining popularity and attracting more researchers.

Personalized Virtual Training Synthesis. On the other side, given the observation that most of the existing construction safety VR training programs work through repeatedly rehearsing similar tasks pre-programmed in the VR applications [134], this loses the training’s effectiveness concerning the fact that real-world hazards occur randomly. Therefore, randomness becomes a factor considered in the system proposed by Xie et al. [135] to create many variations of the training scenarios. However, randomized hazard simulations make it non-trivial and time-consuming for the designers to manually generate personalized training

scenarios for VR training. As inspired by the recent works which successfully synthesized personalized training scenarios for users to emphasize specific skills such as improving driving habits[2] and improving the proficiency in manipulating a wheelchair[119], optimization techniques can be applied to substitute manual configurations and results in effective training effects such as user-specified exertion targets settings for exertion games[14, 27, 46]. Emerging optimization technologies such as optimization for navigation graph-based way-finding algorithms proposed by Huang et al. [7], optimization for personal preferences-driven functional workspace scene layout synthesis proposed by Liang et al. [8], and optimization for realistic furniture arrangement in indoor scenes proposed by Yu et al. [9] indicate that stochastic optimization-based approaches are effective to synthesize realistic virtual scenarios timely and accurately. Motivated by these works, we focus on devising an optimization-based approach to automatically synthesize the personalized construction safety training scenarios for VR training.

4.3 Overview

As shown in Figure 4.1, we build up the construction sites on the VR platform and synthesize the training scenario where the users are pre-evaluated. A training scenario includes a route for users to navigate through (Green curve) and a sequence of nearby hazards (colored regions) that need to be correctly identified by the users. Through optimizing the navigation route along with the sequence of nearby hazards in a fully automatic manner, our approach synthesizes several personalized training scenarios to improve the user’s specific skills which are poorly performed in pre-evaluation. During the VR training, users are put into the personalized training scenarios optimized through our approach such as the ones with more fall hazards, more caught-in/between hazards, and more electrical hazards to efficiently improve the trainee’s skills in inspecting a specific type of hazard. In the end, we validate the effectiveness of our synthesized training scenarios by comparing the users’ performances in post-evaluation with the ones in pre-evaluation and testing the improvement in users’

performances statistically.

4.4 Training Scenario Synthesis

Hazard Types. According to the official documentation from the United States Department of Labor[136], major construction hazards types include (1) Struck-by Hazards, (2) Fall Hazards, (3) Caught-In/Between Hazards, and (4) Electrical Hazards as shown in Figure 4.2. In our optimization approach, we classify the struck-by as dynamic hazards and other types of hazards are static hazards or potential hazards. Noted that this classification method only applies to general situations. More strictly, in some special situations, struck-by is not the only dynamic hazard. For example, sometimes caught-in/between is dynamic as well when it is related to moving mechanical parts, or electrocution when the contact of a boom vehicle with a power line can also have a dynamic nature. However, in our approach, in order to separate more clearly the "moving and obvious" hazard that needs the users to avoid and those "hidden or potential" hazards for users to identify, we tag those moving hazards as "dynamic" while those staying hazards as "static". Our approach can be generalized to more complex training scenarios with both dynamic and static features where dynamic hazards are simulated through animations while static hazards need to be identified correctly by the users during safety training. Here the common hazards for safety inspection training are mainly divided into four types:

- Struck-By Hazards. As dynamic hazards, struck-by hazards are stimulated through mechanical animations. During the VR training, struck-by hazards can directly run into the users when they are passing nearby.
- Fall Hazards. Refer to the hazard caused by the falling of people. Such as un-tied workers, no guardrail protection on high, ladder hazards, and tripping hazards.
- Caught-In/Between Hazards. Refer to the hazards that crush people by the objects from two sides, such as cave-in hazards.

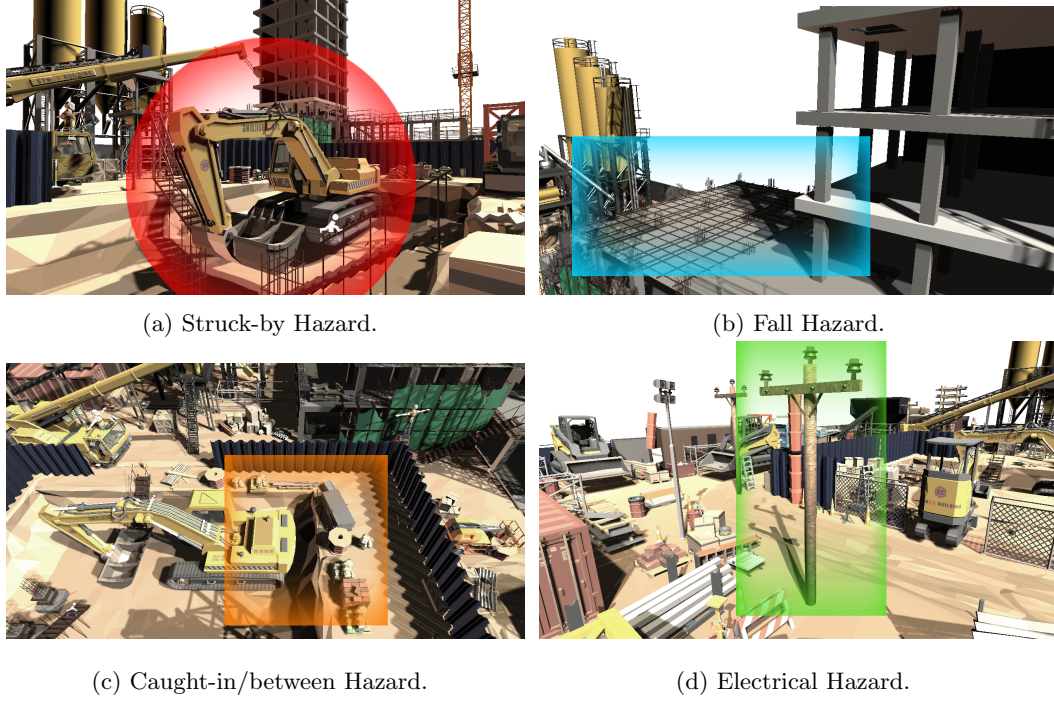


Figure 4.2: Examples of common hazards. (a) The struck-by hazard caused by the swing of excavator. (b) The fall hazard caused by missing guardrail protections. (c) The caught-in/between hazard caused by the objects which are too close to the cave’s edge. (d) The electrical hazard caused by electrical component such as light pole.

- **Electrical Hazards.** Refer to the hazards caused by electrical components, such as overhead power lines and no clearance between the power line and scaffold.

As shown in Figure 4.2, there are examples of common hazards highlighted in different colored regions: the red circle denotes a struck-by hazard, the blue, yellow, and green rectangles denote the fall, caught-in/between, and electrical hazard respectively.

4.4.1 Cost Functions

Total Cost. In our approach, a training scenario is synthesized as a route and several hazards along the route. A route starts at the entrance of a construction site and ends at a random position. Route R along with the hazards H are optimized given to the total cost function $C_{\text{total}}(R, H)$:

$$C_{\text{total}}(R, H) = w_D C_D(R, H) + w_S C_S(R, H) + w_T C_T(R, H), \quad (4.1)$$

where the dynamic hazard cost $C_D(R, H)$ encodes how much probability that the user will encounter dynamic hazards such as struck-by hazards H when navigating along the route R , the static hazard cost $C_S(R, H)$ encodes the users' preferred training targets with respect to their personal background of proficiency in identifying different types of static hazards such as fall, caught-in/between and electrical hazards, and the training time cost $C_T(R, H)$ encodes the target amount of training time. The w_D , w_S , and w_T represent the respective blending weights of these three cost terms.

Dynamic Hazard Cost. During our training process, dynamic hazards such as struck-by are simulated through mechanical animations to mimic the vital accidents that are likely to happen during the construction process in the real world. The spatial distribution of dynamic hazards affects the difficulty of safety inspection training. For example, if the dynamic hazards are distributed to the place near the navigating route, it is more challenging for the user to identify other nearby static hazards while avoiding being struck by the dynamic hazards. Otherwise, if the route is passing through the spaces without dynamic hazards, it is much easier for the user to focus on inspecting the static hazards, in this case, the training is less challenging. In order to consider this training factor, let $D(R, H)$ be a function proportional to the conditional probability of the occurrence of any dynamic hazard H that may happen along a route R , we evaluate the dynamic hazard cost function $C_D(R, H)$ by measuring the difference between that estimated probability $D(R, H)$ and users' target setting ρ_D as:

$$C_D(R, H) = 1 - \exp\left(-\left(\frac{D(R, H) - \rho_D}{\sigma_D}\right)^2\right), \quad (4.2)$$

where we empirically set $\sigma_D = 0.25$.

Hazard Probability. As the simulations of dynamic hazardous accidents are random and unpredictable, we introduce the conditional probability to evaluate the probability of the occurrence of hazardous accidents. Given any point on route R as \mathbf{r} , let $d(\mathbf{r}, H_i)$ denotes the distance

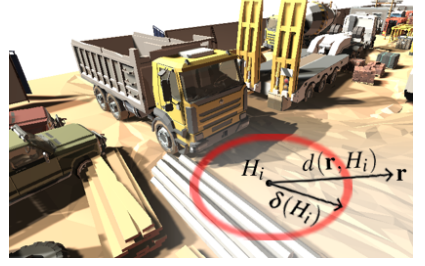


Figure 4.3: Dynamic Hazard.

between \mathbf{r} and the center of hazard H_i , let $\delta(H_i)$ denoted the affecting range of hazard H_i . Then, given the condition that if point \mathbf{r} is close enough to the hazard H_i , namely, $d(\mathbf{r}, H_i) < \delta(H_i)$, we define the conditional probability of the occurrence for each hazard H_i denoted as $\Pr(H_i | d(\mathbf{r}, H_i) < \delta(H_i))$. Therefore, we evaluate the overall probability $D(R, H)$ that any dynamic hazardous events may happen along route R as a function proportional to the summation of line integral for the conditional probability of each hazard H_i along route R as:

$$D(R, H) = \frac{k}{|R|} \sum_{i=1}^{|H|} \int_R \Pr(H_i | d(\mathbf{r}, H_i) < \delta(H_i)) d\mathbf{r} \quad (4.3)$$

where the number $|H|$ is the total amount of the synthesized hazards H , the $|R|$ is the total length of the route calculated through the line integral along the route R , and k is a fine-tuned scale factor to adjust the overall dynamic hazardous probability returning the values evenly distributed within the range $[0, 1]$, empirically, we set $k = 0.3$. Please refer to Supplementary for more details about calculating a dynamic hazard H_i 's conditional probability $\Pr(H_i | d(\mathbf{r}, H_i) < \delta(H_i))$.

Static Hazard Cost. According to the users' performance in the pre-evaluation, we synthesize personalized training scenarios to emphasize the specific skills for individuals, namely, inspect specific types of static hazards more efficiently. Theoretically, given M different types of

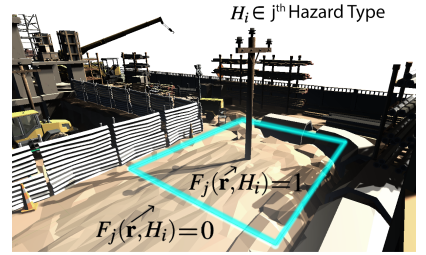


Figure 4.4: Static Hazard.

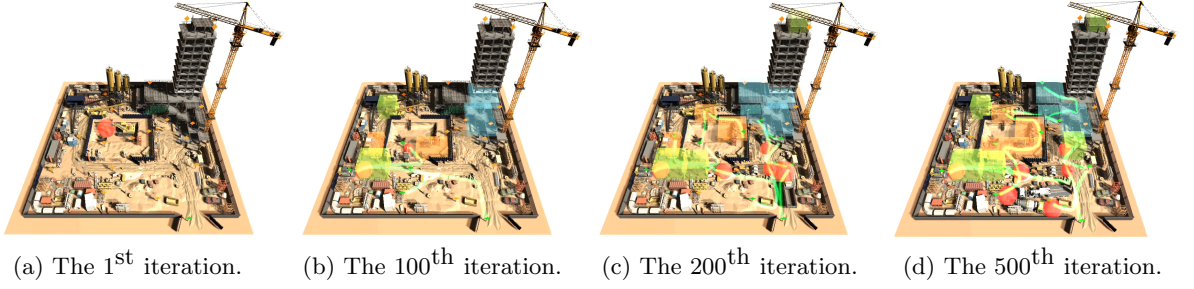


Figure 4.5: Process of optimization. The green curves denote the navigation routes synthesized through optimization. (a) The route initialized at the entrance of the construction site. During the iterations, more vertices are randomly added, replaced, or removed to form new navigation routes. At the same time, different types of hazards (colored in red, blue, yellow, or green) will be randomly added, replaced, or removed simultaneously. Figure (b-d) shows a new frame in the optimization process based on the number of iterations completed at the time. Figure (d) shows the result of the final synthesized training scenario.

static hazards, let the personalized preference weight be denoted as λ_j for the j^{th} type of hazard where $j = 1, 2, \dots, M$. If weight λ_j is higher than other weights, the static hazards in j^{th} type appear more frequently than other types of static hazards. We encode such personalized preference settings through the static hazard cost function $C_S(R, H)$:

$$C_S(R, H) = 1 - \exp\left(-\left(\frac{S(R, H)}{\sigma_S}\right)^2\right), \quad (4.4)$$

where we empirically set $\sigma_S = 0.25$ and $S(R, H)$ measures the difference between the frequency of different types of hazard appear near the route R and the target personalized preference settings:

$$S(R, H) = \frac{1}{M} \sum_{j=1}^M \left(\frac{k}{|R|} \sum_{i=1}^{|H|} \int_R F_j(\mathbf{r}, H_i) d\mathbf{r} - \lambda_j \right)^2 \quad (4.5)$$

where the hazard score function $F_j(\mathbf{r}, H_i)$ returns 1 when the hazard H_i appears within the route point \mathbf{r} 's nearby region and H_i belongs to j^{th} type of static hazard; otherwise, returns

0. $|H|$ denotes the total amount of the synthesized hazards H . $|R|$ is the total length of the route calculated through the line integral along the route R . Where k is a scaling factor to adjust each term value evenly distributed between 0 and 1, empirically, we set $k = 0.2$.

