

Prozedurale Modellierung von historischen 3D-Stadtmodellen und deren Aufbereitung für 3D-Visualisierungen in einer Smartphone-Applikation am Beispiel der römischen Stadt Augusta Raurica

MARTIN CHRISTEN¹, STEFAN BLASER¹, BENJAMIN LOESCH² & STEPHAN NEBIKER¹

Zusammenfassung: Ein texturiertes 3D-Stadtmodell auf einem Smartphone zu visualisieren, ist immer noch eine grosse Herausforderung, da die Grafikleistung auf diesen Geräten zwar immer leistungsstärker wird, jedoch nicht zu vergleichen ist mit herkömmlichen Desktops oder Laptops. In diesem Beitrag wird ein mögliches Verfahren am Beispiel des historischen 3D-Stadtmodells von Augusta Raurica gezeigt.

1 Einleitung

Augusta Raurica in der Nähe der heutigen Stadt Basel war eine der grössten römischen Städte nördlich der Alpen. Mit einer Augmented Reality App soll den zukünftigen Besucherinnen und Besuchern ein neuer Einblick in die ehemalige Römerstadt ermöglicht werden. An mehreren Stationen sollen dazu verschiedene interaktive Inhalte zum aktuellen Standort abgerufen werden können. Dazu gehören unter anderem detailliert rekonstruierte 3D-Modelle antiker Monumente als Augmented Reality oder gerenderte Panoramaansichten des ehemaligen Strassenraums als Virtual Reality. Die App sollte entweder vorgängig oder vor Ort auf das eigene Smartphone heruntergeladen werden können. Neben den Stationsinhalten sollte die App sowohl eine aktuelle, als auch eine historische 3D-Übersichtskarte zur Orientierung im Gelände bieten.

Dieser Beitrag zeigt wie das flächendeckende texturierte 3D-Stadtmodell von Augusta Raurica erstellt wurde. Die prozedurale Modellierung eignete sich am besten dafür. Als geometrische Grundlage dienten einerseits die aktuellen Grabungsdaten aus dem archäologischen GIS sowie andererseits existierende physikalische 3D-Modelle der Römerstadt, welche vorgängig photographisch erfasst wurden. In die regelbasierte 3D-Modellierung flossen zudem die aktuellsten archäologischen Erkenntnisse mit ein.

Das resultierende historische 3D-Stadtmodell umfasst verschiedene LOD-Stufen. So wurde eine Repräsentation mit einem geringeren Detaillierungsgrad für die Erstellung eines grossen Bronze-modells ausserhalb des Museums verwendet (SCHAUB 2014). Aus einer texturierten Repräsentation wurden mit serverseitigem Rendering 3D-Schrägansichten für die historische Übersichtskarte prozessiert. Die Augmented Reality- und Virtual Reality-Anwendungen beinhalten Teile der detailliertesten Repräsentation des digitalen 3D-Stadtmodells.

¹ Fachhochschule Nordwestschweiz Hochschule für Architektur Bau und Geomatik, Institut Vermessung und Geoinformation, Gründenstrasse 40, CH-4132 Muttenz, E-Mail: [martin.christen, stefan.blaser, stephan.nebiker]@fhnw.ch

² iNovitas AG, Oberrohrdorferstrasse 1c, CH-5405 Baden-Dättwil, E-Mail: benjamin.loesch@inovitas.ch

Dieser Beitrag erläutert einerseits wie die mobile Applikation aufgebaut ist, andererseits wie das 3D-Stadtmodell prozedural modelliert und die 3D-Kartenansicht mit vorprozessierten Schrägan-sichten realisiert wurden. Die Augmented Reality App von Augusta Raurica dient als exemplarisches Beispiel dafür, wie für mobile Endgeräte effizient aufbereitete 3D-Daten zukünftig einem breiten Publikum zugänglich gemacht werden können.

2 Die Römerstadt Augusta Raurica

Augusta Raurica war eine der grössten römischen Städte nördlich der Alpen. Sie liegt am südlichen Rheinufer der heutigen Gemeinden Augst und Kaiseraugst, ca. 15 km östlich von Basel. Die Stadt wurde um 15 v. Chr. gegründet und entwickelte sich in 200 Jahren zu einer regionalen Metropole und einem blühenden Handelsplatz mit rund 15'000 Einwohnerinnen und Einwohnern. Die pulsierende Stadt bot alles, was zum damaligen städtischen Leben gehörte: u.a. ein grosses Theater für bis zu 30'000 Personen (Abb. 1), mehrere Tempel und Heiligtümer, ein eindrückliches Forum, ein Amphitheater sowie eine mächtige Zentraltherme.

Das heutige Augusta Raurica ist einerseits eines der beliebtesten Ausflugsziele in der Nordwestschweiz mit einem Römermuseum und mehr als zwanzig Monumenten und Fundstellen. Dazu gehören das besterhaltene römische Theater nördlich der Alpen, ein komplett eingerichtetes Römerhaus oder etwa eine gewaltige Kastellmauer. Andererseits ist es eine wichtige archäologische Forschungsstätte mit einem wissenschaftlichen Team von rund 60 Mitarbeitenden. Dieses Team hat unter anderem ein umfassendes Geoinformationssystem aufgebaut (BROMBACH 2010). Die archäologischen Daten der Römerstadt sind grösstenteils über das Geodatenportal des Kantons Basel-Landschaft frei zugänglich (<http://geoview.bl.ch/>).



