ReWalking Project

Redirected Walking Toolkit Demo

Advisor: Miri Ben-Chen
Students: Maya Fleischer, Vasily Vitchevsky

Introduction
- Equipment

Project Description
- Curvature change
- Translation change
- Rotation change
- ZigZag

Challenges
- Equipment
- Unity
- Vive
- Random movements error
- Redirected Walking Algorithms
  - Curvature change
  - Translation change
  - Rotation change
  - ZigZag

References

Introduction

Redirected Walking (ReWalking) is a name for methods designed to give the user an illusion that he moves in a large virtual world, while staying in a room of a limited size. The need for these methods arose from the fact that in practice it's not possible to allocate a room large enough to encapsulate such virtual worlds (mainly due to economic cost). Moreover, such techniques will provide an immersive virtual reality experience, allowing to build much more enjoyable systems.

The techniques we used are based on the following papers:

- Redirected Walking [1]
- Estimation of Detection Thresholds for Redirected Walking Techniques [2]

Equipment

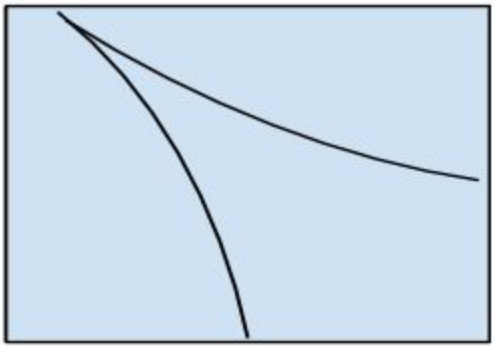
The System includes an HTC Vive VR headset, with two base stations (position tracking sensors) placed in two opposite corners of a room.
The HTC Vive is connected to a regular PC, which runs the Unity Project that contains our software.

Project Description

In this project we built a demo of the various methods for ReWalking based on the results of the study [2]. We created a virtual world, where the following techniques are demonstrated.

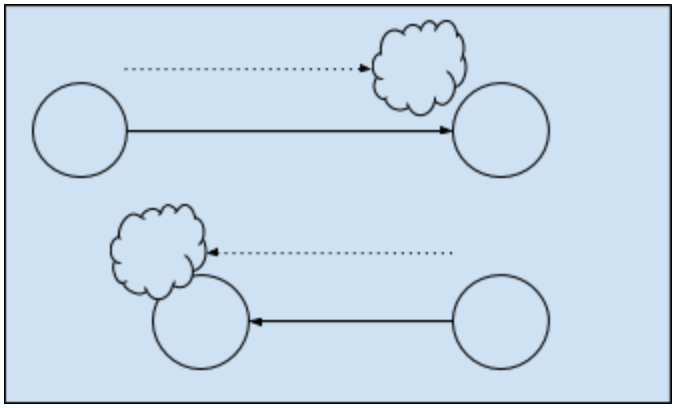
Curvature change

We induced curvature to the track, such that the virtual straight line is a circular path in reality. This allows the user to walk down an infinite hall in a virtual world while staying in a limited place in the physical world where he walks around a circle.
In this method, the first person view camera is rotated as the user walks, making the user subconsciously correct for this error. In result, the user feels like he walks straight, while in reality he walks on an arc. According to the papers [1, 2], users usually can not discern whether they were going straight or on an arc, when the radius of the circle is 22 meters or more.
Our demonstration is more space limited due to the HTC Vive only supporting rooms up to 5x5m.
In our demo, we let the user walk along a corridor and back, while in reality he will walk on two arcs as follows: (also see video demos: short and long)



Translation change

We Changed the length of the track, increasing or decreasing the user's walking distance in the virtual world compared to the actual distance he passes in the physical world. According to the papers [1, 2], usually users will not notice a change if a distance of 5 meters in the virtual world, is translated into a physical distance of between 4.3 to 6.3 meters.
In order to experience the distance in the best way, we created the system so the translation will be increased while walking forward, and will be decreased on the way back. This way, the user can walk forward for a few meters, turn around and return to the same virtual reality position. While the user thinks that in the virtual worked he returned to the same position, he can then take off the glasses and see that he actually did not returned to the starting physical point.