Training Time Cost. The training time cost function $C_T(R, H)$ measures the difference between the estimated training time $T(R, H)$ navigating through the route R and the target time ρ_T as:

$$C_T(R, H) = 1 - \exp\left(-\left(\frac{T(R, H) - \rho_T}{\sigma_T}\right)^2\right), \quad (4.6)$$

where we empirically set $\sigma_T = 2\rho_T$ and the total training time $T(R)$ includes the amount of time for navigating along the route R and the amount of time to inspect all hazards:

$$T(R, H) = k_1 \frac{1}{v} \int_R d\mathbf{r} + k_2 \sum_{i=1}^{|H|} T(H_i) \quad (4.7)$$

where $|H|$ is the total number of the synthesized hazards H . Pre-defined constant v is the users' default navigation speed. The line integral along the route returns the total length of the route, denoted as $|R|$. While divided $|R|$ by the constant velocity gives the traversal time along the route R . $T(H_i)$ is the estimated time to finish inspecting out the hazard H_i . Where k_1 and k_2 are two balancing factors used to balance the importance of two considerations: path navigation time and hazard inspection time. During the training, users are prompt to specify the hazards while navigating, navigation time plays a more important role in estimating the overall training time, we empirically set a larger $k_1 = 0.8$ and a smaller $k_2 = 0.2$.

4.4.2 Optimization

As the training targets such as total training time and personalized preferences are set by the users, our work is to synthesize a navigation route R and the hazards H which are optimized with respect to the total cost function. In order to synthesize realistic navigation routes, we generate the navigation mesh on our developed virtual construction sites. We randomly pick up some points of interest within the scene and put signs (such as work ahead signs) on those random points. In this way, the training routes can be assembled through those shortest navigation paths between ev-



Figure 4.6: Sign.

every two signs. Such configurations form a navigation graph structure where the signs are the vertices and the navigation paths connecting two signs are the edges connecting two vertices. Empirically, we formalize the navigation graph as a k -regular graph[137] with k -shortest distance navigation paths connecting every two vertices[138], where we set $k = 4$. Hereby, a synthesized route R consists of an arbitrary number of vertices and the edges connecting every two adjacent vertices.

As shown in Figure 4.5, the initialization of the route R contains only one vertex as both the starting point and the ending point without edges. This starting point corresponds to the location of the construction site’s entrance. We formulate the optimization problem as a graph searching problem by employing the Markov chain Monte Carlo method [139] to search for a solution that minimizes the total cost function. Given to any randomly chosen vertex v in the current route status R , the proposed new route status R' is randomly searched within the solution space through three types of move:

- *Add a Vertex*: a random vertex v' adjacent to v is added to the current route R to create a proposed route R' .
- *Remove a Vertex*: the vertex v is removed from the current route R to create a proposed route R' while keeping the connectivity.

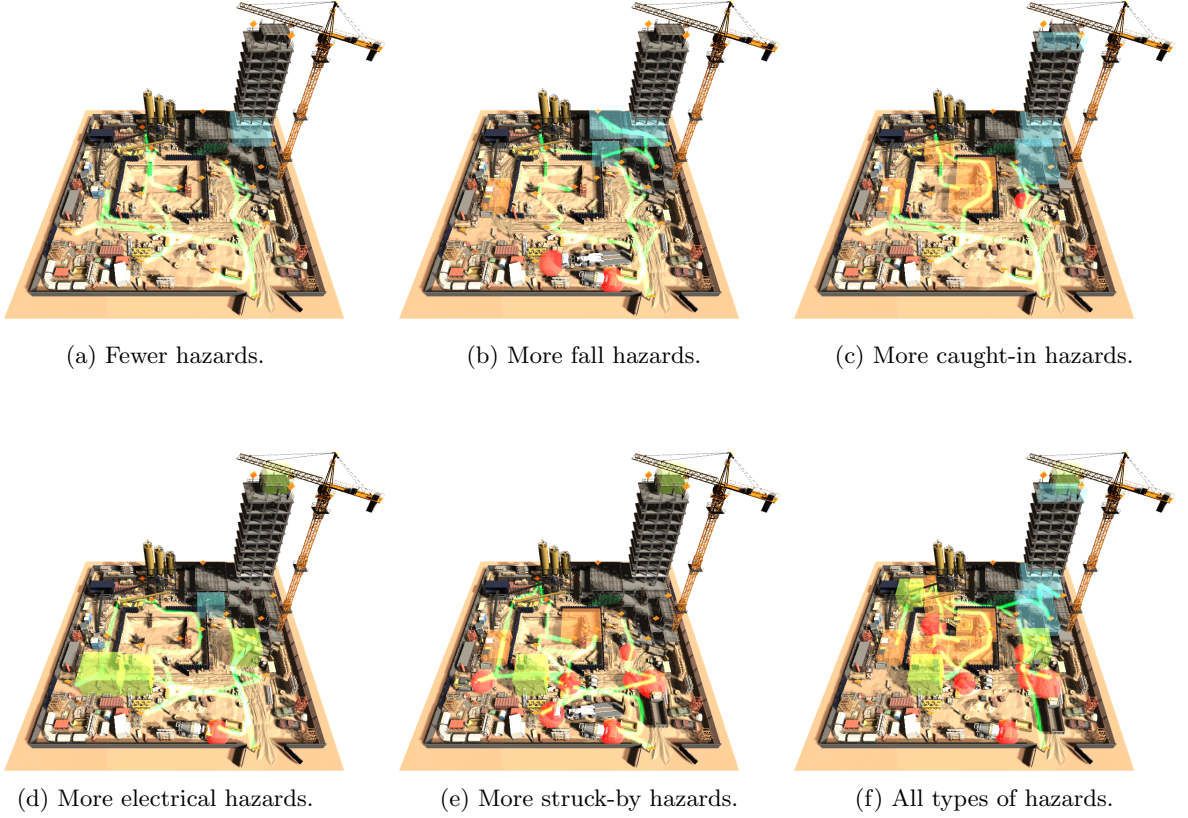


Figure 4.7: Personalized training scenarios synthesized on the same construction site. Green curves are the navigation routes. The regions with fall hazards are highlighted in blue, caught-in/between hazards in yellow, electrical hazards in green, and the struck-by hazards in red. Six different scenarios are shown, including (a) a route passing through fewer hazardous regions; (b) a route passing through more regions with fall hazards; (c) a route passing through more regions with caught-in/between hazards; (d) a route passing through more regions with electrical hazards; (e) a route passing through more regions with struck-by hazards; and (f) a route passing through regions with all types of hazards.

- *Modify a Vertex*: the vertex v is replaced by a random vertex v' adjacent to v to create a proposed route R' .

Similarly, given to any randomly chosen hazards in the current list H , the proposed new list of hazards H' are randomly searched within the solution space through three types of move:

- *Add a Hazard*: a hazard is randomly chosen to be added into the current list of hazards H to create proposed hazards H' .

- *Remove a Hazard*: a random hazard is removed from the current list of hazards H to create proposed hazards H' .
- *Modify a Hazard*: a random hazard in the current list of hazards H is replaced into another random hazard not in the current list to create proposed hazards H' .

For a proposed update of the route R' and hazards H' , in the formulation of the method by Kirkpatrick et al. [59], the acceptance probability function $\Pr(R', H'|R, H)$ is defined as:

$$\Pr(R', H'|R, H) = \min(1, \frac{f(R', H')}{f(R, H)}), \quad (4.8)$$

where $f(R, H)$ is a Boltzmann-like objective function related to a Metropolis-Hastings state searching step [103]:

$$f(R, H) = \exp(-\frac{1}{t}C_{\text{total}}(R, H)) \quad (4.9)$$

where t is the temperature parameter of simulated annealing, which decreases gradually throughout the optimization. As the temperature t decreases over iterations, the optimizer becomes less aggressive and more greedy. By the end, the temperature drops to a low value near zero, the optimizer tends to accept better solutions only. We empirically use temperature $t = 1.0$ at the beginning of the optimization and decrease it by 0.2 every 100 iterations until it reaches zero or terminated if the total cost change is smaller than 3% over the past 50 iterations.

Changing Parameters. As shown in Figure 4.7, different parameters are applied for generating different training scenarios. As declared in section 4.4.1, adjusting the parameter settings of the targets for each type of hazard will synthesize different training scenarios that emphasizing trainee's different aspects of the construction safety inspection skills. We specify a low probability of the occurrence of a hazard as 0.1 and the high probability as 0.9. As specified in Equation 4.5, let λ_F denotes the target occurrence frequency for fall hazard, let λ_C denotes the one for caught-in/between hazard, let λ_E denotes the one for

electrical hazard, those parameter settings are:

- *Fewer Hazard*: A route with fewer hazard occurred: $\lambda_F = \lambda_C = \lambda_E = \rho_D = 0.1$.
- *More Fall Hazard*: A route with more fall hazards occurred: $\lambda_F = 0.9$, $\lambda_C = \lambda_E = \rho_D = 0.1$.
- *More Caught-in/between Hazard*: A route with more caught-in/between hazards occurred: $\lambda_C = 0.9$, $\lambda_F = \lambda_E = \rho_D = 0.1$.
- *More Electrical Hazard*: A route with more electrical hazards occurred: $\lambda_E = 0.9$, $\lambda_F = \lambda_C = \rho_D = 0.1$.
- *More Dynamic Hazard*: A route with more struck-by hazards occurred: $\rho_D = 0.9$, $\lambda_F = \lambda_C = \lambda_E = 0.1$.
- *All Types of Hazard*: A route with all types of hazards occurred: $\lambda_F = \lambda_C = \lambda_E = \rho_D = 0.9$.

In general, we set the weight for training time cost $w_T = 0.2$, the weight for dynamic hazard cost $w_D = 0.4$, and the weight for static hazard cost $w_S = 0.4$. We set an overall training time $\rho_T = 8$ mins as a soft regularization term to avoid over-long or over-short training. It will result in a training scenario including a route with a total length of about 400m to 500m (2 to 4 mins walking distance) and 6 to 12 selected nearby hazards. During the training, the user may use the controller’s speed-up button to shorten the training time. Please refer to Supplementary for examples that synthesize training scenarios on another construction site with above parameter settings.

4.5 Experiments

4.5.1 Implementation

We implemented our approach using C# and Unity 2020.1.11. The VR training runs on a PC with 64.0 GB RAM, a 3.60Hz Intel(R) Core i7-9700K CPU processor, a Fresco Logic IDDCX Adapter graphics card, and using a Samsung HMD Odyssey+ with Samsung Hmd Odyssey Windows Mixed Reality Headset and 2 Wireless Controllers (XE800ZAA-HC1US).



Figure 4.8: Three virtual construction sites for user studies: (a) Construction Site 1 is used for pre-train and VR training. (b) Construction Site 2 is used for VR training. (c) Construction Site 3 is used for pre and post-evaluation. By differing the construction sites used for training and the ones used for evaluations, we avoid the biased result in the user study that users might repeat similar tasks in VR training that also appear in evaluations if both scenarios were synthesized on the same construction site.

4.5.2 Training and Evaluation

We conduct a series of user studies to measure the effectiveness of the construction safety virtual training scenarios synthesized with our approach. Each user study consists of four sessions: (1)pre-train, (2)pre-evaluation, (3) training, and (4)post-evaluation. As shown in Figure 4.8, we have designed three different construction sites applied to different training sessions. The Pre-train is synthesized on the first construction site, the VR training scenarios are synthesized on the first and second construction sites. The pre-evaluation and the post-evaluation share the same training scenario, which is synthesized on the third construction site.

Participants. We recruited 24 participants among whom most are college students age between 20 to 30 and randomly recruited from different majors such as electrical engineering, computer science, data analysis, art majors, and most have insufficient background knowledge of construction safety. In order to compare the effectiveness of our synthesized personalized VR training with other types of training, those participants are randomly and evenly separated into two groups: one is the control group and another is the VR group.

Pre-Training. In order to eliminate the negative bias of user study results among the



Figure 4.9: Two operations on controllers: (a) the button on the left controller is used to control the navigation speed while the button on the right controller is used to click the correct answers. (b) Four answer buttons pop up after the user clicks on the "Select Hazard" button. When the front lightsaber on the right controller intersects with, for example, the "Caught-In" button, that button starts twinkling. Until the user clicks with the front trigger on the right controller, that answer will be submitted, and all the answer buttons will disappear.

users who are not familiar with the VR controllers, we prepare a pre-train, a short warm-up training session, to get users familiar with how to use VR controllers correctly. During this training session, the users from both control and VR groups are put into the first construction site with no performance evaluations recorded. The goal of pre-train is to let the users practice two basic operations: one being the navigation control and another being the answer selection. As shown in Figure 4.9, (a) navigation control operations are based on clicking a speed button on the left-hand controller includes (1)start moving(one click), (2)stop moving (another click), and (3) speeding-up(keep pressing the button while moving). When users move into any hazardous region, a button captions "Select Hazard" popups; users are prompt to select the hazard type by clicking that button with the front trigger on the right-hand controller. As shown in Figure 4.9(b), four buttons specifying different hazard types, once the user clicked any one of them, the answer will be submitted to the system.

Pre-Evaluation. The pre-evaluation is synthesized on the third construction site and

optimized with the parameter settings: *All Types of Hazard* as specified in Section 4.4.2. In this session, users from both groups are put into the same training scenario to evaluate their background knowledge of construction safety. Users who have no background in construction safety are encouraged to intuitively select the answers. As the accuracy of identifying different types of potential hazards is crucial for evaluating safety training effectiveness, we track the user’s answers during the evaluation and compare their answers with the correct answers, the numbers of the mistakes that users have made for each type of hazards are counted. Please refer to Supplementary for detailed evaluation tasks.

Training. During this training session, the 12 users from the control group are not trained from the synthesized training scenarios in the virtual scene, instead, they are taught about the construction safety fundamental knowledge through a short lecture presentation using slides. On the slides, there are real-case pictures demonstrating examples from different types of hazards. During the presentation, the lecturer carefully goes through each slide and explains each type of hazard’s concept. At the same time, participants are encouraged to ask questions to make every knowledge point clear. By the end of the training, the lecturer gives a Q&A session to make sure the participants fully understand the content covered in the lecture and are ready for the post-evaluation. Please refer to the Supplementary for detailed content in the lecture slides.