Abb. 1: Das restaurierte Theater von Augusta Raurica (AUGUSTA RAURICA 2016)

3 Das neue 3D-Stadtmodell

3.1 Stadtmodellierungsansätze und Anwendungen in der Archäologie

Die explizite und die prozedurale Modellierung beschreiben zwei verschiedene Ansätze, welche im Rahmen der digitalen 3D-Stadtmodellierung in der Archäologie eingesetzt wurden. Um historische Monumente für die Kulturgütererhaltung zu dokumentieren, fand primär die explizite 3D-Modellierung mit 3D-CAD-Software Anwendung. Die manuelle Digitalisierung von 3D-Polygonen und die Anpassung der so erzeugten 3D-Modelle an den jeweils neusten archäo-

logischen Erkenntnisstand erwiesen sich als sehr zeitaufwändig. Durch die explizite Definition der 3D-Geometrie resultierten jedoch Modelle ohne geometrische Redundanzen und folglich mit geringen Speichervolumina. Mit dem 3D-GIS DILAS stand eine Verwaltungs- und Visualisierungslösung für explizit definierte 3D-Modelle zur Verfügung, welche unter anderem für das frühere 3D-Stadtmodell von Augusta Raurica eingesetzt wurde (WÜST, NEBIKER & LANDOLT 2004). Der prozedurale Ansatz der 3D-Stadtmodellierung wird beispielsweise in PARISH & MÜLLER (2001) abgehandelt. Die urbane Umgebung, bestehend aus Strassennetzwerk und Grundstücken, wird automatisch generiert. Mit einer eigens dafür entwickelten und Computer Generated Architecture (CGA) genannten Programmiersprache lässt sich der Siedlungsraum, inklusive Gebäude, regelbasiert modellieren. Die prozedurale Modellierung wurde schon mehrfach im Zusammenhang der Archäologie und Kulturgütererhaltung eingesetzt. MÜLLER et al. (2006) fokussierten sich mit der 3D-Rekonstruktion der Puuc-Gebäude in Xkipché auf die detaillierte Modellierung einzelner archäologischer Monumente für deren Dokumentation, während dem in SALDANA et al. (2013) die Modellierung ganzer Strassenzüge des antiken Roms als Grundlage für Computerspiele im Vordergrund standen. In HAEGLER et al. (2009) wurde das Prinzip der prozeduralen Modellierung, insbesondere der CGA-Regeldefinition, an verschiedenen Beispielen der Archäologie und der Kulturgütererhaltung diskutiert. DYLLA et al. (2009) zeigen mit dem Projekt Rome Reborn 2.0 auf, dass mithilfe prozeduraler Modellierung, ein homogenes 3D-Städtemodell von Rom erstellt werden konnte.

3.2 Prozedurale Modellierung von Augusta Raurica

Das erstellte 3D-Stadtmodell von Augusta Raurica wurde technisch stark an das 3D-Stadtmodell der ehemaligen römischen Stadt Pompeii (HAEGLER et al. 2009) angelehnt. Für beide Stadtmodelle wurde die Software ESRI CityEngine mit dem in PARISH & MÜLLER (2001) vorgestellten prozeduralen Modellierungsansatz für 3D-Städte eingesetzt. Beide Modelle wurden mit mehreren Detaillierungsgraden (LOD) realisiert und konnten Lücken im Modell mit Wahrscheinlichkeiten schliessen.

Das Modell von Pompeii war vom bekannten antiken Strassennetzwerk ausgehend aufgebaut worden, indem die Grundrisse gewöhnlicher Stadtbauten nach römischem Siedlungsmuster automatisch aus den Freiflächen generiert wurden. Im Gegensatz dazu, waren die Grundrisse beim Modell von Augusta Raurica bereits von früheren physikalischen Modellen gegeben und konnten als Polygone importiert werden. Die CGA-Regeln für die Gebäude wurden somit direkt den importierten Gebäudegrundrissen zugeordnet. Ein weiterer Unterschied zwischen den beiden Modellen besteht im Einbezug der Topographie. Währendem das Modell von Pompeii auf einer flachen Ebene liegt, wurde beim Modell von Augusta Raurica die Höhenlage der Geometrien an die DTM-Höhen angeglichen. Ein wichtiger Grund dafür, war die geplante spätere Nutzung für genaue ortsbezogene Augmented Reality Anwendungen.

Die Angleichung von Gebäudegrundrissen an DTM-Höhen erfolgte in der Annahme, dass es sich dabei stets um horizontale Ebenen handeln würde. Aufgrund dessen kam es innerhalb der für Augusta Raurica charakteristischen Säulendurchgänge, den sogenannten Portiken, zu unerwünschten Abstufungen (Abb. 2, links). Mit einem Python-Skript konnten die Abstufungen automatisiert beseitigt werden, indem die Höhen von Punkten mit identischer Lage aneinander angepasst wurden (Abb. 2, rechts).

