

Serious 3D-VR-Game mit realistischen Stadtwelten

Studenten



Josip di Benedetto



Fabian Freitag

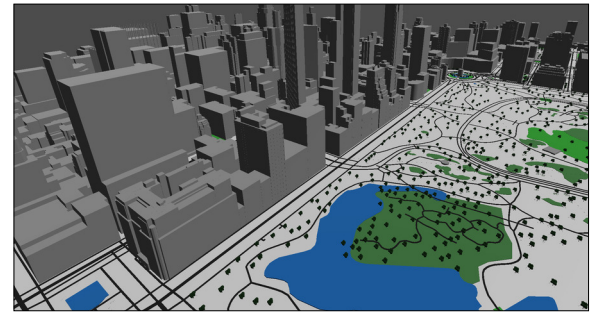
Ausgangslage: Die bisher vorhandenen Projekte, die Daten von OpenStreetMap (OSM) als 3D-Welt darstellen, haben aus Sicht des Industriepartners zwei Probleme: Das erste Problem ist, dass nur ein vordefinierter Bereich dargestellt wird und somit keine globale Abdeckung möglich ist. Das zweite Problem ist, dass sie nicht für Videospiele geeignet sind. Ziel dieser Arbeit ist es, eine realistische 3D-Weltkarte zu erstellen, in der sich der Nutzer bzw. Spieler frei bewegen kann, ähnlich wie in einem Open-World-Spiel. Auf diese Weise kann der Spieler seine Umgebung erkunden. Für die Visualisierung der Geodaten wird die freie Game Engine Godot mit der integrierten Programmiersprache GDScript verwendet. Als Eingabedaten werden ein Datenschema "Mapbox Streets v8" und das "Mapbox Vector Tiles"-Format verwendet. Diese standardisierten, speichereffizienten Kacheln werden aus OSM-Daten mit einer leicht modifizierten Version einer externen Software namens Planetiler (Java) erzeugt. Godot liest diese Vektorkacheln und dekodiert zunächst die Geometrien, um die Objekte darzustellen. Zudem muss die Software den Objektklassen Farben, Materialien und 3D-Symbole (bzw. Modelle) zuordnen. Derzeit sind dies Punktobjekte wie Bäume und Bänke, Polygone wie Gebäude, Linienobjekte (Pfade) wie Strassen, Gehwege und Schienen sowie Flächenobjekte wie Bodenbedeckung, Flüsse und Seen, die ebenfalls Polygone sind.

Ergebnis: In Abb. 1 sieht man einen Ausschnitt des Central Parks in New York und einige Gebäude. Für die meisten Gebäude in New York sind die Höhendaten in OSM erfasst. Für viele andere Gegenden ist das in OSM weniger der Fall. Die Qualität der Darstellung einzelner Objekte hängt von der Datenqualität in OSM ab sowie von der Qualität der Vektorkacheln, die vom Planetiler produziert werden. Als Spieler kann man die Software starten und man landet an einem vordefinierten Startpunkt. Der Spieler kann sich überall auf der Welt hinbewegen oder den Startpunkt selbst anpassen. Dabei wird immer ein Bereich von 4x4 Vektorkacheln dargestellt. Bewegt sich der Spieler in eine bestimmte Richtung, werden automatisch die neuen Kacheln heruntergeladen und visualisiert. Der alte Bereich, von dem sich der Spieler entfernt, wird gleichzeitig aus dem Speicher entfernt. Zu beachten ist, dass die vom Planetiler generierten Vektorkacheln nicht alle OSM-Daten enthalten. In der Abb. 2 sieht man hinter dem Garten Frederiksberg Have die Gebäude, bei denen die Höhendaten nicht so umfangreich erfasst sind, wie in New York.

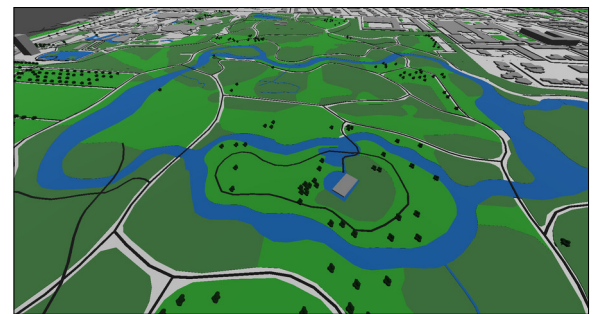
Fazit: Das Hauptziel dieser Studienarbeit wurde erreicht. Was u.a. noch gelöst werden muss, ist die Behandlung der Multipolygone. Aktuell werden Polygone, die innere Ränder haben als solide Strukturen dargestellt, da das "Loch" auch als

Polygon visualisiert wird. Auch die Konfigurierbarkeit kann noch verbessert werden. Je mehr Daten aus OSM dargestellt werden, desto realistischer wird die 3D-Welt. Und die Vektorkacheln enthalten noch einige zusätzliche Informationen, die nicht genutzt werden. Hier setzt auch die Idee an, diese Anwendung nicht nur zum Spielen sondern zur Verbesserung der OSM-Daten zu nutzen, indem der Nutzer auf spielerische Weise Fotos aus der realen Welt - z.B. von Mapillary oder Google Street View - mit der OSM-basierten 3D-Welt in Godot vergleichen kann und dadurch in OSM fehlende Objekte oder Objekteigenschaften ergänzen kann. Beispiele dafür sind Gebäudehöhen, Sitzbänke oder Mülleimer. Eine

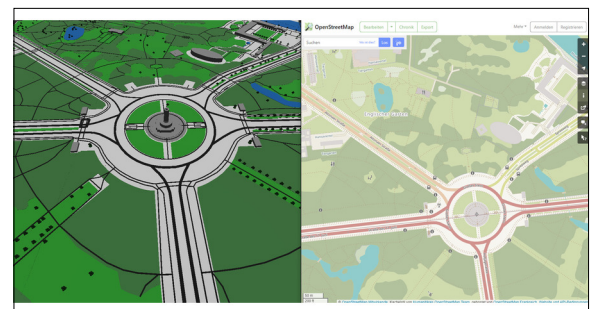
3D-Darstellung des Central Parks in New York mit Godot und mit OpenStreetMap-Daten in der selbst entwickelten Software.
Eigene Darstellung



Der Garten Frederiksberg Have in Kopenhagen. Zu erkennen ist der kleine Fluss und die Landmassen mit einzelnen Pfaden.
Eigene Darstellung



Vergleich der OSM-Daten mit der Darstellung in Godot. Zu erkennen ist der Kreisel mit der Siegestsäule in Berlin.
Eigene Darstellung



Referent

Prof. Stefan F. Keller

Themengebiet Software

Projektpartner
Sourcepole AG, Zürich,
Zürich