Different from the control group, the 12 users from the VR group are trained with personalized synthesized training scenarios which depend on their performance in pre-evaluation. Given the parameters settings in Section 4.4.2, users will be put into *More Fall Hazard*, *More Caught-in/between Hazard*, or *More Electrical Hazard* according to which types of hazards they have made mistakes mostly during the pre-evaluation. If the users perform well and have less than two mistakes in all types of hazards separately, they will be put into only one training session using the setting as *All Types of Hazard*. All of the personalized virtual training scenarios are optimized on the first and second construction sites to avoid the similarity with the post-evaluation, which is optimized on the third construction site. During the VR training, the instructor will ask the users to slow down and inspect

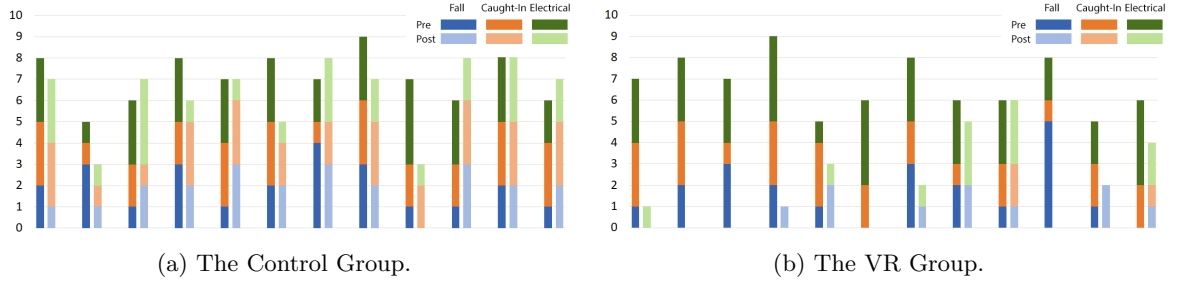


Figure 4.10: Inspection errors. The number of mistakes users made on each type of hazard during the pre-evaluation and post-evaluation are plotted in this diagram. Darker colors represent the number of mistakes made in pre-evaluation while the lighter colors represent those in post-evaluation. Blue, yellow, and green colors denote the errors made on inspecting fall, caught-in/between, and electrical hazards, respectively.

if there is any hazard around them. The instructor will explain why there are such types of hazards by pointing out that hazardous objects in VR. Sometimes if there is no obvious hazard around, the instructor also asks users whether there are any hazards. If the answer is yes, the instructor will explain why that is not the proposed hazard and correct the users' misunderstanding.

Post-Evaluation. In order to investigate whether after the lecture-based training or after personalized VR training, users' accuracy in detecting different types of hazards has improved, the same tasks in pre-evaluation were assigned to the users again. In this session, the instructor has no hints or suggestions for the users. Similar to the pre-evaluation, the correctness of the users' answers is evaluated as well. After having the users completed the assigned post-evaluation tasks, the users were prompted to complete a questionnaire. The questionnaire offered different questions for feedback on the virtual evaluation tasks, virtual training tasks(if applicable), and the effectiveness of our training programs to improve their proficiency in inspecting different types of hazards.

4.6 Results and Discussion

We analyze and discuss the results of users' accuracy in hazard inspection from pre and post evaluations and compare the difference between the users' improvements from the control group and those from the VR group. As shown in Figure 4.10, the diagram plots the number of mistakes users made on each type of hazard during the pre-evaluation and post-evaluation. Darker colors represent the number of mistakes made in pre-evaluation, while the lighter colors represent those in post-evaluation. Different types of hazards are represented with different colors: blue denotes the fall hazard, yellow denotes the caught-in/between hazard, and green denotes the electrical hazard. For each type of hazard, we measure the improvement of the users' performance before training and after by subtracting the number of mistakes made in post-evaluation from the number of mistakes made in pre-evaluation. Noted that assume users make improvements, then the number of mistakes in post-evaluation is supposed to be less than the number of mistakes in pre-evaluation. In that case, the numbers measuring users' improvements are supposed to be positive. Please refer to Supplementary for users' answers for evaluation tasks.

Statistical Analysis. According to the descriptive statistics, for those users from the control group, the average improvement in inspecting the fall hazards, the caught-in/between hazards, and the electrical hazards are 0.08, -0.08, and 0.75, respectively. In total, the average improvement is 0.25. For those users from the VR group, the average improvement in inspecting the fall hazards, the caught-in/between hazards, and the electrical hazards are 0.91, 1.83, and 2.0, respectively. In total, the average improvement is 1.58. Therefore, in general, we conclude that users from both the VR group and the control group have improved their accuracy in identifying different types of hazard, however, users from the VR group improves more than those from the control group on average.

Furthermore, we analyze whether there is any statistically significant difference between the improvement made by the users from the control group and the users from the VR group. We applied two factors ANOVA tests (with replication)[140] for users' improvement

on each type of hazard. Two factors null hypothesis includes:

- *Among groups*: there is no statistically significant difference between the users' improvements from the control group and the users' improvements from the VR group.
- *Among columns*: there is no statistically significant difference between users' improvements in inspecting three different types of hazards.
- *Interaction*: there is no statistically significant interaction between two factors: the difference between the two groups and the difference between three types of hazard.

By setting $\alpha = 0.05$ (95% confidence interval), we get the ANOVA test result showing that among different groups (VR group and control group) $P_{\text{value}} = 0.00005 < 0.05$, among different columns (three hazard types) $P_{\text{value}} = 0.07 > 0.05$, and the interaction $P_{\text{value}} = 0.35 > 0.05$. Therefore, with 95% confidence, we reject the null hypothesis among groups that there is no statistically significant difference between the improvements made by the users from the control group and the improvements made by the users from the VR group. Alternatively speaking, with 95% confidence, we conclude that the users from the VR group who are trained through our personalized synthesized training scenarios improve significantly more than the users from the control group who are trained through the lecture-based presentation using slides.

Common Mistakes. From the user study, we conclude some common mistakes made by the users. As shown in Figure 4.11, such common mistakes include (a) mistaken struck-by as fall hazard, (b) mistaken caught-in/between as fall hazard, (c) mistaken electrical hazard as none of above, and (d) mistaken caught-in/between as none of above. For the users from the control group, after the lecture-based training, they sometimes make those common mistakes again. Reasons for those mistakes are concluded as below:

- **Struck-by as Fall Hazard.** Users mistake the falling objects as fall hazard, actually, falling objects are dynamic hazards that can strike people, which causes a struck-by hazard.

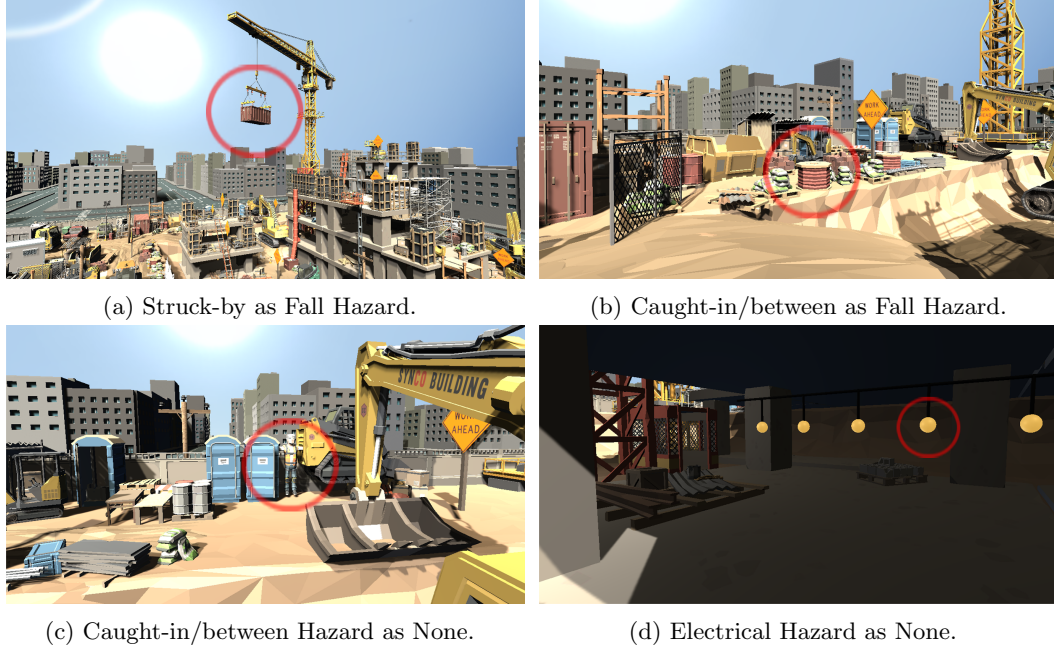


Figure 4.11: Examples of common mistakes: (a) mistaken struck-by as fall hazard; (b) mistaken caught-in/between as fall hazard; (c) ignored caught-in/between hazard; (d) ignored electrical hazard.

- Caught-in/between as Fall Hazard. Users mistake the objects near the cave falling down into the cave causes the fall hazards, actually, heavy storage near the cave can collapse the cave and caught the people inside the cave, more professionally, this causes a cave-in hazard, which is one type of caught-in/between hazard.
- Caught-in/between Hazard as None of Above. Sometimes, people caught-in between two objects and potentially being crushed from two sides are easily ignored.
- Electrical Hazard as None of Above. Sometimes, daily-used electrical components such as temporary lights suspended by their electric cords are easily ignored. However, ignoring these potential electrical hazards sometimes causes the death of those people who are exposed to damaged wires with high voltage.

User Feedback. With our designed questionnaire we collect feedback from the users in both groups. As shown in Figure 4.12, we statistically discover that most of the users

agree that: our VR construction safety training and evaluation programs are realistic and comfortable; The animation of VR construction simulations looks natural; The control of VR navigation feels natural; The training tasks in VR or training lectures are enjoyable. And most of the users believe that through the training (either by VR or by slides) their construction safety knowledge and skills are improved. At the same time, most of the users believe post-evaluation is easier than pre-evaluation. This is reasonable even post-evaluation is the same as pre-evaluation due to that users have learned more concepts and definitions of different types of hazards through the training sessions (either by slides or VR) so that when they face the same questions they are armed with more knowledge to handle them confidently.

4.7 Summary

We have devised an optimization-based approach to synthesize personalized training scenarios for construction safety training in virtual reality. We developed several virtual scenes for different construction sites, generated the navigation meshes for the scenes, and analyzed different types of common hazards such as struck-by, fall, caught-in/between, and electrical within the scenes. By specifying a couple of points of interest as vertices, our approach automatically simulates the agent's navigation paths between every two vertices and generate a k -regular navigation graph with k shortest distance paths (Set $k=4$). As shown in Figure 4.7, by optimizing the navigation route along with the sequence of hazards simultaneously in a fully automatic manner, our approach can synthesize personalized training scenarios with more fall hazards, more caught-in/between hazards, or more electrical hazards to efficiently improve the trainee's specific skills.

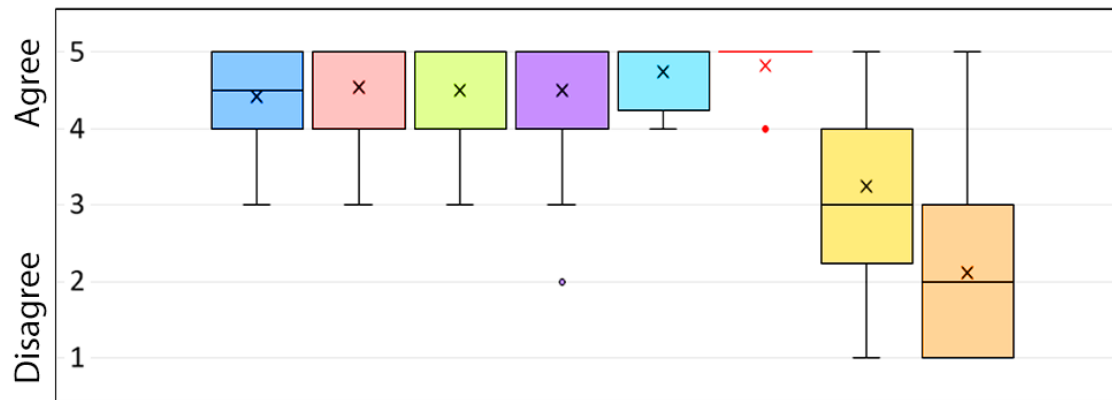
In order to measure the difference between our proposed VR training method and other training methods such as lecture-based training, we separate the users into VR group and control group. We prepare a short warm-up training session for both groups to get the users familiar with how to use VR controllers. Before or after the training session, both groups are put into the same pre or post-evaluation scenario. According to users' performance in

pre-evaluation, the users from VR group are put into the personalized VR training scenarios and navigate through our optimized routes. Along with the navigation, users are prompted to inspect the nearby hazards with a VR controller. Users from control group are trained with slides. When users from both groups finished the training, we post-evaluate their performance and investigate their improvement.

After we applied two factors ANOVA tests (with duplication) for users' improvement on inspecting each type of hazard, as shown in Section 4.6, with 95% confidence we conclude that users from the VR group improved significantly more than those users from the control group. We designed a questionnaire to collect the users' feedback on their training experience, according to their feedback, we discover that most of the users agree that our VR construction safety training and evaluation programs are realistic, comfortable, and enjoyable.

Limitation. According to the users' feedback, most of them like this VR training and evaluation program design. For example, some users like the questions' settings and the immersive construction site where those hazardous objects are carefully placed. Some users love the lightsaber shooting from the right controller, while some users like the feeling of being elevated by the auto-navigation. However, some users provide negative feedback, such as they feel a little dizzy when they tried speeding up. Some users claim that this program cannot be comfortable for older people to play. Sometimes users can run into virtual objects if their VR headsets were not correctly collaborated with the scene's center. Sometimes right controller's front trigger does not react instantly and perhaps needs to click twice for submitting an answer. Please refer to Supplementary for more details of users' feedback.

Future Work. We are interested in the following future explorations that may enhance users' construction safety training experiences in VR. For example, it can provide more hazard inspection tasks or increases the number of surrounding hazardous objects to avoid long tedious navigation in the scene which causes dizziness. It can increase the scale of construction sites and incorporate more diverse types of hazards to make virtual training



- VR construction safety training or evaluation is realistic.
- VR construction safety training or evaluation is comfortable.
- The animation of the VR construction simulation looks natural.
- The control of VR navigation is natural.
- The training or evaluation tasks are enjoyable.
- Through the training, users think their construction safety knowledge is improved.
- The pre-evaluation is difficult.
- The post-evaluation is difficult.

Figure 4.12: Users' Feedback.

more complex and professional. It can label different types of hazardous objects to make VR training sessions more effective. It can also add audio narration in the VR training to help the user understand the concepts and definitions of different types of hazards. Therefore, future work will focus more on improving the training quality and implementing a more immersive virtual training environment with more impressive virtual experiences.