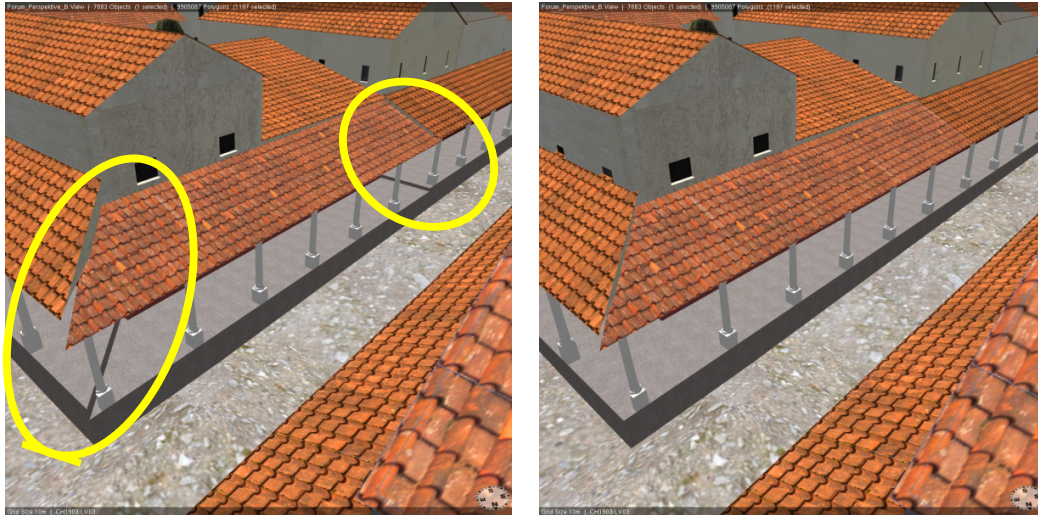


Abb. 2: Säulendurchgänge mit unerwünschten Abstufungen (links) und das Ergebnis nach der automatischen Beseitigung (rechts).

Grossflächige Geometrien, wie beispielsweise Plätze (Abb. 3, links), liessen sich kaum passend an das DTM angleichen, da die DTM-Höhen auch innerhalb der Flächen variierten. Eine Möglichkeit wäre die Unterteilung der Flächen gewesen. Je nach Objekttypus hätte eine weitere Unterteilung in Kombination mit den CGA-Regeln, welche den Teilflächen separat zugeordnet worden wären, zu unerwünschten Mustern geführt. Dies wurde beim Strassen- und Gewässernetzwerk umgangen, indem die betreffenden texturierten Geometrien mit GDAL direkt in die DTM-Textur eingebrannt wurden (Abb. 3, rechts).



Abb. 3: Vergleich Strassennetzwerk mit grossem Platz als 2D-Geometrie (links) und als Objekte welche direkt in die Geländetextur gebrannt wurden (rechts).

3.3 Detaillierungsgrade

Das prozedural modellierte 3D-Stadtmodell von Augusta Raurica umfasst drei verschiedene Detaillierungsgrade. Der niedrigste Detaillierungsgrad LOD1 (Abb. 4, links) repräsentiert

Augusta Raurica als eingefärbtes Strukturmodell. Dabei wurden die aus den GIS-Daten importierten Gebäudegrundrisse extrudiert und zusätzlich mit Dachgeometrien versehen. Sowohl die Gebäudehöhe als auch die Ausprägung des Dachs (Dachform, Dachwinkel und Dachausrichtung) wurden parametrisch definiert, welche sich bei Bedarf dynamisch anpassen liessen. Die LOD1-Repräsentation von Augusta Raurica wurde unter anderem für die Erstellung des Bronze-modells (SCHAUB 2014) verwendet. Der mittlere Detaillierungsgrad LOD2 umfasst eine texturierte Repräsentation des 3D-Modells (Abb. 4, Mitte). Die Dachflächen wurden in Quadrate mit einer Seitenlänge von 1.80 m, was ungefähr der Seitenlänge von vier römischen Ziegeln entspricht, unterteilt. Nach zufälligem Muster wurden die Quadrate mit Texturen aus einer, aus 14 Dachtexturen bestehender Bibliothek versehen. Dadurch konnte eine realitätsnahe und uniforme römische Dachlandschaft ohne störende repetitive Muster erstellt werden. Die Gebäudefassaden wurden zusätzlich zur LOD1-Repräsentation parametrisch ausmodelliert. Die Parameter von Fenster und Türen nahmen standardmässig zufällige Werte innerhalb eines definierten Wertebereichs an und liessen sich bei Bedarf pro Etage und Fassade separat anpassen. Die LOD2-Repräsentation diente zur Erstellung der 3D-Schrägansichten von Augusta Raurica. Der höchste Detaillierungsgrad LOD3 enthält, im Gegensatz zum LOD2, weitere ausmodellierte Details, wie einzelne Ziegelsteine oder ausgearbeitete Fenster mit Fensterglas, welches physikalisch korrekt auf dessen Beleuchtung reagiert (Abb. 4, rechts). Der LOD3 eignete sich für die Visualisierung und Betrachtung einzelner Bauten und Monumente aus der Nähe.



Abb. 4: Vergleich der drei Detaillierungsgrade LOD1 (links), LOD2 (Mitte), LOD3 (rechts) am Beispiel eines römischen Stadtquartiers.

4 Das 3D-Stadtmodell als Basis für interaktive Anwendungen

Das flächendeckende Stadtmodell von Augusta Raurica bildet eine 3D-Geodatenbasis der historischen Römerstadt. Die Daten lassen sich erst mit geeigneten Anwendungen und Produkten für eine breite Öffentlichkeit nutzbar machen. Denkbar dafür wären Webanwendungen oder Applikationen für mobile Endgeräte, womit das Modell, dank geeigneter Navigations- und Visualisierungskonzepte, auf einfache und zugleich attraktive Art und Weise betrachtet werden könnte. Die grossen Herausforderungen solcher Anwendungen liegen in den zurzeit begrenzten Rechen- und Speicherkapazitäten von mobilen Endgeräten. Schon die Darstellung eines 3D-Stadtmodells in der Grösse von Augusta Raurica erweist sich aufgrund der genannten Limitierungen als schwierig. Um das erzeugte 3D-Stadtmodell in mobilen Anwendungen nutzen zu können, wur-

den daraus verschiedene Produkte bzw. 3D-Anwendungen abgeleitet, die in der Folge vorgestellt werden.

