VR Tennis Training Game

Iris Zhao, Jaylyn Zhang, Yiqing Wang
University of Washington

Problem

This project presents a VR Tennis Training Game on Meta Quest 2, designed to provide users with a realistic tennis experience. Motivated by the potential of VR to enhance accessibility to sports training, our work focuses on the modeling of ball physics and racket interactions, carefully tuned to replicate authentic tennis gameplay dynamics. The results demonstrate a compelling virtual tennis experience with accurate ball trajectory and responsive gameplay.

Related Work

One of the most relevant commercial VR tennis games is Tennis League VR, available on the Meta Quest 2. Tennis League VR offers a variety of features, including multiplayer capabilities, customizable avatars, and different game modes.

Solution

Through our development process, we incorporated physics formulas, fine-tuned constants, made use of the existing VR technologies (Meta Quest2) and used Unity components to achieve a realistic and immersive VR tennis game experience.

References

[1] FPS Full Game Tutorial
[2] How to fix Pink Materials in Unity
[3] Oculus Quest2 set up guide

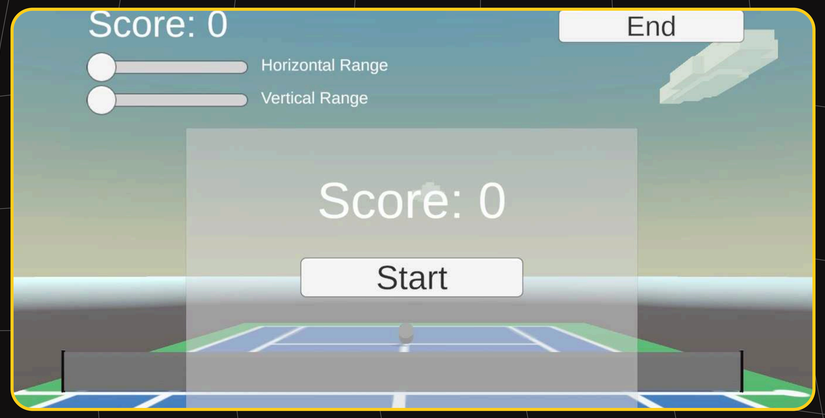

Figure 1: Starting page

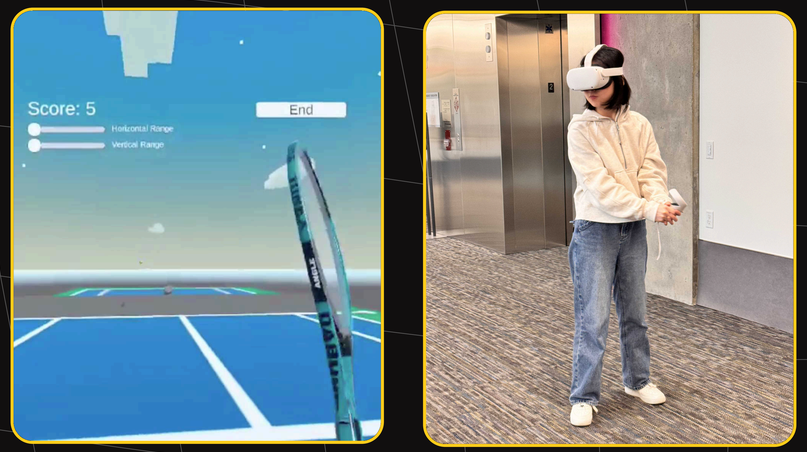

Figure 2: During the game

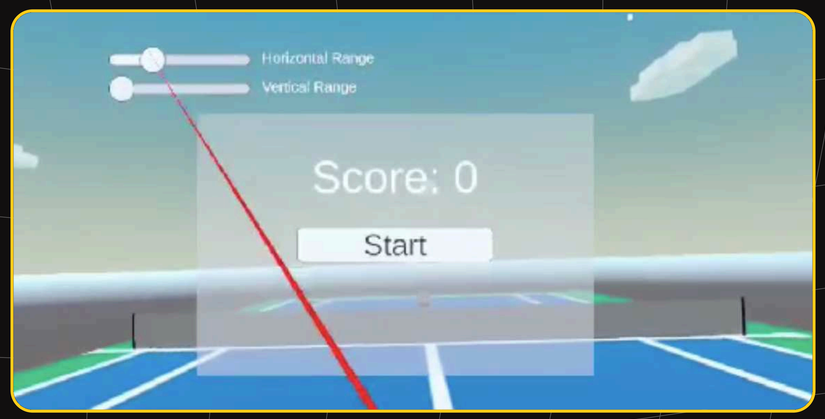

Figure 3: Adjusting settings

Method

1. Collision Detection
   We implemented a ray-casting solution. This technique projects an infinite line along the ball's velocity vector. The ray travels with the ball and detects intersections with any collider in its path.
2. Ball-Ground Bounce Behavior
   We simulated the bouncing effect of the ball from the ground by implementing reflection based on surface normals. We also applied a velocity reduction factor of 0.8 to simulate friction and energy loss during bounces, which creates the most natural result.
3. Ball-Racket Bounce Behavior
   We decomposed the post-collision ball velocity calculation into two components -- the reflected velocity and the velocity transfer from the racket to the ball during contact. Through testing, we found that an equal weighting of these two components produced the most natural feel. We also computed the spin axis and the spin amount based on the velocity of the ball at the point of impact to simulate the ball spin during hitting.

Results

Our collision detection between the ball and the racket had about a 90% success rate using ray casting. The 0.8 velocity reduction factor created realistic bounce patterns, but spin effect was not very prominent. The user feedback we got was mostly positive, though some found the game too difficult without tennis experience.