Chapter 5: Conclusion and Future Work

5.1 Summary

This dissertation has solved the two problem statements specified in Section 1 through our hypotheses by demonstrating how optimization approaches can be applied to synthesize training scenarios and exergames. Inspired by the recent works that automatically synthesize different training scenarios to achieve different personalized training targets through optimization techniques that can be applied to substitute manual configurations, this dissertation is proposed along with two research topics that are aiming at synthesizing personalized training scenarios where the first one being the automatic synthesis of virtual wheelchair training scenarios and the second being the exertion-aware path generation for VR biking. As shown in Chapter 2, we conducted an evaluation to validate the efficacy of our proposed virtual wheelchair training approach. Users showed improvement in wheelchair control skills in terms of proficiency and precision after receiving the proposed virtual reality training. As shown in Chapter 3, our user study results showed that users found exercising with our generated paths in virtual reality more enjoyable compared to traditional exercising approaches. Their energy expenditure in biking the generated paths also matched with the specified targets, validating the efficacy of our approach. As shown in Chapter 4, we validated our approach by conducting a series of user studies during which we get the users trained with our synthesized construction safety training scenarios and analyze users' performances before VR training and after. All in all, personalizing VR training experiences through optimization is proved by our approaches that it can deliver the users effective training effects without demanding extra manual effort from the training program designers or the exergame level designers.

5.2 Limitation

5.2.1 Hardware-Related Limitations

During the user studies, most of users like the synthesized VR training programs or exergames using our approaches. However, some of users provide negative feedback such as they will feel a little dizzy when they tried speeding up in VR environment, some users suggest that this program can be not comfortable for older people to play with. Therefore, 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. Therefore, according to these users' feedback, those hardware-related limitations and side effects are listed below:

User Population Limitation. Due to the hardware constraints of VR devices, the target population to be trained in our synthesized training programs exclude those who would be at greater risk of injuries, such as those with epilepsy, vertigo, those who suffer from motion sickness or have ongoing headaches and those who are easily getting nausea caused by motion, especially by traveling in a vehicle. Besides, individuals with pacemakers or other medical devices that could be impacted by the magnets in the headset are excluded.

Potential Side Effects of VR Training. There are potential side effects of VR training for those people are not belongs to the target user population such as the risk of seizure, migraines, motion sickness, dizziness, headache, and risk of infection from others using the device. Participants may feel some discomfort such as possible dizziness associated with the VR navigation. As specified in the Virtual Reality Headset warnings document: We will withdraw their participation when the user feels tired, sleepy, under the influence of alcohol or drugs, being hung-over, having digestive problems, under emotional stress or anxiety, or suffering from cold, flu, migraines, seizure, nausea or headaches. Therefore, users who are at increased risk of seizures, migraines, motion sickness, dizziness, and any medical devices/implants affected by magnets are prevented from using our VR training programs.

5.2.2 Assessment-Related Limitations

The main steps of our proposed research methodology in general from a high-level point-of-view can be illustrated as: Given any interesting training story, we first parameterize the training scenarios according to their key features so as to enable the users to adjust their personalized settings. Then, by proposing an optimization-based approach, synthesize training scenarios considering the user-specified parameter settings in a fully automatic manner. And in the end get users trained in the synthesized training scenarios and statistically evaluate their improvements. Therefore, in our training program, training evaluations are mainly focused on identifying immediate participant reactions to the overall training and measuring participant level of learning upon completion of the synthesized training scenarios. Although given to the current technological advancements that have impacted both the delivery of training (virtual and simulated) and the evaluation of training (biometrics and analytics) and the VR simulation training can be assessed and evaluated through a number of tests including both quantitative, qualitative, and biometrical ones, it is still questioned that how much extend to which the virtual training can substitute the real-world training. Therefore, there still are assessment-related limitations that can not eliminate the gap between VR training and real-world training. Such limitations are inevitable and mainly caused by the mental gaps and the physical gaps between virtual and real training.

5.3 Future Work

As personalizing VR training experiences through optimization is a general research topic, it can be specialized into multiple interdisciplinary research topics. Therefore, here those potentially interesting topics that are relevant to personalizing VR training are discussed.

5.3.1 Personalized Public Safety Training Synthesis

As inspired by Querrec et al. [45] that multiagent-based virtual environment can be used to help for firemen fire fighting training, it will give us a promising working direction that

synthesize fire fighting training that considering user-specified training goals such as traffic condition, fire hazard types and estimated training time using optimization approaches. It can also be a good topic to synthesize a personalized emergency escape training in virtual reality[141] according to users' personalized preferences such as the complexity of the indoor furniture arrangements, total escaping distance, and different types of hazardous source. As proposed by An et al. [142] that VR system can be effective for chemical incident response training for incident response staff, it can also be a good topic to train users through personalized chemical incident response training scenarios that are taking considering the personalized factors such as synthesizing laboratory tasks sequences with respect to highly poisonous, highly explosive, and highly radioactive operations. Evacuation training can also be personalized given to the existing work on VR drills for evacuations from buildings[143], subways[144], campuses[145], and nightclubs[146] by considering the training factors such as crowd density, events hazardousness, and spatial narrowness. As aviation safety training [147] can also be personalized, factors such as flight height, flight distance, and landing conditions can also be optimized with the user-specified targets. Venue safety training[148] and traffic safety training[149] can be personalized to be suitable for children's education which can efficiently train little children about avoiding dangerous traffic events that potentially happened or the hazardous constructional objects that are placed nearby.

5.3.2 Personalized Law Enforcement Training Synthesis

As some existing successful VR-based training platforms for law enforcement training for police such as the one designed by Caserman et al. [150] and the one developed by Conway et al. [151] prove that the effectiveness of virtual police training program and user's skills learned within the serious games are transferable to the real work tasks. Therefore, it is meaningful to switch the research focus onto synthesizing personalized law enforcement training. As proposed by Sharm et al. [152], an immersive collaborative VR platform simulating shooting scenarios is proved to be successful to teach individuals how to respond appropriately in such situations according to the "run, hide, and fight" model. Such a

shooter training program can be suitable for educating employees on how to protect themselves during shooting scenarios in the workplace and training professionals including police officers and soldiers. As future work, personalized factors such as interested virtual locations, number of corners or road crosses, crimes escaping speed and total training time should be optimized according user-specified training goals. Synthesized personalized law enforcement training scenarios can train users to quickly locate and reach the crime area, and locate the target object among a bunch of virtual objects. During the training, there might be many VR agents walking around on the route. Some agents might have potential criminal behaviors. The user needs to judge whether the agent will commit a crime based on the agent’s behavior and the surrounding conditions. If the agent can be identified in advance to commit a crime, the user wins.

5.3.3 Personalized Exergame Level Synthesis

On the exergame side, exercise-driven terrain generation[153] can be a good topic to explore: Given a serial of user-specified treadmill interval workout plan table including inclines angles changing with time, our future work can predict the path elevations/heights using the geometrical calculations. After combining the path heights with the user-specified urban layout and path geometry, we will synthesize realistic terrains using a generative neural network[154] to match the target path heights with the heights of the path projected on the terrain. Similarly, as future work, the rowing machines will be accompanied by exertion-aware riverscapes generation[155], which will achieve both realistic visual effects and user-desired exertion effect through procedural modeling and optimizations. Another promising future work is that the versaclimber[156] workout for weight loss can be combined with realistic virtual canyons procedural modeling[157] techniques to synthesize the players’ visual experience when climbing the machine as if climbing the cliff in the canyons. Such exergame experience can be optimized with the target climbing distance, climbing speed and climbing strength which is corresponding to the parameters for canyons’ inverse procedural modeling through optimizations.

Appendix A: Exertion-Aware Path Generation

A.1 Experiments

A.1.1 Obstacles

Here is one more example of forbidden zone design. In this case, during optimization, paths will avoid the manually specified obstacle-Armadillo standing in the center of the terrain. Users can specify any bounding area around the Armadillo as the forbidden zone as shown in Figure A.1.

A.2 User Study

A.2.1 Details of Enjoyment Test

To test the enjoyment of biking generated paths using our VR setup, we recruited 23 participants to bike our generated paths. They were university students and staff. They were 23 years old on average and their average body mass index was 23kgm^{-2} . 11 were males, 11 were females, and 1 did not disclose his/her gender.

Procedure.

Each participant was briefed about the experiment process and given a warm-up session to get familiar with the bike. Then the participant was asked to ride a generated path (from Section 5.1) in three different exercise modes (VR mode, TV mode and Gym mode) given in a randomized order. In the VR mode, the participant biked while viewing the path via the VR headset. In the TV mode, the participant biked while seeing the game view on a television screen. In the Gym mode, the participant biked while watching some basic information of the current exercising progress like using a regular biking machine in a gym. Note that in all modes the participant used our custom-built bike whose force feedback changed according to the path's elevation angle. After playing each mode, the participant

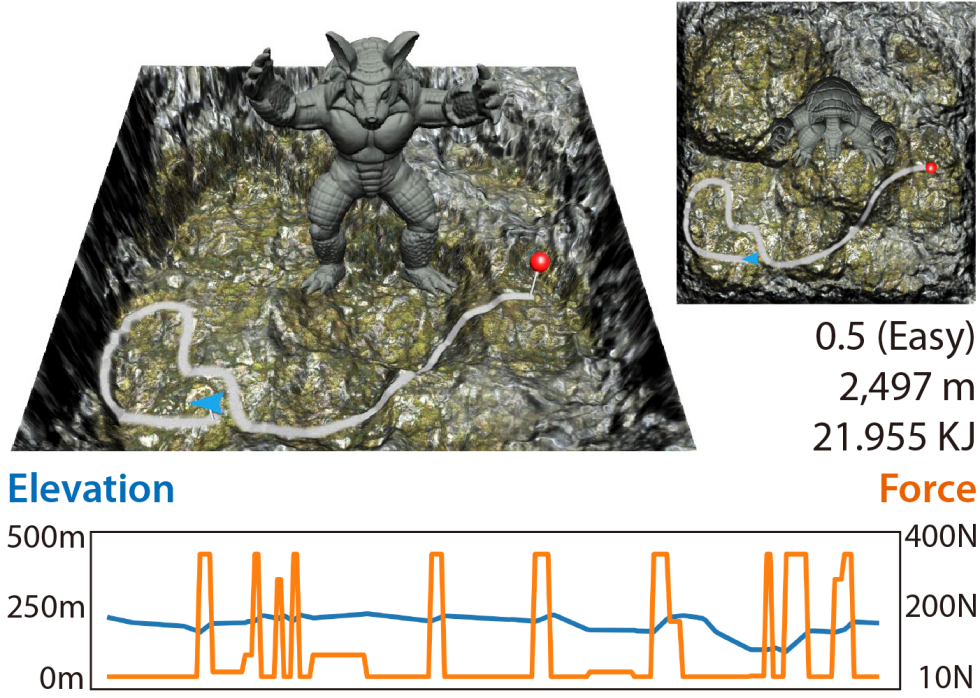


Figure A.1: Paths generated with the avoidance of manually specified obstacles which is an Armadillo in this case.

took a 3-minute break. During the break, we asked the participant to fill out a questionnaire about the enjoyment as shown in Table A.3.

Results.

We examined descriptive statistics on our participants' performance using a Friedman Test to detect if there was an overall difference across the three different modes. We also conducted Wilcoxon signed-rank tests to pinpoint where the difference was located in each pair of modes. (e.g., VR-TV pair).

Figure A.2 shows the physical activity enjoyment scale questionnaire (PACES) results. PACES [158] is a quantitative measurement in exercise science to examine one's perceived enjoyment level of exercise activity. We used a modified version [159] for our user evaluation. In the Friedman Test, there was a statistically significant difference in average ratings of the three different modes ($\chi^2 = 23.079, p < 0.017$).

A post hoc analysis with Wilcoxon signed-rank tests was conducted with a Bonferroni correction applied, resulting in a significance level set at $p < 0.017$ ($\alpha = 0.05$, $n = 3$). The medians (IQR) for the PACES questions across the VR, TV and Gym modes were shown in Figure A.2. Significant differences were found in average ratings between VR and TV ($Z = -3.052, p = 0.002$), VR and Gym ($Z = -3.621, p < 0.001$), as well as Gym and TV ($Z = -3.480, p = 0.001$). From the results, we conclude that the participants generally preferred exercising in virtual reality since it was more enjoyable.

A.2.2 Conversion Between Expected Feedback Force and Expected Power Output Target

As described in Section 4.2 of our main chapter, we used the resistance force $f(t)$ to compute the total work cost $C_W(\mathcal{P})$. The resistance force $f(t)$ is formulated in Newton. In our custom-built bike, we obtain feedback power (in watts) from the power meter.

We conducted a trial experiment with 10 participants to obtain conversion between each level of feedback force of our bike and feedback power output. There are six discrete levels of feedback force in our exercise bike. For each feedback level, we asked them to ride the bike with a constant speed of 60 rpm for two minutes. Then we obtained the average power output of that level from a power meter.

Then we can analyze the participants' energy expenditure during the exertion test by comparing their power output results during the entire gameplay with the expected power output targets of the generated paths.

A.2.3 Additional Details of Demographic Information

Enjoyment Test.

We asked our participants to complete a generated path (randomly selected out of *Rocky Hill*, *Terrace Hill*, and *Desert Hill*) under three different exercise modes (VR mode, TV mode and GYM mode) given in a randomized order. Table A.1 shows the demographic of our participants.

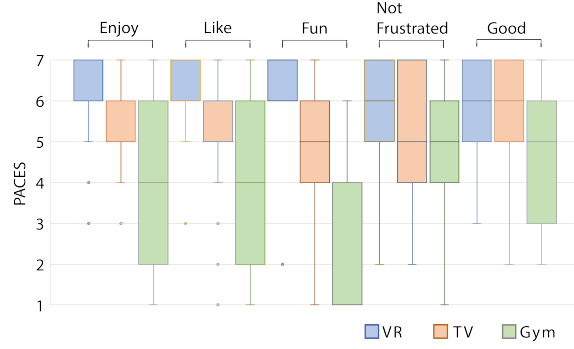


Figure A.2: Enjoyment test. PACES results for different modes.

Table A.1: Demographic information of participants (Mean (SD)) in the enjoyment test.

| Demographic Information | |
|--|--|
| Average age | 23 years old (3) |
| Gender | Male (47.8%), Female (47.8%) Prefer Not To Say (4%) |
| Weight (kg) | 68.08 (28.398) |
| Height (cm) | 169.8 (10.2) |
| BMI (kg/m ²) | 23.08 (7.19) |
| Are you an athlete? (do sports regularly) | Yes (22%), No (78%) |
| How often do you exercise with indoor bike? 1=never, 5=often | 1.39 (0.7) |

Exertion Test.

Similarly, we asked another group of participants to ride the Hard 15kJ and Hard 30kJ paths of the *Rocky Hill* for an exertion test. Their demographic information is shown in Table A.2.