4.1 Panoramaansicht im Strassenraum

Eine realisierte Anwendung zur interaktiven Betrachtung des 3D-Modells besteht aus 360°-Panoramen von ausgewählten Standorten. Für die spätere Darstellung sind vorgerenderte Ansichten des Modells notwendig, welche aus Bilddaten bestehen. Für die Visualisierung von 360°-Panoramen werden in der Computergraphik verbreitet Cube Maps (GREENE 1986) eingesetzt. Die 360°-Ansicht wird dabei als Würfel angenähert (Abb. 5, links), wobei sechs entsprechend ausgerichtete perspektivische Renderings auf die Würfelflächenseiten zu liegen kommen. Der Würfel kann beliebig um die im Würfelzentrum liegende virtuelle Kamera gedreht werden. Aufgrund des perspektivischen Renderings wären bei der Cube Map ungewünschte Verzerrungen entstanden. Durch die Verwendung eines Zylinders anstelle eines Würfels liessen sich die Verzerrungen in der horizontalen Richtung vermeiden. Dabei kam ein 360°-Panoramabild auf die Innenfläche des Zylindermantels zu liegen. Da sowohl für die Unter- als auch die Oberseite keine Renderings erstellt wurden, liess sich der Zylinder bei der Betrachtung lediglich um die Zylinderachse drehen. Beim Rendering des 360°-Panoramabilds wurde die Zylinderprojektion mit 36 perspektivisch gerenderten Streifen angenähert (Abb. 5, rechts). Die 36 streifenförmigen Renderings wurden in CityEngine mithilfe eines Python-Skripts automatisiert erstellt und anschliessend mit der Python Image Library (PIL) zu einem Panoramabild zusammengefügt.

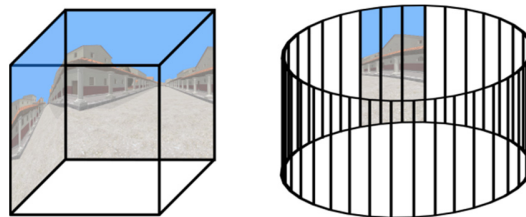


Abb. 5: Schematische Darstellungen des Panoramarenderings für eine Cube Map (links) und für einen Zylinder (rechts).

Das erstellte Panoramabild wurde mittels Bildbearbeitung zusätzlich künstlerisch aufgewertet, indem Römerinnen und Römer, sowie diverse kleinere Objekte in die Szene eingefügt worden waren (Abb. 6). Dieser Schritt könnte künftig automatisiert werden, indem die zusätzlichen Elemente als 3D-Objekte direkt in das Modell eingefügt würden.



Abb. 6: Mit CityEngine gerendertes, künstlerisch aufgewertetes Panoramabild einer Strassenszene in Augusta Raurica.

4.2 Aufbereitung detaillierter 3D-Architekturmodelle

Die Darstellung von detaillierten 3D-Architekturmodellen mit markerbasiertem Augmented Reality wurde als weitere Anwendung realisiert. Beim markerbasierten Augmented Reality werden Features im Live-Kamerabild des mobilen Endgeräts detektiert und mit Features eines bereits bestehenden Markers verglichen. Mit den übereinstimmenden Features können die äusseren Orientierungsparameter der Kamera in Echtzeit errechnet werden. Dank der laufend bekannten äusseren Orientierung, können 3D-Inhalte, welche sich im Markerkoordinatensystem befinden, lagekorrekt ins Kamerabild eingeblendet werden (LOESCH et al. 2015). Die zu visualisierenden detaillierten 3D-Architekturmodelle wurden aus dem mit CityEngine erstellten prozeduralen Stadtmodell von Augusta Raurica exportiert. Die exportierten 3D-Modelle wiesen redundante Geometrien, wie doppelt vorhandene Punkte und unnötige Unterteilungen innerhalb ebener Flächen auf, welche auf die prozedurale Modellierung zurückzuführen waren. Da die Redundanzen einerseits unnötigen Speicherbedarf und andererseits höheren Rechenaufwand für die Darstellung verursacht hätten, wurden die exportierten Modelle skriptbasiert mit der 3D-Graphiksoftware Blender optimiert. Mit dem Löschen doppelter Stützpunkte, sowie innenliegender Flächen, konnte der Speicherbedarf für die Geometrien um über die Hälfte verringert werden.

4.3 Cloud-basiertes Rendering der 3D-Schrägansichten

Für die 3D-Visualisierung grosser 3D-Geodatenätze in Smartphone Apps kann aus Gründen von Performance und App-Grösse ein bildbasierter Ansatz verwendet werden, welcher mit dem 3dmaps.ch Framework realisiert wurde (CHRISTEN 2016). Es werden verschiedene Ansichten eines 3D-Stadtmodells inklusive Orthophoto und Geländemodell als Cloud-Dienst gerechnet, gecached und zum Client gestreamt. Das Rendering unterstützt verschiedene Zoom-Level. Die berechneten Bilder folgen dem Prinzip des G-Buffers (SAITO 1990). Der erste Bildtyp ist eine Farbkarte, welche die Farben – ohne Beleuchtung – der 3D-Ansicht enthält. Der zweite Bildtyp ist eine Normalmap, welche die Normalen in RGB kodiert enthält. Mit diesen zwei Bildtypen ist es beispielsweise möglich eine dynamische Beleuchtung zu rechnen. Ein dritter Bildtyp enthält die ID der Objekte, kodiert als Farbe. Ein vierter Bildtyp speichert die Tiefeninformation als Bild ab. Die 32bit Gleitkommazahl wird in RGBA kodiert, und kann auch dekodiert werden (CHRISTEN 2016). Mit der Tiefenkarte kann die 3D-Position jedes Pixels auf dem Bildschirm ermittelt werden.

