

Führerschein AR

SoSe 2024
Bei Prof. Jung

Vu Minh Le, 563226
Cong Nguyen-Dinh, 567103

Inhaltsverzeichnis

Inhaltsverzeichnis	2
Idee	3
Fahrzeug-platzierung mit Image Targets	3
Schilderkennung	3
Schildplatzierung	4
Probleme	4
Libraries, Plugins, Assets und Quellen	4

Idee

Unsere Idee war es, eine App zu entwickeln, die Fahrschülern und Fahrlehrern mithilfe von Augmented Reality dabei hilft, Verkehrsschilder zu lernen und deren Bedeutung zu verstehen. Durch die Platzierung von Schildern und Fahrzeugen in der AR-Umgebung kann der Fahrlehrer seinen Schülern verschiedene Verkehrsszenarien demonstrieren. Unsere benutzerfreundliche Oberfläche ist dabei sowohl für Schüler als auch für Lehrer intuitiv gestaltet. Zusätzlich können Fahrschüler unterwegs mit ihrem iPhone Verkehrsschilder scannen und erhalten dabei sofortige Erklärungen zu den jeweiligen Schildern. Diese Funktion ermöglicht ein effektives Lernen auch außerhalb des Unterrichts.

Fahrzeug-platzierung mit Image Targets

Die Fahrzeugplatzierung in unserer App erfolgt mittels Image Targets. Wir haben kostenlose Automodelle von Sketchfab verwendet, die anhand verschiedener Geldscheine aktiviert werden. Durch das Scannen eines bestimmten Geldscheins erscheint ein entsprechendes Automodell:

- 100€ - Porsche¹
- 50€ - Audi²,
- 20€ BMW³,
- 10€ Chervolet⁴

Diese Funktion ermöglicht es, unterschiedliche und komplexe Verkehrsszenarien zu demonstrieren, indem verschiedene Fahrzeuge in die AR-Umgebung eingebunden werden.

Schilderkennung

Ein weiteres wichtiges Feature unserer App ist die Schilderkennung. Für die Implementierung nutzten wir den XR Image Manager und erstellten Prefabs für die verschiedenen Verkehrsschilder. Die Bilder der Schilder erhielten wir von der Bußgeldstelle. Anfänglich hatten wir zu viele Prefabs und Bilder, was die App erheblich verlangsamte. Nach der Optimierung erkennt das System ein Schild und zeigt die zugehörige Beschreibung an. Allerdings ist die Schilderkennung manchmal fehlerhaft, besonders bei Schildern, die sich sehr ähnlich sehen.

¹

<https://sketchfab.com/3d-models/modified-porsche-911-carrera-4s-9f3d4ddc3eb340b8820251dbe3e059b8>

² <https://sketchfab.com/3d-models/audi-e-tron-7d5e2f11dad943cf86f493faf5b7896a>

³ <https://sketchfab.com/3d-models/bmw-m4-g82-2022-d1f7bb2b9115479b9a4e5be68279d379>

⁴ <https://sketchfab.com/3d-models/chevrolet-spark-3833436502494906a2ea9c3975f4de94>

Schildplatzierung

Ein weiteres Feature unserer App ist die Schildplatzierung. Hierbei wurden die Schilder durch ein eigens geschriebenes Skript mithilfe des AR Raycast Managers platziert. Die Modelle stammen aus dem "Road Sign - Big Pack" von Mixall, das wir kostenlos aus dem Unity Asset Store bezogen haben. Zwar handelt es sich bei den Schildern um amerikanische Verkehrsschilder, doch der kostenlose Zugang zu diesen Assets war für uns entscheidend.

Probleme

Während der Entwicklung unserer Führerschein-AR-App standen wir vor mehreren Herausforderungen. Ein zentrales Ziel war es, die App plattformübergreifend verfügbar zu machen, doch dies gestaltete sich schwierig, insbesondere mit Android und ARCore. Zudem hatten wir mit Versionsproblemen in Unity zu kämpfen, was die Kompatibilität und Stabilität beeinträchtigte. Beim Erstellen der App traten auch hardwarebedingte Schwierigkeiten auf, da unsere Teammitglieder sowohl MacBooks mit M2-Chip als auch mit Intel-Prozessoren nutzten. Ein weiteres Hindernis war die Schilderkennung mithilfe von TensorFlow, die sich als problematisch erwies und aufgrund unzureichenden Supports nicht integriert werden konnte. Einige Verkehrsschilder wurden nicht korrekt erkannt, da sie sich zu ähnlich waren, was uns zwang, die Anzahl der erkannten Schilder auf ein Minimum zu reduzieren.

Libraries, Plugins, Assets und Quellen

Für unsere Führerschein-AR-App haben wir eine Vielzahl von Tools und Ressourcen eingesetzt, um ein optimales Ergebnis zu erzielen. Das Projekt wurde hauptsächlich mit Unity 2022 entwickelt, was uns eine leistungsstarke und flexible Entwicklungsumgebung bot. Zur Version Control und Zusammenarbeit im Team nutzen wir GitHub. Für die iOS-Entwicklung kam Xcode zum Einsatz, und für die AR-Funktionalitäten haben wir ARKit sowie Unity XR integriert, was uns erweiterte Features ermöglichte. Für die Textdarstellung setzten wir auf die neueste Version von TextMeshPro, die uns eine präzise und ansprechende Schriftwiedergabe ermöglichte. Als Asset-Quelle verwendeten wir das "Road Sign - Big Pack" von Mixall⁵, das eine Vielzahl an Verkehrsschildern enthielt und so zur Realitätsnähe unserer App beitrug. Wie bereits im oberen Teil beschrieben nutzen wir für die Automodelle kostenlose Sketchfab-Modelle.

⁵ <https://assetstore.unity.com/packages/3d/props/industrial/road-sign-big-pack-139858>