Table A.2: Demographic information of participants (Mean (SD)) in the exertion test.

| Demographic Information | |
|---|--------------------------|
| Average age | 26 years old (6) |
| Gender | Male (60%), Female (40%) |
| Weight (kg) | 60.5 (9.1) |
| Height (cm) | 169.0 (6.6) |
| BMI (kg/m ²) | 21 (3) |
| Are you an athlete? (do sport regularly?) | Yes (0%), No (100%) |
| How often do you exercise with indoor bike? 1=never, 5=often | 1.5 (0.7) |
| How often do you exercise with outdoor bike? 1=never, 5=often | 1.5 (0.8) |

A.2.4 Additional Analysis of User Study Results

Deviation From Target Power Output in Exertion Test

As described in section 6.2, we observed that overall the results were close to the targets. On average, the energy expenditure for the Hard15kJ path was 17.8kJ and that for the Hard30kJ path was 27.9kJ. In order to explain such slight deviations between the user study result and the specified target, we introduce the concept of inertia of power I_P . By the definition, inertia is a property of matter by which it continues in its existing state and it influences on how much time it takes to change from a current status into a new status. Let users' power output $p(t) = f(t)v(t)$ and $v(t)$ be the velocity of a moving object at time t and $f(t)$ is the feedback force.

From the observations, when the target power output $P(t)$ increases due to a increasing feedback force $f(t)$, a user tends to decrease his velocity $v(t)$ to maintain his current power output $p(t)$, which means, it takes seconds for him to increase the speed back to normal in order to adapt himself into a higher target power output level. Therefore, his power output is lower than expected target power output during this time periods. This explains that the energy expenditure for the Hard-30 path was 27.6kJ which is a little bit lower than the target 30kJ.

Similarly, when the user is riding at a stable velocity $v(t)$, if the target power output decreases as feedback force $f(t)$ decreases, the user tends to increase his velocity to maintain current power output $p(t)$. Thus, his power output is higher than expected target power output $P(t)$ during this periods. This explains that the energy expenditure for the Hard-15 path was 17.8kJ which is a little bit higher than the target 15kJ.

Mathematically, these phenomenon can be easily explained by further exploring the power output's inertia effect. Assuming the value of the power output is a status, then at time t the current power output $p(t)$ is affected by the target power output $P(t)$ through differential equation:

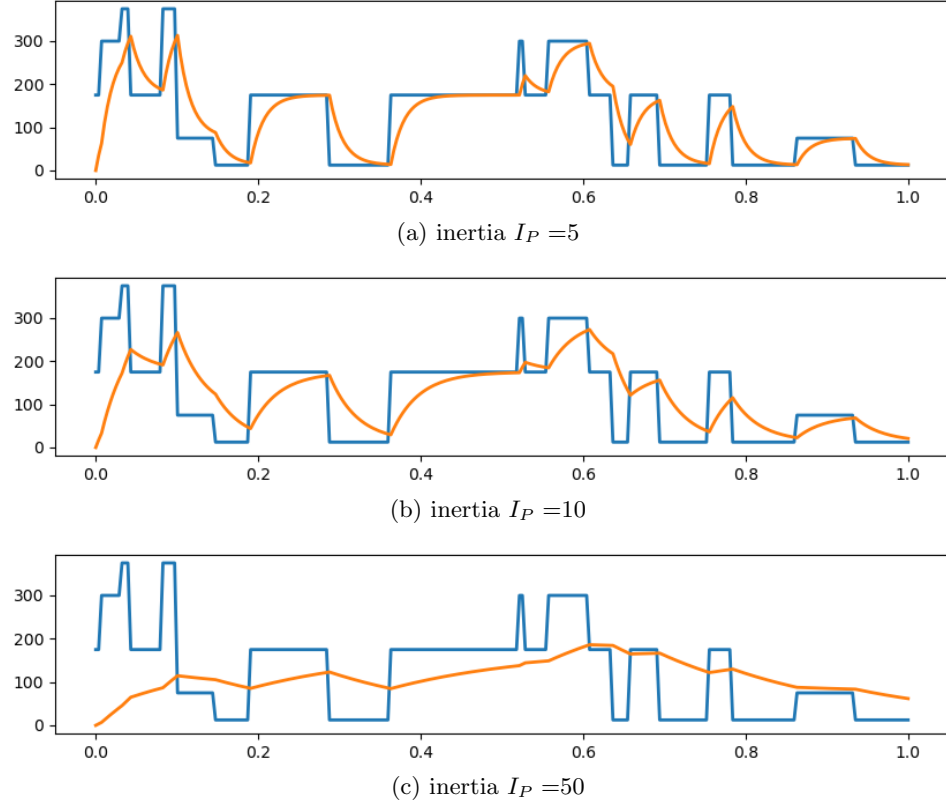


Figure A.3: Illustration of the inertia effects: Blue curve is the targets power output $P(t)$ from Hard 30kJ. Yellow curve is the predicted power output $p(t)$ using Equation A.1 which is closer to the user study result.

$$\frac{dp(t)}{dt} = \frac{P(t) - p(t)}{I_P} \quad (\text{A.1})$$

where I_P is the inertia of the power output.

Therefore, we validate this assumption by approximating predicted power output $p(t)$ using Equation A.1 by setting inertia $I_P = 5, 10, 50$ respectively shown in Figure A.3, from the trend of these plots, as the inertia $I_P =$ increases, the power output curve becomes more resistant to the target change and in turn introduces more deviation from the target settings. As matters of fact, our custom-built bike is flexible enough to catch up with target settings so that the deviation is relatively less significant and ignorable. However, our observation can be further studied for reducing the deviation for larger-inertia bike

Table A.3: PACES items. Please rate how you feel at the moment about the physical activity you have been doing.

| | | | | | | |
|--|---|---|---|---|---|--------------------------------------|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| * I enjoy it | | | | | | I hate it |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| I dislike it | | | | | | I like it |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| It's no fun at all | | | | | | It's a lot of fun |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| *I fell good physically while doing it | | | | | | I feel bad physically while doing it |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| I am very frustrated by it | | | | | | I am not at all frustrated by it |
| *Item is reverse scored (e.g., "7" becomes "1") when we report it in the main chapter. | | | | | | |

setting-up to produce more accurate exertion-aware path generations.

Appendix B: Synthesizing Construction Safety Training Scenarios for VR Training

B.1 Lecture-based Training

During this training session, the 12 users from the control group are not trained from the synthesized training scenarios in the virtual scene, instead, they are taught about the construction safety fundamental knowledge through a short lecture presentation using slides. As shown in Figure B.1 and Figure B.2, the slides' content for lecture-based training covers the fundamental knowledge for construction safety training. Those details for the lecture include:

Introduction. As shown in Figure B.1 (a), page 1 is a welcome page. As shown in Figure B.1 (b), page 2 covers the brief introduction of construction safety where the importance of hazard inspection is emphasized. It will be stressed on this page that the ignorance of hazard inspection knowledge sometimes causes the death of the people and economic losses.

Construction Signs. As shown in Figure B.1 (c), page 3 introduces the common hazardous events and the construction signs that need attention. Users are taught to be careful about the construction signs such as work ahead signs that there might be hazards nearby. This will address the importance of safety inspection skills for workers who are put in construction sites to identify and avoid common types of hazards.

Dynamic Hazards. As shown in Figure B.2 (a), page 4 introduces the dynamic hazard including struck-by hazard and any type of moving hazard that might be able to strike people. On this page, the users are trained to understand that moving vehicles such as trucks,

swinging excavators, and driving cranes are typical struck-by hazards and need critical attention when they appear nearby. Especially, it will be emphasized that falling objects that might strike people are struck-by hazards instead of fall hazard.

Fall Hazards. As shown in Figure B.2 (b), page 5 introduces the fall hazards caused by the falling of people and cause death or injuries. It will be repeatedly emphasized that falling objects do not belong to fall hazards, instead, they are struck-by hazards. Fall hazards mostly are caused by missing protection or improper rail guards on high opening areas and un-tied workers stepping on the top cap of ladders.

Caught-In Hazards. As shown in Figure B.2 (c), page 6 introduces the caught-in hazards caused by keeping materials and equipment near the edge of the cave, which might collapse the cave and caught the people inside the cave, also called cave-in hazard. As another example, it is emphasized that caught-in hazards happen when people are crushed from two sides and be constrained in a close space so that it becomes hard for escaping. Especially, the users are taught that when considering the caught-in directions, it needs to consider both the ones from left to right and from up to down.

Electrical Hazards. As shown in Figure B.2 (d), page 7 introduces the electrical hazards caused by the electrocution from the nearby electrical components, such as the light pole, power lines, and the lighting fixtures that are suspended by electric cords. Especially, users are taught to be careful about any objects in front of them by looking above, some of the electrical components are hanging high.

Construction Safety Training Fundamentals

Presenter: XXXX



(a) Slide Page 1

What is Construction Safety Inspection?

Pay attention to the hazards!
It is important for the workers' life!



(b) Slide Page 2

What should you do when you are put in a construction site?

Avoid the hazards that are moving towards you.
Classify the hazards and point them out.



(c) Slide Page 3

Figure B.1: Lecture-based Training Slides(Part I).

What are the hazards?

Moving hazards!



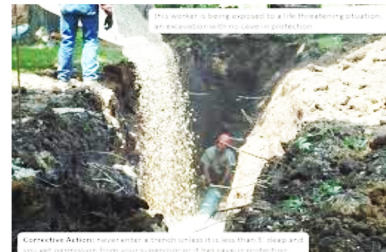
(a) Slide Page 4

Fall hazards!



(b) Slide Page 5

Cave-in hazards!



(c) Slide Page 6

Electrocution hazards!



(d) Slide Page 7

Figure B.2: Lecture-based Training Slides(Part II).

| User ID | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 1 | None | Fall | Fall | Fall | None | None | Fall | NA | CaughtIn | None | Electric | NA |
| 2 | CaughtIn | CaughtIn | Fall | CaughtIn | Fall | CaughtIn | CaughtIn | Fall | Electric | CaughtIn | Electric | Electric |
| 3 | NA | Fall | Fall | Fall | Fall | CaughtIn | Fall | None | Fall | Fall | Fall | Electric |
| 4 | CaughtIn | NA | Fall | Fall | None | Fall | Fall | CaughtIn | CaughtIn | Fall | Fall | Electric |
| 5 | Fall | Fall | Fall | Fall | None | None | None | None | None | Fall | Fall | Electric |
| 6 | Fall | Electric | NA | Fall | Fall | Electric | Fall | Electric | Fall | CaughtIn | CaughtIn | Electric |
| 7 | CaughtIn | None | None | None | Fall | CaughtIn | CaughtIn | Fall | CaughtIn | CaughtIn | Electric | Electric |
| 8 | Fall | CaughtIn | CaughtIn | Fall | None | Fall | Fall | Fall | Fall | Fall | CaughtIn | Electric |
| 9 | None | Fall | Fall | Fall | Fall | None | None | CaughtIn | Fall | Fall | Fall | None |
| 10 | Fall | Fall | CaughtIn | Fall | Fall | NA | Electric | CaughtIn | CaughtIn | Fall | NA | Electric |
| 11 | CaughtIn | Fall | Fall | Fall | CaughtIn | None | Fall | None | CaughtIn | NA | Fall | None |
| 12 | None | Fall | Fall | Fall | Fall | Fall | Fall | Fall | Electric | Electric | None | None |

Table B.1: Answers for pre-evaluation(Control Group).

| User ID | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 1 | CaughtIn | Fall | Fall | Fall | Fall | Fall | Fall | NA | CaughtIn | Electric | CaughtIn | None |
| 2 | NA | Fall | Fall | Fall | None | NA | Fall | None | Electric | CaughtIn | CaughtIn | None |
| 3 | None | CaughtIn | Fall | CaughtIn | Fall | CaughtIn | CaughtIn | Fall | CaughtIn | Fall | Fall | Electric |
| 4 | None | None | Fall | Fall | Fall | Fall | Fall | None | None | NA | Fall | NA |
| 5 | Fall | Fall | Fall | Fall | None | None | Fall | Fall | Electric | Fall | Electric | Electric |
| 6 | Fall | Fall | Fall | Fall | Fall | None | CaughtIn | Fall | CaughtIn | Fall | Fall | None |
| 7 | CaughtIn | Fall | CaughtIn | None | Fall | NA | CaughtIn | Electric | None | CaughtIn | CaughtIn | Electric |
| 8 | None | Fall | CaughtIn | Fall | Fall | Electric | CaughtIn | CaughtIn | Electric | CaughtIn | CaughtIn | CaughtIn |
| 9 | None | Fall | Fall | Fall | Fall | CaughtIn | None | None | Electric | Fall | Fall | None |
| 10 | CaughtIn | CaughtIn | CaughtIn | CaughtIn | Electric | CaughtIn | CaughtIn | Fall | Electric | Fall | None | Electric |
| 11 | CaughtIn | Fall | Fall | Fall | Fall | CaughtIn | Fall | Fall | Electric | Fall | CaughtIn | Electric |
| 12 | Fall | Fall | Fall | Fall | Fall | Electric | CaughtIn | None | CaughtIn | None | Fall | CaughtIn |

Table B.2: Answers for pre-evaluation(VR Group).

B.2 Evaluation Tasks

The pre-evaluation and post-evaluation are sharing the same evaluation tasks which are synthesized on the third construction site and optimized with the parameter settings: *All Types of Hazard* as specified in the main chapter. As shown in Figure B.3, there are 12 separate hazard inspection tasks in total among which there are 5 fall hazards inspection tasks, 4 electrical hazards inspection tasks, and 3 caught-in hazards inspection tasks respectively. In order to avoid the influences from the dynamic hazards that can distract users' attention to inspecting nearby hazards, the dynamic hazards are disabled during the pre and post-evaluation.