Die 3dmaps.ch Web-Applikation besteht aus verschiedenen Python-Scripts, welche von einem Webinterface gestartet werden können. Das Webinterface wurde mittels Flask (FLASK 2015; LERNER 2014) entwickelt und benutzt MongoDB, eine Open Source Dokument-Datenbank, welche insbesondere für die Skalierung entwickelt wurde (MONGODB 2015). Das Web-Interface ist optional, die Python-Scripts können auch direkt aus der Konsole gestartet werden. Die gesamte Server-Architektur folgt der “View Extension” des OGC 3D Portrayal Service (OPEN GEOSPATIAL CONSORTIUM 2015). Zusätzlich wurde ein Caching hinzugefügt, damit dieselbe Ansicht nur einmal gerechnet wird.

Zunächst wird das Skript „createScene.py“ gestartet, um eine neue 3D-Szene zu definieren. Eine 3D-Szene besteht normalerweise aus 3D-Geometrien, Texturen und Materialien. Eine 3D-Szene kann auch Punktwolken enthalten. Sobald die Szene definiert ist, können die Daten hinzugefügt werden. Dies geschieht mit dem Skript „addFilesToScene.py“. Orthophotos und Geländemodelle

können zusammen auch zu einem 3D-Modell konvertiert werden. Ist die Szene komplett definiert, so kann dieses optimiert werden, beispielsweise durch Entfernung doppelter Flächen, Punkte und Materialien. Eine weitere Optimierung ist die Erstellung einer räumlichen Datenstruktur, um die Renderingzeit signifikant zu verkürzen. Eine der populärsten räumlichen Struktur für diesen Zweck ist die BVH - bounding volume hierarchy (BARRINGER & AKENINE-MÖLLER 2014). Die BVH wird mit allen 3D-Objekten, inklusive Punktwolken und Terrain (NEBIKER et al. 2010), berechnet. Zudem werden Texturen in ein Format konvertiert, welches Mehrfachauflösung und direkten Pixelzugriff unterstützt. Sobald die Szene fertig vorbereitet ist, können Render-Ansichten konfiguriert werden. Die Definition umfasst die Bildtypen (Farbkarte, Normalenkarte, Tiefenkarte, Id-Karte), die Kameraposition und die Blickrichtung. Die Renderingdefinition geschieht mit dem Python-Script „createRendering.py“. Beispiele für verschiedene Blickrichtungen sind in der Abb. 4 zu sehen.

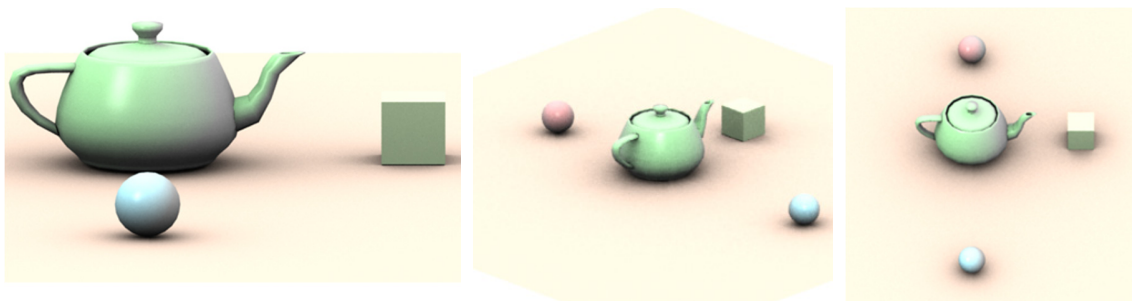


Abb. 7: Verschiedene Blickwinkel auf dieselbe Szene.

Die Ansichten werden als Kacheln in einer Quadtree-Struktur gerechnet, wie in Abb. 7 gezeigt. Die Objekte werden dabei in Parallelprojektion gerendert. Der Vorteil dabei ist, dass ein gleichmässiger Level of Detail mit einem konstanten Massstab über den ganzen Bildschirm dargestellt wird. Ein weiterer Vorteil der Quadtree-Struktur ist, dass nur die unterste Zoom-Stufe gerechnet werden müssen und die restlichen Kacheln über Bildverarbeitung erstellt werden. Somit kann das volle Potential der Beschleunigungs-Datenstruktur (BVH) ausgenutzt werden.

Das Rendering der untersten Stufe geschieht über Ray-Tracing. Dies ermöglicht die Verwendung von nahezu beliebig viel Geometrie und Texturen, da bei der Bildgenerierung jede Operation pixelweise ausgeführt wird. Für das Ray Tracing kann Renderman (RENDERMAN 2015) oder POV-Ray verwendet werden (POVRAY 2015). Die Kacheln, welche gerendert werden, sind als Render-Jobs in der MongoDB gespeichert und können in der Cloud von einem beliebigen Node gerechnet werden. Es können bei Bedarf jederzeit neue Rendering-Nodes hochgefahren werden. Die resultierenden Bilder werden wiederum direkt in der Cloud gespeichert. Das 3D-Modell von Augusta Raurica wurde auf 6 Zoomstufen vorgerechnet und ist in Abb. 8 zu sehen.

