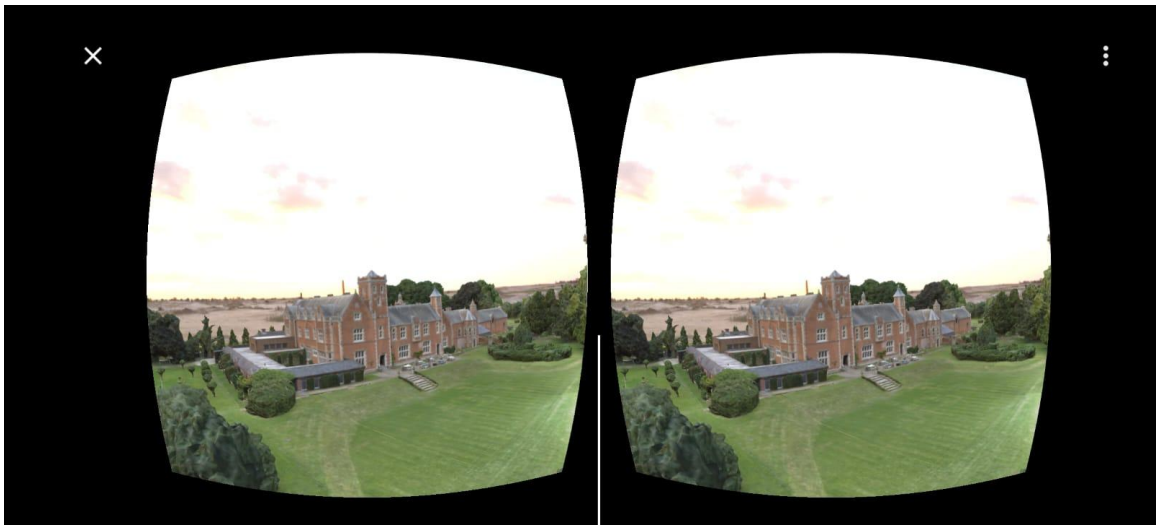
VR Autowalk Project



📖 Project Overview

The VR Autowalk Project is an interactive 3D web experience built using Three.js, WebXR, and GLTF models. This project enables two modes of navigation: traditional desktop controls using keyboard and pointer lock, and an innovative gaze-based movement system for mobile VR (e.g., Google Cardboard). Users on mobile VR can simply look in a direction for a set amount of time, and the system will automatically move them forward in that direction.

✨ Features

- Desktop navigation using WASD, Arrow Keys, and Pointer Lock.
- Mobile VR navigation using gaze-based movement — look to move.
- Gaze ring visual feedback to indicate gaze progress before movement.
- HDR environment setup for realistic lighting.
- GLB model loading and rendering in Three.js.
- Support for both VR and non-VR users seamlessly.

Technical Setup

This project is powered by Three.js for rendering, WebXR for immersive VR functionality, and GLTFLoader for loading 3D models. The implementation includes a VR rig group to handle smooth positional updates for both VR and desktop users.

Installation & Usage

1. Clone or download this repository to your local machine.
2. Serve the project files using a local development server (e.g., `npx serve`, `python3 -m http.server`, or similar).
3. Open the project URL in a supported browser:
   • Desktop: Use Chrome or Firefox with WebXR support.
   • Mobile: Use Chrome for Android, then place your phone in a VR headset (Google Cardboard or similar).
4. For VR: Tap 'Enter VR' and begin exploring by looking in a direction.
5. For desktop: Click to lock pointer and use WASD or Arrow Keys to move.

Controls

Desktop:
• W / Arrow Up → Move Forward
• S / Arrow Down → Move Backward
• A / Arrow Left → Strafe Left
• D / Arrow Right → Strafe Right
• E / Page Up → Move Up
• Q / Page Down → Move Down

Mobile VR:
• Gaze at a point for 2 seconds → Move automatically 10 meters in that direction

Code Highlights

• **Three.js Integration:** Sets up a 3D scene, camera, lighting, and renderer.
• **VR Rig:** Camera is wrapped in a Group object (rig) to manage movement consistently in VR and desktop modes.
• **PointerLockControls:** Allows free camera look and movement for desktop users.
• **Gaze Ring:** A 3D ring mesh that scales up to visually indicate gaze progress before initiating movement.
• **Auto-Move Logic:** Moves user exactly 10 meters in the direction of gaze once the gaze threshold is reached.

Requirements

- Node.js or Python (for running a simple server)
- Modern browser with WebXR support (Chrome recommended)
- Three.js (latest build)
- A GLB 3D model file (scene-optimized.glb)
- An HDR file for the environment (env.hdr)

Future Improvements

- Add more advanced interactions (e.g., gaze-triggered UI elements).
- Implement speed customization and extended movement options.
- Add audio/haptic feedback for gaze-triggered movement.
- Support multiplayer WebXR sessions.

Conclusion

The VR Autowalk Project is a foundation for immersive WebXR navigation using Three.js. It demonstrates how to blend traditional desktop control schemes with modern VR gaze-based interactions, paving the way for intuitive, hands-free VR experiences.