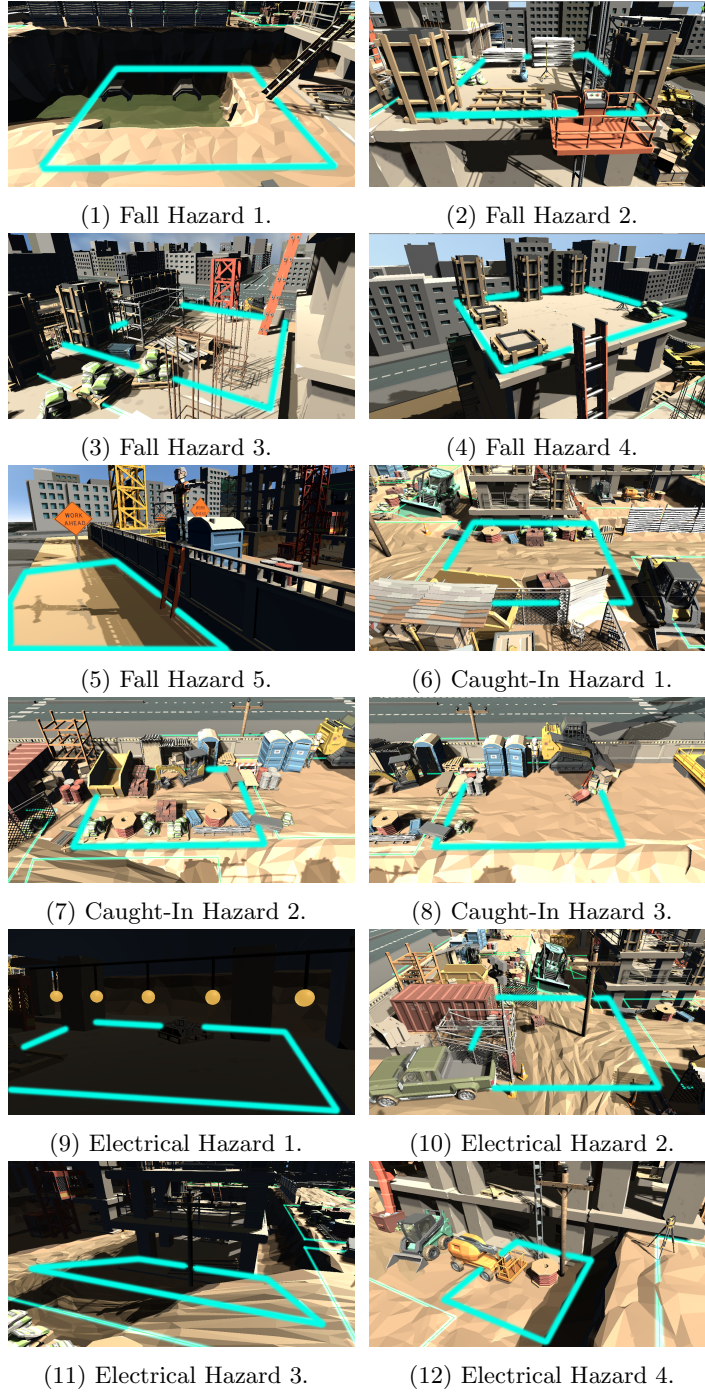


Figure B.3: The 12 evaluation tasks. Including 5 fall hazards, 3 caught-in hazards, and 4 electrical hazards. Blue boundaries highlight the hazardous object in that region. Users from both groups are suggested to pay attention to the blue boundary where they are and only inspect the hazardous objects within the same boundary.

| User ID | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
|---------|----------|------|------|----------|------|----------|----------|----------|----------|----------|----------|----------|
| 1 | Fall | Fall | Fall | Fall | NA | None | Fall | NA | None | Electric | NA | CaughtIn |
| 2 | CaughtIn | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | None | Electric | Fall | Electric | Electric |
| 3 | CaughtIn | NA | Fall | Fall | Fall | CaughtIn | CaughtIn | Fall | Fall | Fall | Fall | NA |
| 4 | None | Fall | Fall | Fall | None | None | None | None | None | Electric | Electric | Electric |
| 5 | CaughtIn | None | Fall | Fall | None | None | None | None | Electric | Electric | Electric | None |
| 6 | CaughtIn | Fall | Fall | Fall | None | None | None | CaughtIn | Electric | Electric | Electric | None |
| 7 | None | Fall | Fall | None | None | Electric | CaughtIn | None | Electric | CaughtIn | CaughtIn | CaughtIn |
| 8 | None | Fall | Fall | Fall | None | Fall | Fall | Fall | Electric | CaughtIn | None | Electric |
| 9 | Fall | Fall | Fall | Fall | Fall | None | Fall | CaughtIn | Electric | None | Electric | Electric |
| 10 | Fall | Fall | None | CaughtIn | None | Electric | None | Electric | None | Fall | Electric | Electric |
| 11 | None | Fall | Fall | Fall | None | None | None | None | CaughtIn | Fall | None | None |
| 12 | CaughtIn | Fall | Fall | Fall | None | None | None | None | Electric | Electric | None | CaughtIn |

Table B.3: Answers for post-evaluation(Control Group).

| User ID | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 1 | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric |
| 2 | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
| 3 | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
| 4 | Fall | CaughtIn | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
| 5 | None | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric |
| 6 | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
| 7 | CaughtIn | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | CaughtIn | Electric | Electric |
| 8 | Fall | Fall | CaughtIn | CaughtIn | Fall | CaughtIn | CaughtIn | CaughtIn | CaughtIn | None | CaughtIn | Electric |
| 9 | Fall | Fall | Fall | Fall | None | Fall | Fall | CaughtIn | None | Fall | Fall | Electric |
| 10 | Fall | Fall | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
| 11 | None | Fall | Fall | Fall | CaughtIn | CaughtIn | CaughtIn | CaughtIn | Electric | Electric | Electric | Electric |
| 12 | None | Fall | Fall | Fall | Fall | None | CaughtIn | CaughtIn | Fall | Fall | Electric | Electric |

Table B.4: Answers for post-evaluation(VR Group).

B.3 Evaluation Answers

Here we demonstrate the users' answers collected from the evaluations in the tables. The column names are the correct answer for that hazard type. The rows are the answers from each user. Table B.1 lists the answers for pre-evaluation collected from the users in the control group. Table B.2 lists the answers for pre-evaluation collected from the users in the VR group. Table B.3 lists the answers for post-evaluation collected from the users in the control group. Table B.4 lists the answers for post-evaluation collected from the users in the VR group. The mistakes are counted as the number of mismatches between the users' answers listed in the rows and the correct answers listed in the columns. As we can see from Table B.4, only users from the VR group can find the answers almost similar to the correct answers in post-evaluation.

B.4 Users' Feedback Details

After having the users completed the post-evaluation tasks, we collected the users' feedback through a questionnaire that offered different questions to evaluate our training programs. Here are some selected answers from the users with respect to the positive and negative feedback towards our training and evaluation programs are listed here.

Positive Feedback. (1)The graphics were very realistic and engaging. After the training, it was easier to complete the post-evaluation. (2)I really liked the training, it's fun and interesting. (3)The simulation is very good and helpful. (4)One feature I liked in the training program was the setting and the way the objects were placed because it resembles an actual construction site which is very realistic. (5)As a first-time VR user, this training was very realistic and entertaining for me. Not the ones (games) where people might sleep out of boredom. (6)The movement(navigation) of the player as well as the animations of the trucks is realistic. (7)The caught-in hazards look brilliant. (8) The selection of a hazard with a pointer(lightsaber) is innovative. (9)I like the interactions through controllers, which are engaging. (10)I like the placement of hazards, especially, there are some hazards coming from above. (11) The fall hazard was informative. (12)Multiple-choice questions are the parts I like. (13)I like the navigation and control. (14)I love the path planning in VR, which mimic reality.

Negative Feedback. (1)I think some cases consist of more than one hazard. So sometimes I feel a little confused. (2)More scenarios would be great especially if there were more populations in the site. If there were more workers, there could have been more hazardous situations similar to the real world. (3)I didn't like the not clear pathways. Sometimes, obstructions were breaking my concentration. (4)Questions need to match the zones precisely. Sometimes I was unsure about which problem is corresponding to which zones. (5)It may not be comfortable for some older people to play this 3D game. (6)The graphics are good

but they can be even more realistic. (7)When I speed up, I feel a little dizzy. (8)I feel a little dizzy when I go upstairs in the program. (9)I feel terrible when I run into VR objects, especially when I am struck by moving hazards.

B.5 Conditional Probability for Dynamic Hazard

In our animation model for construction simulation, we simulate the dynamic hazard through periodic linear motion of a vehicle or periodic rotation of an excavator's arm. Let $T_{d(B_i, H_i) < \delta(H_i)}$ denote the period time that dynamic hazard H_i 's corresponding moving object B_i (e.g., a vehicle.) enters inside H_i 's affecting range $\delta(H_i)$, let $T_{d(B_i, H_i) \geq \delta(H_i)}$ denote the period time that H_i 's moving object B_i exits outside the affecting range, then dynamic hazard H_i 's conditional probability $\Pr(H_i | d(\mathbf{r}, H_i) < \delta(H_i))$ is calculated as:

$$\Pr(H_i | d(\mathbf{r}, H_i) < \delta(H_i)) = \frac{T_{d(B_i, H_i) < \delta(H_i)}}{T_{d(B_i, H_i) < \delta(H_i)} + T_{d(B_i, H_i) \geq \delta(H_i)}}$$

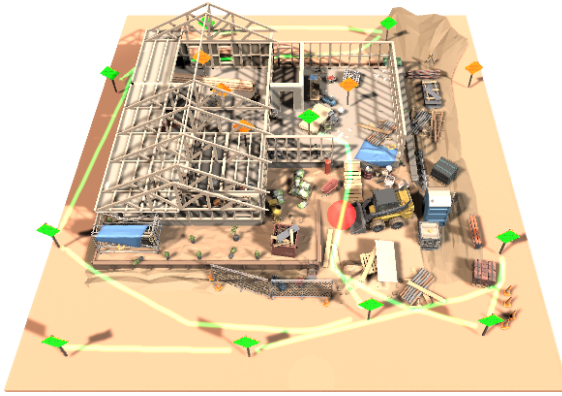
where \mathbf{r} is the position of route R 's arbitrary point. Noted that both $T_{d(B_i, H_i) < \delta(H_i)}$ and $T_{d(B_i, H_i) \geq \delta(H_i)}$ can be calculated during the simulation by a timer counting the time when dynamic hazard H_i 's moving object B_i 's bounding box collides with a bounding sphere centered at H_i 's position with radius of H_i 's affecting range $\delta(H_i)$. This mathematical formula can be extended to calculate the conditional probability for dynamic hazardous events with more complex periodic mechanical motions.

B.6 Synthesized Training Scenarios

As shown in Section *Changing Parameters*, different parameters are applied for generating different training scenarios on Construction Site 2. For those users from the VR group with a very poor background, we also synthesize different training scenarios on construction site 1 to further improve their VR training effects. Similarly, we synthesize personalized training

scenarios that emphasizing users' skills for inspecting each type of hazard by adjusting the parameter settings of the targets as below:

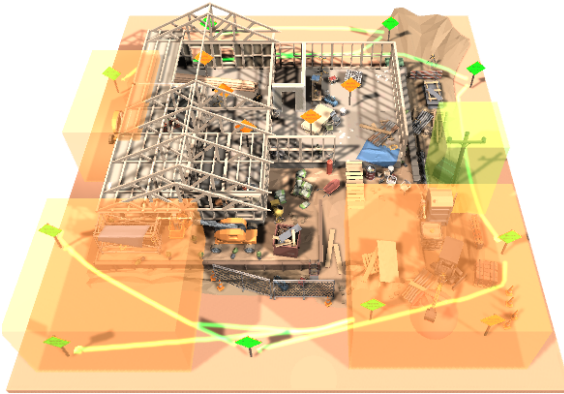
- *Fewer Hazard*: A route with fewer hazard occurred: $\lambda_F = \lambda_C = \lambda_E = \rho_D = 0.1$.
- *More Fall Hazard*: A route with more fall hazard occurred: $\lambda_F = 0.9$, $\lambda_C = \lambda_E = \rho_D = 0.1$.
- *More Caught-In Hazard*: A route with more caught-in hazard occurred: $\lambda_C = 0.9$, $\lambda_F = \lambda_E = \rho_D = 0.1$.
- *More Electrical Hazard*: A route with more electrical hazard occurred: $\lambda_E = 0.9$, $\lambda_C = \lambda_F = \rho_D = 0.1$.



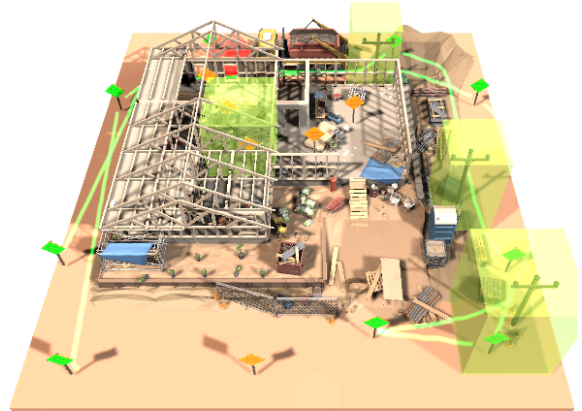
(a) Route with fewer hazards.



(b) Route with more fall hazards.



(c) Route with more caught-in hazards.



(d) Route with more electrical hazards.

Figure B.4: Personalized training scenarios synthesized on the construction site 1. Green curves are the navigation routes. The regions with fall hazards are highlighted in blue, caught-in hazards in yellow, electrical hazards in green, and the struck-by hazards in red. Here are four different training scenarios including (a) a route passing through fewer hazardous regions, this scenario is easy for the beginners to start with, and is used for the pre-train. (b) a route passing through more regions with fall hazards. (c) a route passing through more regions with caught-in hazards. (d) a route passing through more regions with electrical hazards.

Bibliography

Bibliography

- [1] 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*, vol. 23, no. 4, pp. 1275–1284, 2017.
- [2] 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)*. IEEE, 2018, pp. 297–304.
- [3] F. Mueller, M. R. Gibbs, F. Vetere, and D. Edge, “Supporting the creative game design process with exertion cards,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2014, pp. 2211–2220.
- [4] E. Graether and F. Mueller, “Joggbot: a flying robot as jogging companion,” in *CHI’12 Extended Abstracts on Human Factors in Computing Systems*, 2012, pp. 1063–1066.
- [5] Y.-S. Ye, H.-Y. Chen, and L. Chan, “Pull-ups: Enhancing suspension activities in virtual reality with body-scale kinesthetic force feedback,” in *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, 2019, pp. 791–801.
- [6] S. M. LaValle, A. Yershova, M. Katsev, and M. Antonov, “Head tracking for the oculus rift,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 187–194.
- [7] H. Huang, N.-C. Lin, L. Barrett, D. Springer, H.-C. Wang, M. Pomplun, and L.-F. Yu, “Automatic optimization of wayfinding design,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 9, pp. 2516–2530, 2017.
- [8] W. Liang, J. Liu, Y. Lang, B. Ning, and L.-F. Yu, “Functional workspace optimization via learning personal preferences from virtual experiences,” *IEEE transactions on visualization and computer graphics*, vol. 25, no. 5, pp. 1836–1845, 2019.
- [9] L. F. Yu, S. K. Yeung, C. K. Tang, D. Terzopoulos, T. F. Chan, and S. J. Osher, “Make it home: automatic optimization of furniture arrangement,” *ACM Transactions on Graphics (TOG)-Proceedings of ACM SIGGRAPH 2011*, v. 30,(4), July 2011, article no. 86, vol. 30, no. 4, 2011.
- [10] Z. Zhang, “Microsoft kinect sensor and its effect,” *IEEE multimedia*, vol. 19, no. 2, pp. 4–10, 2012.