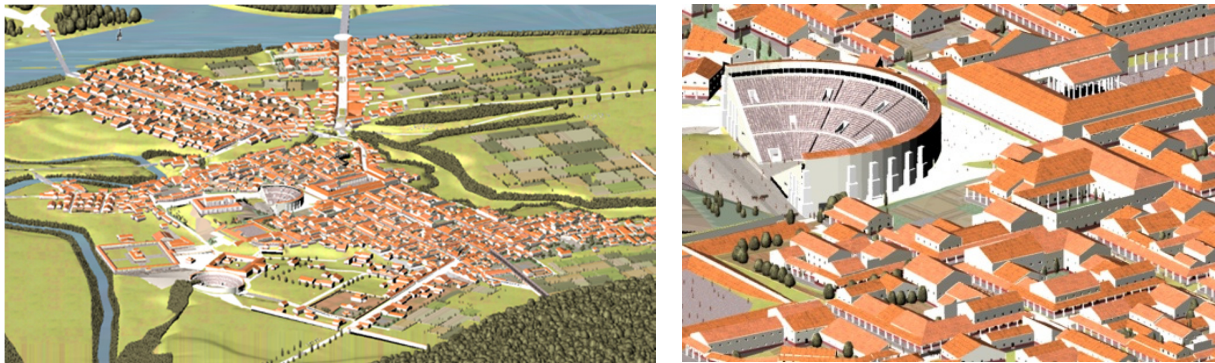


Abb. 8: 3D-Ansicht von Augusta Raurica mittels vorgerechneter Schrägansichten

5 Smartphone-Applikation

In Zusammenarbeit mit dem Archäologen-Team von Augusta Raurica wurde die App „unsichtbares sichtbar machen“ für Android und iOS entwickelt. Die App soll beim Besuch des historischen Geländes zusätzliche Informationen liefern. Da heute grosse Teile der antiken Römerstadt verschwunden sind, soll auch mittels Augmented Reality ein Eindruck der einstigen Situation vermittelt werden. Zusätzlich soll die historische 3D-Ansicht angezeigt werden.

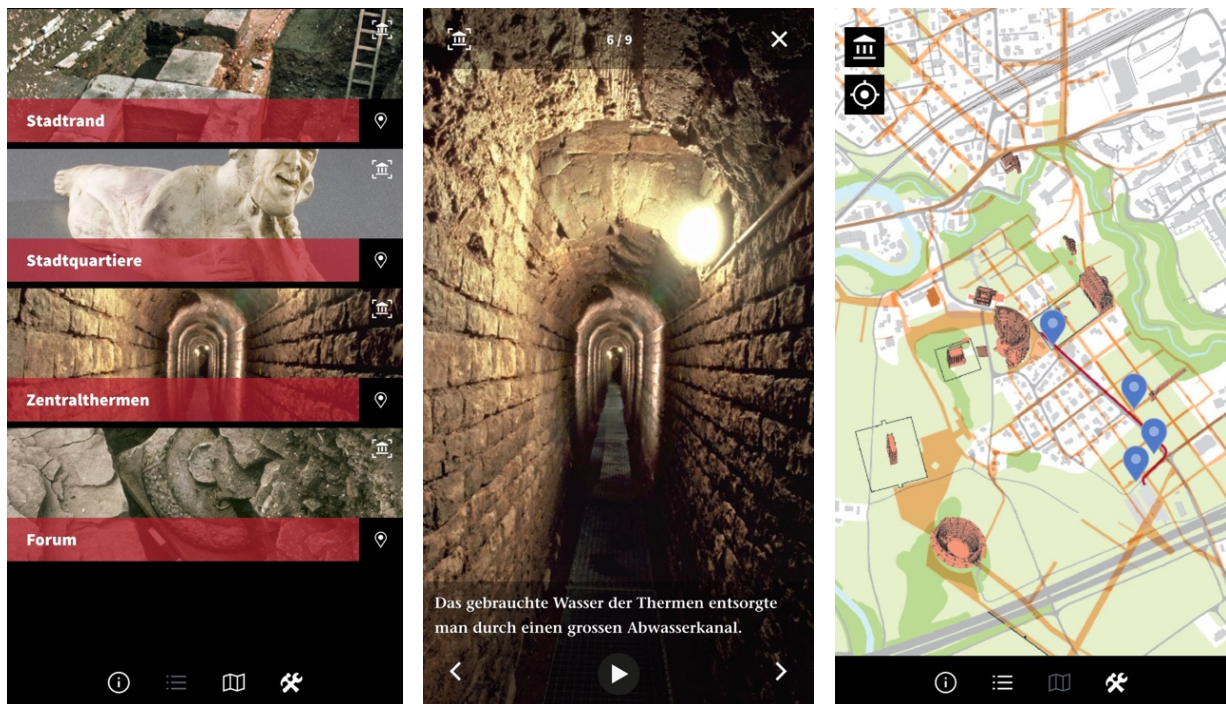


Abb. 9: Das Hauptmenu der App (links) und Beispielinhalt mit Audio-Guide (Mitte) und Kartenansicht mit aktueller Position und den 4 Posten (rechts).

Die App hat 4 verschiedene Posten, welche zusätzliche Informationen in Form von Bildern und Text liefern. Die Informationen können auch vorgelesen werden, somit dient die App auch als

Audio-Guide. Die Posten werden über das Hauptmenu ausgewählt. Die aktuelle Position sowie die Postenpositionen werden auf einer 2D Karte dargestellt (Abb. 9).