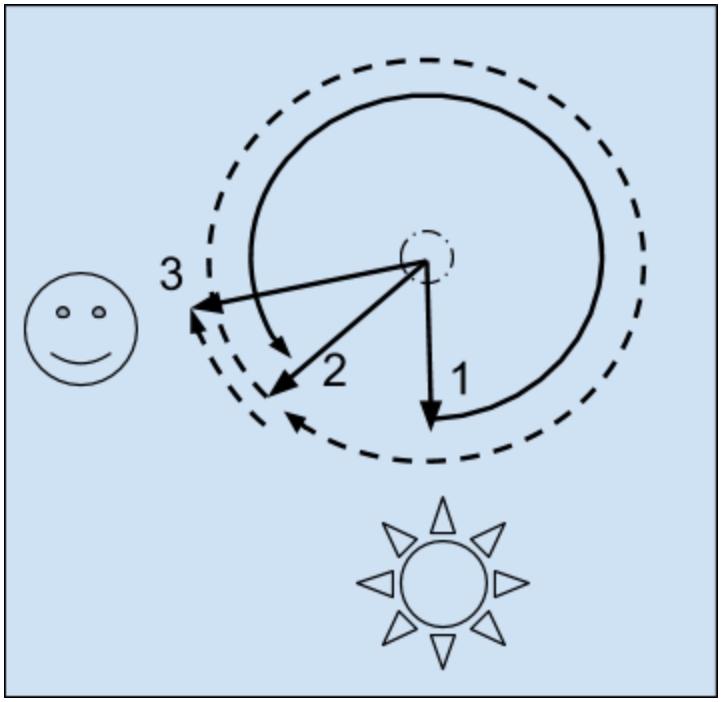


Rotation change

We resized the angle of rotation, so the user is not aware that he turned in a different angle than it seems in the virtual world. For example, the user would feel that he turned in 270 degrees, when in fact he turned 180 degrees. According to the papers [1, 2], users usually have no ability to distinguish distortion when for rotating 90 degrees in the virtual world, they have to actually rotate between 72.6 degrees to 134.3 degrees in the real world.
We created a system where CCW rotations are increased and CW rotations are decreased. This way, the user can rotate in one direction and rotate back, so to get back to the starting

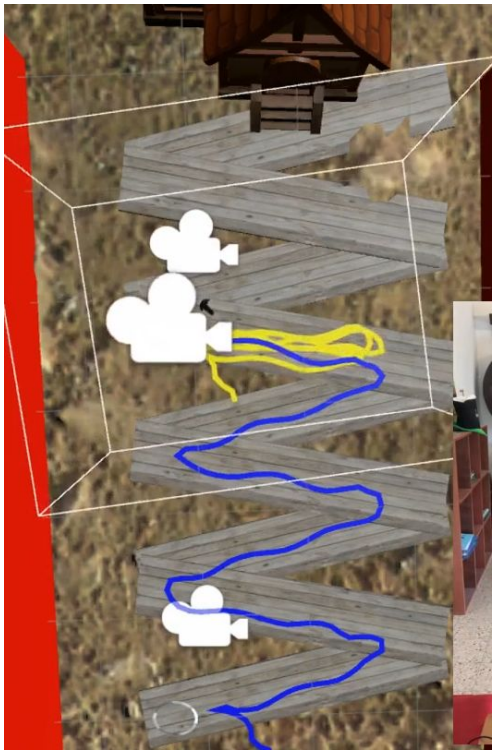
heading in the virtual world, and see that in the physical world he is not facing the starting heading.

In the system that we built, we gave the user the possibility to choose between the above redirecting methods. Once the user chose the redirection method he wishes to try, he can walk freely in the virtual world to experience the impact of the redirection.



ZigZag

From our observations, the most surprising demo for the users.
None of the above methods fully demonstrate walking in a bigger room than the user is placed in. With the ZigZag path we demonstrate walking infinitely far. The user thinks he is walking towards a house located in the end of a wooden path, far larger than the room dimensions, making sharp turns along the way. However, he actually walks along the same path back and forth.



Challenges

Equipment

When we started the project, we read previous papers and planned how we were going to build a working demo. The papers used proprietary VR and position tracking systems with a cost of tens of thousands of USD, a budget we did not have. We had no experience with consumer VR headsets, and did not really know if this was going to work, but eventually the HTC Vive looked promising - it had sub-millimeter, low latency position tracking, and it worked in relatively large rooms. And yet, for a true infinitely straight virtual path, one has to work in a room as large as 20 meters in diameter, which current consumer products don't offer.

Unity

Learning Unity from zero was not easy. We downloaded many worlds to learn from, understanding the various assets, to create a world suitable for our demos. We also learned how to use the Vive controller in unity.
Using camera movements was very confusing and tricky. Moving the camera according to the headset movement took time to get right, with every demo having a different algorithm for the camera movement.
We had to craft two special scenes/paths where the user can experience the demos.

Vive

This was the first time we used the HTC Vive headset, and it proved to be very user friendly! No challenges here :)
Using the Vive controller in Unity was also relatively easy with the SteamVR asset.

Random movements error

Depending on the redirection technique, the camera movement is always added, regardless of movement direction. In such cases, small random movements accumulate and can introduce a drift which the user may notice. Therefore, we have a threshold to determine if the user is moving.

Redirected walking methods

For every redirecting method, we found a way to demonstrate it in a surprising way, such that after the user takes off his headset, he easily notices he was redirected.

Curvature change

A full demo of infinitely long path requires a room of about 20 meters in diameter. Instead, we let the user walk along a corridor back and forth, after which he returns to a different location.

Translation change

To experience this method, the walking distance is increased when the user is walking in one direction, but decreased in the other.

Rotation change

Like in the translation demo, in this demo we increase the turn in one direction and decrease in the other.

ZigZag

With this method we demonstrated walking infinitely far, along a ZigZag path. The user thinks he is walking a big distance, while he actually walks along the same way back and forth.

References

[1] Sharif Razzaque, Zachariah Kohn, Mary C. Whitton.
Redirected Walking.

[2] Frank Steinicke, Gerd Bruder, Jason Jerald, Harald Frenz, and Markus Lappe.
Estimation of Detection Thresholds for Redirected Walking Techniques.