- [11] E. Behm-Morawitz, J. Lewallen, and G. Choi, “A second chance at health: how a 3d virtual world can improve health self-efficacy for weight loss management among adults,” *Cyberpsychology, Behavior, and Social Networking*, vol. 19, no. 2, pp. 74–79, 2016.
- [12] A. E. Staiano, A. A. Abraham, and S. L. Calvert, “Adolescent exergame play for weight loss and psychosocial improvement: A controlled physical activity intervention,” *Obesity*, vol. 21, no. 3, pp. 598–601, 2013. [Online]. Available: <http://dx.doi.org/10.1002/oby.20282>
- [13] B. Xie, Y. Zhang, H. Huang, E. Ogawa, T. You, and L.-F. Yu, “Exercise intensity-driven level design,” 2018.
- [14] 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*. ACM, 2019, p. 554.
- [15] K. P. Padala, P. R. Padala, S. Y. Lensing, R. A. Dennis, M. M. Bopp, P. K. Roberson, and D. H. Sullivan, “Home-based exercise program improves balance and fear of falling in community-dwelling older adults with mild alzheimer’s disease: A pilot study,” *Journal of Alzheimer’s Disease*, vol. 59, no. 2, pp. 565–574, 2017.
- [16] S. Wüest, N. Borghese, M. Pirovano, R. Mainetti, R. van de Langenberg, and E. de Bruin, “Usability and effects of an exergame-based balance training program,” *Games for Health Journal*, vol. 3, no. 2, pp. 106–114, 2014.
- [17] E. F. Ogawa, T. You, and S. G. Leveille, “Potential benefits of exergaming for cognition and dual-task function in older adults: a systematic review,” *Journal of Aging and Physical Activity*, vol. 24, no. 2, pp. 332–336, 2016.
- [18] D. Schoene, T. Valenzuela, S. R. Lord, and E. D. de Bruin, “The effect of interactive cognitive-motor training in reducing fall risk in older people: a systematic review,” *BMC Geriatrics*, vol. 14, no. 1, p. 107, 2014.
- [19] D. S. Ebert, F. K. Musgrave, D. Peachey, K. Perlin, J. C. Hart, and S. Worley, *Texturing & modeling: a procedural approach*. Morgan Kaufmann, 2003.
- [20] K. Compton and M. Mateas, “Procedural level design for platform games.” 2006.
- [21] J. Roberts and K. Chen, “Learning-based procedural content generation,” *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 7, no. 1, pp. 88–101, 2014.
- [22] D. Dimovska, P. Jarnfelt, S. Selvig, and G. N. Yannakakis, “Towards procedural level generation for rehabilitation,” in *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*. ACM, 2010, p. 7.
- [23] P. Shi and K. Chen, “Learning constructive primitives for real-time dynamic difficulty adjustment in super mario bros,” *IEEE Transactions on Games*, vol. 10, no. 2, pp. 155–169, 2018.

- [24] M. Jennings-Teats, G. Smith, and N. Wardrip-Fruin, “Polymorph: A model for dynamic level generation,” in *Sixth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2010.
- [25] N. Shaker, G. N. Yannakakis, and J. Togelius, “Towards player-driven procedural content generation,” in *Proceedings of the 9th conference on Computing Frontiers*. ACM, 2012, pp. 237–240.
- [26] W. Li, J. Talavera, A. G. Samayoa, J.-M. Lien, and L.-F. Yu, “Automatic synthesis of virtual wheelchair training scenarios,” in *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 2020, pp. 539–547.
- [27] W. Li, B. Xie, Y. Zhang, W. Meiss, H. Huang, and L.-F. Yu, “Exertion-aware path generation,” *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 115–1, 2020.
- [28] Pants Up Easy, “U.S. Wheelchair User Statistics,” <https://www.pantsupeasy.com/u-s-wheelchair-user-statistics/>, April 2016, accessed November 24, 2019.
- [29] 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*, vol. 11, no. 12, p. e0168330, 2016.
- [30] 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*, vol. 82, no. 7, pp. 504–510, 2003.
- [31] 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*, vol. 86, no. 12, pp. 2316–2323, 2005.
- [32] 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*, vol. 23, no. 4, pp. 434–441, 2008.
- [33] 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*, vol. 85, no. 1, pp. 41–50, 2004.
- [34] 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.
- [35] 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*, vol. 31, no. 04, pp. 310–313, 1999.

- [36] U. Kühnapfel, H. K. Cakmak, and H. Maaß, “Endoscopic surgery training using virtual reality and deformable tissue simulation,” *Computers & graphics*, vol. 24, no. 5, pp. 671–682, 2000.
- [37] 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*, vol. 91, no. 2, pp. 146–150, 2004.
- [38] 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*, vol. 95, no. 9.
- [39] 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*, no. 1, 2009.
- [40] J. Cremer, J. Kearney, and Y. Papelis, “Driving simulation: challenges for vr technology,” *IEEE Computer Graphics and Applications*, vol. 16, no. 5, pp. 16–20, 1996.
- [41] 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*, vol. 18, pp. 289–296, 2016.
- [42] N. Gavish, T. Gutiérrez, S. Webel, J. Rodríguez, M. Peveri, U. Bockholt, and F. Tecthia, “Evaluating virtual reality and augmented reality training for industrial maintenance and assembly tasks,” *Interactive Learning Environments*, vol. 23, no. 6, pp. 778–798, 2015.
- [43] 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*. ACM, 2009, pp. 53–63.
- [44] W. G. Zaenglein Jr, “Shooting simulating process and training device using a virtual reality display screen,” Jun. 24 1997, uS Patent 5,641,288.
- [45] R. Querrec, C. Buche, E. Maffre, and P. Chevaillier, “Sécurévi: virtual environments for fire-fighting training,” in *5th virtual reality international conference (VRIC’03)*, 2003, pp. 169–175.
- [46] 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*, vol. 24, no. 4, pp. 1661–1670, 2018.
- [47] R. Williams and J. Dattilo, “Using wheelchair simulations to teach about inclusion,” *Schole: A Journal of Leisure Studies and Recreation Education*, vol. 20, no. 1, pp. 140–145, 2005.
- [48] 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)*. IEEE, 2015, pp. 19–21.

- [49] 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*. Springer, 2010, pp. 286–293.
- [50] 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*, 2000, pp. 9–14.
- [51] 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*, vol. 24, no. 11-12, pp. 599–606, 2002.
- [52] 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, 2016, p. 134.
- [53] M. Fisher, D. Ritchie, M. Savva, T. Funkhouser, and P. Hanrahan, "Example-based synthesis of 3d object arrangements," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 135, 2012.
- [54] C. Erikson and D. Manocha, "Simplification culling of static and dynamic scene graphs," *UNC-Chapel Hill Computer Science TR98-009*, 1998.
- [55] T. Shao, W. Xu, K. Zhou, J. Wang, D. Li, and B. Guo, "An interactive approach to semantic modeling of indoor scenes with an rgb-d camera," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 136, 2012.
- [56] X. Cui and H. Shi, "A*-based pathfinding in modern computer games," *International Journal of Computer Science and Network Security*, vol. 11, no. 1, pp. 125–130, 2011.
- [57] 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*, vol. 21, no. 6, pp. 1087–1092, 1953.
- [58] E. Aarts and J. Korst, "Simulated annealing and boltzmann machines," 1988.
- [59] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [60] R. A. DeVore and V. N. Temlyakov, "Some remarks on greedy algorithms," *Advances in computational Mathematics*, vol. 5, no. 1, pp. 173–187, 1996.
- [61] 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*, vol. 26, no. 5-6, pp. 655–662, 1990.
- [62] 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*, vol. 40, no. 12, pp. 1320–1324, 1993.

- [63] 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*, vol. 25, no. 2, pp. 290–302, 1954.
- [64] P. Moran and C. Smith, “The correlation between relatives on the supposition of mendelian inheritance,” *Transactions of the Royal Society of Edinburgh*, vol. 52, pp. 899–438, 1918.
- [65] G. Chen, G. Esch, P. Wonka, P. Müller, and E. Zhang, “Interactive procedural street modeling,” in *ACM transactions on graphics (TOG)*, vol. 27, no. 3. ACM, 2008, p. 103.
- [66] J. Beneš, A. Wilkie, and J. Křivánek, “Procedural modelling of urban road networks,” in *Computer Graphics Forum*, vol. 33, no. 6, 2014, pp. 132–142.
- [67] G. Nishida, I. Garcia-Dorado, and D. G. Aliaga, “Example-driven procedural urban roads,” in *Computer Graphics Forum*, vol. 35, no. 6, 2016, pp. 5–17.
- [68] E. Galin, A. Peytavie, N. Maréchal, and E. Guérin, “Procedural generation of roads,” in *Computer Graphics Forum*, vol. 29, no. 2, 2010, pp. 429–438.
- [69] G. Cordonnier, M.-P. Cani, B. Benes, J. Braun, and E. Galin, “Sculpting mountains: Interactive terrain modeling based on subsurface geology,” vol. 24, no. 5, pp. 1756–1769, 2018.
- [70] E. Guérin, J. Digne, E. Galin, A. Peytavie, C. Wolf, B. Benes, and B. Martinez, “Interactive example-based terrain authoring with conditional generative adversarial networks,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, p. 228, 2017.
- [71] Y. I. Parish and P. Müller, “Procedural modeling of cities,” in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 301–308.
- [72] C. A. Vanegas, I. Garcia-Dorado, D. G. Aliaga, B. Benes, and P. Waddell, “Inverse design of urban procedural models,” *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 168, 2012.
- [73] A. Emilien, U. Vimont, M.-P. Cani, P. Poulin, and B. Benes, “Worldbrush: Interactive example-based synthesis of procedural virtual worlds,” *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, p. 106, 2015.
- [74] R. M. Smelik, T. Tutenel, R. Bidarra, and B. Benes, “A survey on procedural modelling for virtual worlds,” in *Computer Graphics Forum*, vol. 33, no. 6, 2014, pp. 31–50.
- [75] E. Galin, E. Guérin, A. Peytavie, G. Cordonnier, M.-P. Cani, B. Benes, and J. Gain, “A review of digital terrain modeling,” in *Computer Graphics Forum*, vol. 38, no. 2, 2019, pp. 553–577.
- [76] E. Galin, A. Peytavie, E. Guérin, and B. Beneš, “Authoring hierarchical road networks,” in *Computer Graphics Forum*, vol. 30, no. 7, 2011, pp. 2021–2030.

- [77] L. Cardamone, G. N. Yannakakis, J. Togelius, and P. L. Lanzi, “Evolving interesting maps for a first person shooter,” in *European Conference on the Applications of Evolutionary Computation*. Springer, 2011, pp. 63–72.
- [78] D. Hooshyar, M. Yousefi, M. Wang, and H. Lim, “A data-driven procedural-content-generation approach for educational games,” *Journal of Computer Assisted Learning*, vol. 34, no. 6, pp. 731–739, 2018.
- [79] M. Hendriks, S. Meijer, J. Van Der Velden, and A. Iosup, “Procedural content generation for games: A survey,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 9, no. 1, p. 1, 2013.
- [80] ICAROS, “ICAROS - We make you fly,” <https://www.icaros.com/>, 2017, accessed 12-27-2019.
- [81] A. E. Staiano, A. A. Abraham, and S. L. Calvert, “Adolescent exergame play for weight loss and psychosocial improvement: a controlled physical activity intervention,” *Obesity*, vol. 21, no. 3, pp. 598–601, 2013.
- [82] F. Mueller, D. Edge, F. Vetere, M. R. Gibbs, S. Agamanolis, B. Bongers, and J. G. Sheridan, “Designing sports: a framework for exertion games,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2011, pp. 2651–2660.
- [83] F. Mueller and K. Isbister, “Movement-based game guidelines,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2014, pp. 2191–2200.
- [84] D. Weibel, B. Wissmath, S. Habegger, Y. Steiner, and R. Groner, “Playing online games against computer-vs. human-controlled opponents: Effects on presence, flow, and enjoyment,” *Computers in Human Behavior*, vol. 24, no. 5, pp. 2274–2291, 2008.
- [85] F.-C. Huang, K. Chen, and G. Wetzstein, “The light field stereoscope: Immersive computer graphics via factored near-eye light field displays with focus cues,” *ACM Trans. Graph.*, vol. 34, no. 4, pp. 60:1–60:12, Jul. 2015.
- [86] Z. Zhang, N. Raghuvanshi, J. Snyder, and S. Marschner, “Acoustic texture rendering for extended sources in complex scenes,” *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, p. 222, 2019.
- [87] M. Pfeiffer, S. Schneegass, F. Alt, and M. Rohs, “Let me grab this: a comparison of ems and vibration for haptic feedback in free-hand interaction,” in *Proceedings of the 5th augmented human international conference*. ACM, 2014, p. 48.
- [88] N. C. Nilsson, S. Serafin, and R. Nordahl, “Establishing the range of perceptually natural visual walking speeds for virtual walking-in-place locomotion,” vol. 20, no. 4, pp. 569–578, 2014.
- [89] M. Sra, S. Garrido-Jurado, C. Schmandt, and P. Maes, “Procedurally generated virtual reality from 3d reconstructed physical space,” in *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*. ACM, 2016, pp. 191–200.