Die App soll auch von Schulklassen bedient werden, daher ergab sich die Einschränkung, dass für die Bedienung der App keine Internet-Verbindung erforderlich sein darf. Für den Download der App steht am Museumseingang WLAN zur Verfügung. Apps haben für Android eine Maximalgrösse von 100 MB (GLICK 2015). Das texturierte 3D-Modell besteht aus ca. 1 GB an Geometrie und Texturdaten und kann nicht direkt mit der App mitgeliefert werden oder während der Benutzung heruntergeladen werden. Daher wurde der Ansatz mit vorgerechneten Schrägansichten gewählt (siehe Kapitel 4.3). Dabei kann das historische 3D-Modell mit nur ca. 10 MB komplett dargestellt werden und erfordert praktisch keine Rechenressourcen des Smartphones. Die heutige Position wird mit den Ortungsdiensten des Smartphones zusätzlich angezeigt (Abb. 10).

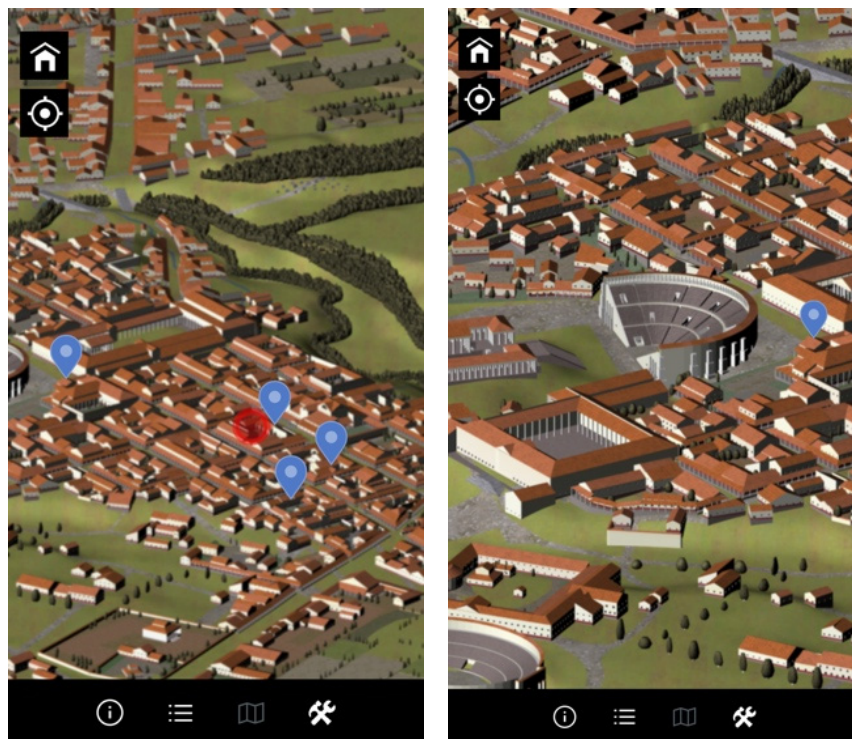


Abb. 10: 2D Karte mit überlagerten historischen Daten (links) und 3D Karte erstellt mit der vorgerechneten Schrägansicht.

Eine weitere Eigenschaft der App ist, dass an jedem Posten mittels Augmented Reality ein Blick in die Vergangenheit ermöglicht wird. Dabei wird ein rekonstruiertes, texturiertes 3D Modell überlagert. Als natürliche Marker dienten Archäologische Rekonstruktionszeichnungen der Monumente, welche auf Hinweistafeln im Gelände angebracht wurden (Abb. 11). Die Augmented Reality-Anwendung wurde um eine zusätzliche Navigationsfunktionalität erweitert, mit welcher das zu überlagernde 3D-Architekturmodell mit Wischgesten innerhalb eines zulässigen Bereichs im Raum gedreht werden kann.

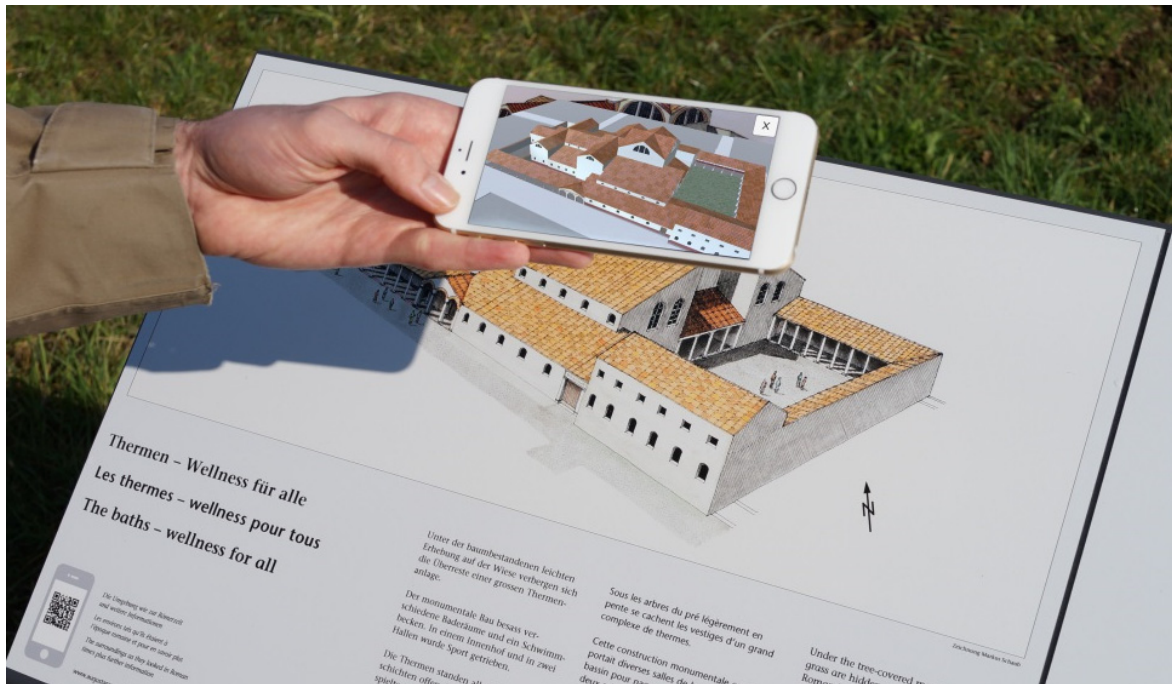


Abb. 11: Detailliertes 3D-Architekturmodell der Zentralthermen als Augmented Reality-Anwendung. Die Rekonstruktionszeichnung auf der Hinweistafel im Gelände dient als Marker.