- [90] M. Löchtefeld, A. Krüger, and H. Gellersen, “Deceptibike: Assessing the perception of speed deception in a virtual reality training bike system,” in *Proceedings of the 9th Nordic Conference on Human-Computer Interaction*. ACM, 2016, p. 40.
- [91] R. Herpers, W. Heiden, M. Kutz, D. Scherfgen, U. Hartmann, J. Bongartz, and O. Schulzyk, “Fivis bicycle simulator: an immersive game platform for physical activities,” in *Proceedings of the 2008 Conference on Future Play: Research, Play, Share*. ACM, 2008, pp. 244–247.
- [92] R. Rakhmatov, A. Abdulali, W. Hassan, M. Kim, and S. Jeon, “Virtual reality bicycle with data-driven vibrotactile responses from road surface textures,” in *2018 IEEE Games, Entertainment, Media Conference (GEM)*. IEEE, 2018, pp. 1–9.
- [93] D. J. Wilkie, J. Van den Berg, M. Lin, and D. Manocha, “Self-aware traffic route planning,” in *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.
- [94] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, “Anytime motion planning using the rrt,” in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1478–1483.
- [95] A. Best, S. Narang, D. Barber, and D. Manocha, “Autonovi: Autonomous vehicle planning with dynamic maneuvers and traffic constraints,” in *IROS*. IEEE, 2017, pp. 2629–2636.
- [96] H. Huang, N.-C. Lin, L. Barrett, D. Springer, H.-C. Wang, M. Pomplun, and L.-F. Yu, “Automatic optimization of wayfinding design,” 2017.
- [97] N. Sturtevant and M. Buro, “Partial pathfinding using map abstraction and refinement,” in *AAAI*, vol. 5, 2005, pp. 1392–1397.
- [98] Z. A. Algfoor, M. S. Sunar, and H. Kolivand, “A comprehensive study on pathfinding techniques for robotics and video games,” *International Journal of Computer Games Technology*, vol. 2015, p. 7, 2015.
- [99] D. González, J. Pérez, V. Milanés, and F. Nashashibi, “A review of motion planning techniques for automated vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1135–1145, 2015.
- [100] G. Tinsley, “Cardio vs weight lifting: Which is better for weight loss?” <https://www.healthline.com/nutrition/cardio-vs-weights-for-weight-loss>, 2017, accessed 10-24-2017.
- [101] S. C. Walpole, D. Prieto-Merino, P. Edwards, J. Cleland, G. Stevens, and I. Roberts, “The weight of nations: an estimation of adult human biomass,” *BMC public health*, vol. 12, no. 1, p. 439, 2012.
- [102] M. d. Neef, “Gradients and cycling: an introduction,” <http://theclimbingcyclist.com/gradients-and-cycling-an-introduction/>, 2013, accessed 1-19-2020.
- [103] S. Chib and E. Greenberg, “Understanding the metropolis-hastings algorithm,” *The American Statistician*, vol. 49, no. 4, pp. 327–335, 1995.

- [104] M. Al Borno, M. De Lasa, and A. Hertzmann, "Trajectory optimization for full-body movements with complex contacts," *IEEE transactions on visualization and computer graphics*, vol. 19, no. 8, pp. 1405–1414, 2012.
- [105] K. Naderi, J. Rajamäki, and P. Hämmäläinen, "Discovering and synthesizing humanoid climbing movements," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–11, 2017.
- [106] A. Hayes, *Building Bridges and Roads: Civil Engineers*, ser. Engineers Rule! PowerKids Press, 2015.
- [107] R. A. Douglas, *Low-volume road engineering: design, construction, and maintenance*. CRC Press, 2016.
- [108] B. McTague and G. Jergeas, *Productivity improvements on Alberta major construction projects: Phase I-Back to basics*. Alberta economic development, 2002.
- [109] A. S. Hanna, C. S. Taylor, and K. T. Sullivan, "Impact of extended overtime on construction labor productivity," *Journal of construction engineering and management*, vol. 131, no. 6, pp. 734–739, 2005.
- [110] Y. Wang, P. M. Goodrum, C. T. Haas, and R. W. Glover, "Craft training issues in american industrial and commercial construction," *Journal of construction engineering and management*, vol. 134, no. 10, pp. 795–803, 2008.
- [111] H. Karimi, T. R. Taylor, and P. M. Goodrum, "Analysis of the impact of craft labour availability on north american construction project productivity and schedule performance," *Construction Management and Economics*, vol. 35, no. 6, pp. 368–380, 2017.
- [112] A. Akintoye, "Analysis of factors influencing project cost estimating practice," *Construction Management & Economics*, vol. 18, no. 1, pp. 77–89, 2000.
- [113] M. Chester and C. Hendrickson, "Cost impacts, scheduling impacts, and the claims process during construction," *Journal of construction engineering and management*, vol. 131, no. 1, pp. 102–107, 2005.
- [114] B.-G. Hwang, S. R. Thomas, C. T. Haas, and C. H. Caldas, "Measuring the impact of rework on construction cost performance," *Journal of construction engineering and management*, vol. 135, no. 3, pp. 187–198, 2009.
- [115] S. Hasanzadeh, B. Esmaeili, and M. D. Dodd, "Measuring the impacts of safety knowledge on construction workers' attentional allocation and hazard detection using remote eye-tracking technology," *Journal of management in engineering*, vol. 33, no. 5, p. 04017024, 2017.
- [116] F. D. Rose, E. A. Attree, B. M. Brooks, D. M. Parslow, and P. R. Penn, "Training in virtual environments: transfer to real world tasks and equivalence to real task training," *Ergonomics*, vol. 43, no. 4, pp. 494–511, 2000.

- [117] F. Mantovani and G. Castelnuovo, "The sense of presence in virtual training: enhancing skills acquisition and transfer of knowledge through learning experience in virtual environments," 2003.
- [118] F. Lin, L. Ye, V. G. Duffy, and C.-J. Su, "Developing virtual environments for industrial training," *Information Sciences*, vol. 140, no. 1-2, pp. 153–170, 2002.
- [119] W. Li, J. Talavera, A. G. Samayoa, J.-M. Lien, and L.-F. Yu, "Automatic synthesis of virtual wheelchair training scenarios," in *IEEE Virtual Reality*, 2020.
- [120] R. Sacks, A. Perlman, and R. Barak, "Construction safety training using immersive virtual reality," *Construction Management and Economics*, vol. 31, no. 9, pp. 1005–1017, 2013.
- [121] K. Aati, D. Chang, P. Edara, and C. Sun, "Immersive work zone inspection training using virtual reality," *Transportation Research Record*, p. 0361198120953146, 2020.
- [122] R. Eiris, M. Gheisari, and B. Esmaeili, "Desktop-based safety training using 360-degree panorama and static virtual reality techniques: A comparative experimental study," *Automation in Construction*, vol. 109, p. 102969, 2020.
- [123] R. E. Pereira, M. Gheisari, and B. Esmaeili, "Using panoramic augmented reality to develop a virtual safety training environment," *Constr. Res. Congr.*, pp. 29–39, 2018.
- [124] H. F. Moore, R. Eiris, M. Gheisari, and B. Esmaeili, "Hazard identification training using 360-degree panorama vs. virtual reality techniques: A pilot study," in *Computing in Civil Engineering 2019: Visualization, Information Modeling, and Simulation*. American Society of Civil Engineers Reston, VA, 2019, pp. 55–62.
- [125] S. Bhoir and B. Esmaeili, "State-of-the-art review of virtual reality environment applications in construction safety," in *AEI 2015*, 2015, pp. 457–468.
- [126] H. Li, M. Lu, S.-C. Hsu, M. Gray, and T. Huang, "Proactive behavior-based safety management for construction safety improvement," *Safety science*, vol. 75, pp. 107–117, 2015.
- [127] V. Benjaoran and S. Bhokha, "An integrated safety management with construction management using 4d cad model," *Safety Science*, vol. 48, no. 3, pp. 395–403, 2010.
- [128] D. Fang, X. Huang, and J. Hinze, "Benchmarking studies on construction safety management in china," *Journal of construction engineering and management*, vol. 130, no. 3, pp. 424–432, 2004.
- [129] M. A. Froehlich and S. Azhar, "Investigating virtual reality headset applications in construction," in *Proceedings of the 52nd Associated Schools of Construction Annual International Conference*, 2016, pp. 13–16.
- [130] Q. T. Le, A. Pedro, and C. S. Park, "A social virtual reality based construction safety education system for experiential learning," *Journal of Intelligent & Robotic Systems*, vol. 79, no. 3-4, pp. 487–506, 2015.

- [131] H. Guo, H. Li, and V. Li, "Vp-based safety management in large-scale construction projects: A conceptual framework," *Automation in Construction*, vol. 34, pp. 16–24, 2013.
- [132] H. Li, G. Chan, and M. Skitmore, "Multiuser virtual safety training system for tower crane dismantlement," *Journal of Computing in Civil Engineering*, vol. 26, no. 5, pp. 638–647, 2012.
- [133] M. Gheisari and B. Esmaili, "Pars: Using augmented panoramas of reality for construction safety training," *Silver Spring, MD: CPWR-The Center for Construction Research and Training*, 2019.
- [134] D. Zhao and J. Lucas, "Virtual reality simulation for construction safety promotion," *International journal of injury control and safety promotion*, vol. 22, no. 1, pp. 57–67, 2015.
- [135] H. Xie, E. Tudoreanu, and W. Shi, "Development of a virtual reality safety-training system for construction workers," *Digital library of construction informatics and information technology in civil engineering and construction*, 2006.
- [136] U. D. of Labor, "Top four construction hazards," <https://www.osha.gov/Publications/3216-6N-06-english-06-27-2007.html#:~:text=Top%20Four%20Construction%20Hazards&text=The%20top%20four%20causes%20of,%2DIn%2FBetween%20and%20Electrocutions.&text=Wear%20and%20use%20personal%20fall,Install%20and%20maintain%20perimeter%20protection>.
- [137] G. Quenell, "Spectral diameter estimates for k-regular graphs," *Advances in mathematics*, vol. 106, no. 1, pp. 122–148, 1994.
- [138] A. E. Brouwer and W. H. Haemers, "Distance-regular graphs," in *Spectra of Graphs*. Springer, 2012, pp. 177–185.
- [139] C. J. Geyer, "Practical markov chain monte carlo," *Statistical science*, pp. 473–483, 1992.
- [140] G. Casella, *Statistical design*. Springer Science & Business Media, 2008.
- [141] G. Mantovani, L. Gamberini, M. Martinelli, and D. Varotto, "Exploring the suitability of virtual environments for safety training: Signals, norms and ambiguity in a simulated emergency escape," *Cognition, Technology & Work*, vol. 3, no. 1, pp. 33–41, 2001.
- [142] S. An, Y. Kim, G. Jung, H. Jang, C. Song, and B. Ma, "Development of chemical incident response training program by applying virtual reality technology," in *Proceedings of the 2019 3rd International Conference on Virtual and Augmented Reality Simulations*, 2019, pp. 6–10.
- [143] A. C. A. Mól, C. A. F. Jorge, and P. M. Couto, "Using a game engine for vr simulations in evacuation planning," *IEEE computer graphics and applications*, vol. 28, no. 3, pp. 6–12, 2008.

- [144] S. Sharma, S. Jerripathula, S. Mackey, and O. Soumare, “Immersive virtual reality environment of a subway evacuation on a cloud for disaster preparedness and response training,” in *IEEE symposium on computational intelligence for human-like intelligence*. IEEE, 2014, pp. 1–6.
- [145] S. Sharma, S. P. Rajeev, and P. Devearux, “An immersive collaborative virtual environment of a university campus for performing virtual campus evacuation drills and tours for campus safety,” in *International Conference on Collaboration Technologies and Systems*. IEEE, 2015, pp. 84–89.
- [146] S. Sharma, I. Amo-Fempong, D. Scribner, J. Grynovicki, and P. Grazaitis, “Collaborative virtual reality environment for a real-time emergency evacuation of a nightclub disaster,” *Electronic Imaging*, vol. 2019, no. 2, pp. 181–1, 2019.
- [147] L. Chittaro and F. Buttussi, “Assessing knowledge retention of an immersive serious game vs. a traditional education method in aviation safety,” *IEEE transactions on visualization and computer graphics*, vol. 21, no. 4, pp. 529–538, 2015.
- [148] P. Wang, P. Wu, J. Wang, H.-L. Chi, and X. Wang, “A critical review of the use of virtual reality in construction engineering education and training,” *International journal of environmental research and public health*, vol. 15, no. 6, p. 1204, 2018.
- [149] D. C. Schwebel, T. Combs, D. Rodriguez, J. Severson, and V. Sisiopiku, “Community-based pedestrian safety training in virtual reality: A pragmatic trial,” *Accident Analysis & Prevention*, vol. 86, pp. 9–15, 2016.
- [150] P. Caserman, M. Cornel, M. Dieter, and S. Göbel, “A concept of a training environment for police using vr game technology,” in *Joint International Conference on Serious Games*. Springer, 2018, pp. 175–181.
- [151] A. Conway, J. I. James, and P. Gladyshev, “Development and initial user evaluation of a virtual crime scene simulator including digital evidence,” in *International Conference on Digital Forensics and Cyber Crime*. Springer, 2015, pp. 16–26.
- [152] S. Sharma, S. T. Bodempudi, D. Scribner, and P. Grazaitis, “Active shooter response training environment for a building evacuation in a collaborative virtual environment,” *Electronic Imaging, 3D Measurement and Data Processing 2019*, 2020.
- [153] H. Hnaidi, E. Guérin, S. Akkouche, A. Peytavie, and E. Galin, “Feature based terrain generation using diffusion equation,” in *Computer Graphics Forum*, vol. 29, no. 7. Wiley Online Library, 2010, pp. 2179–2186.
- [154] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial networks,” *arXiv preprint arXiv:1406.2661*, 2014.
- [155] A. Peytavie, T. Dupont, E. Guérin, Y. Cortial, B. Benes, J. Gain, and E. Galin, “Procedural riverscapes,” in *Computer Graphics Forum*, vol. 38, no. 7. Wiley Online Library, 2019, pp. 35–46.

- [156] B. M. Lee and R. W. Deitrick, "Cardiorespiratory responses to versaclimber and treadmill exercise testing in healthy individuals," *Journal of Cardiopulmonary Rehabilitation and Prevention*, vol. 14, no. 3, pp. 166–172, 1994.
- [157] D. M. De Carli, C. T. Pozzer, F. Bevilacqua, and V. Schetinger, "Procedural generation of 3d canyons," in *2014 27th SIBGRAPI Conference on Graphics, Patterns and Images*. IEEE, 2014, pp. 103–110.
- [158] D. Kendzierski and K. J. DeCarlo, "Physical activity enjoyment scale: Two validation studies," *Journal of sport and exercise psychology*, vol. 13, no. 1, pp. 50–64, 1991.
- [159] L. E. Graves, N. D. Ridgers, K. Williams, G. Stratton, G. Atkinson, and N. T. Cable, "The physiological cost and enjoyment of wii fit in adolescents, young adults, and older adults," *Journal of Physical Activity and Health*, vol. 7, no. 3, pp. 393–401, 2010.

CURRICULUM VITAE

Wanwan Li has received an MS. (Master of Science) in Computer Science major from the University of Central Florida(UCF) in 2016 and a BE. (Bachelor of Engineer) in computer science and technology major from the Harbin Institute of Technology(HIT), Harbin, China in 2014.