6 Diskussion und Ausblick

Im vorliegenden Beitrag wurde die prozedurale Generierung eines texturierten historischen 3D-Stadtmodells am Beispiel der Römerstadt Augusta Raurica aufgezeigt. Das Stadtmodell umfasst drei unterschiedliche Detaillierungsstufen (LOD), zwischen welchen mittels Parameteranpassung gewechselt werden konnte. Damit lässt sich das 3D-Stadtmodell sowohl für Anwendungen mit geringem Detaillierungsgrad, wie im Beispiel des realisierten grossen Bronzemodells, oder aber auch für Augmented Reality Anwendungen mit teilweise sehr hohen Anforderungen an die Realitätstreue verwenden. Am Beispiel dreier Repräsentationsformen wurde aufgezeigt, wie ein sehr grosses 3D-Stadtmodell in einem mobilen Kontext und für ein breites Publikum nutzbar gemacht werden kann. Vorgerenderte interaktive 360°-Panoramen von ausgewählten Standorten wurden erzeugt, um Einblicke in den altrömischen Strassenraum zu ermöglichen. Im Rahmen einer zweiten Anwendung wurde der Speicherbedarf von detaillierten prozeduralen 3D-Architekturmodellen verlustfrei halbiert, sodass die Modelle auf mobilen Endgeräten visualisiert werden können. Schliesslich wurde aus dem Stadtmodell mit 3dmaps.ch ein serverseitiger Bilddienst generiert, der performante Betrachtung aus unterschiedlichen Perspektiven ermöglicht. Die untersuchten Augmented Reality Konzepte wurden erfolgreich in die neue Besucher-App "Unsichtbares sichtbar machen" (<http://www.augustaraurica.ch/app>) der Römerstadt Augusta Raurica integriert.

Die vorgestellten Konzepte und Technologien sollen zukünftig auf andere 3D-Stadtmodelle übertragen werden, um AR-Anwendungen etwa im Tourismus oder im Schulunterricht zu realisieren. Die vorgestellten bildbasierten Rendering-Ansätze und die entsprechenden neuen OGC

Standards bieten zudem ein grosses Potential für inhaltliche, funktionale und semantische Erweiterungen. Dazu gehören beispielsweise die zukünftige Integration von 3D-Punktwolken oder etwas die Nutzung der Tiefeninformation oder zusätzlicher thematischer Layer.

7 Dank

Die Autoren danken der Römerstadt Augusta Raurica und ihren kompetenten wissenschaftlichen Mitarbeitenden für die langjährige Projektzusammenarbeit und für das Vertrauen, insbesondere im Zusammenhang mit dem kürzlich erfolgreich realisierten Projekt "Unsichtbares sichtbar machen". Besonderen Dank sei Markus Schaub, Leiter der Publikumsgrabungen und wissenschaftlicher Zeichner der Römerstadt Augusta Raurica, ausgesprochen. Ohne die Einbringung seines fundierten archäologischen Fachwissens und seiner zeichnerisch-gestalterischen Erfahrung und der sehr engagierten, unkomplizierten Zusammenarbeit wäre die Realisierung des 3D-Modells in dieser Ausprägung undenkbar gewesen.

8 Literaturverzeichnis

- AUGUSTA RAURICA, 2016: Webseite von Augusta Raurica, <http://www.augustaurica.ch>.
- BARRINGER, R. & AKENINE-MÖLLER, T., 2014: Dynamic Ray Stream Traversal. *ACM Transactions on Graphics* **33** (4), Article 151.
- BROMBACH, U., 2010: Fünf Jahre Geografisches Informationssystem in Augusta Raurica. *Jahresberichte aus Augst und Kaiseraugst* **31**, 193-205.
- CHRISTEN, M., 2016: Visualisierung komplexer 3D-Geodaten im (mobilen) Webbrowser mittels Cloud-prozessierten Schrägansichten. *Geoinformationssysteme 2016 – Beiträge zur 3. Münchner GI-Runde 2016*, Kolbe, T.H., Bill, R. & Donaubaue, A. (Hrsg.), Wichmann, 68-76.
- DYLLA, K., FRISCHER, B., MÜLLER, P., ULMER, A. & HAEGLER, S., 2009: Rome Reborn 2.0: A Case Study of Virtual City Reconstruction Using Procedural Modeling Techniques. *Proceedings of the CAA Conference 2009*, Williamsburg, Virginia, USA, 62-66.
- FLASK, 2015: Flask Documentation / User's Guide, <http://flask.pocoo.org/docs/0.10/> (letzter Zugriff: 23.4.2015).
- GLICK K., 2015: Support for 100MB APKs on Google Play. <http://android-developers.blogspot.ch/2015/09/support-for-100mb-apks-on-google-play.html> (letzter Zugriff: 12.4.2016).
- GREENE N., 1986: Environment Mapping and Other Applications of World Projections. *IEEE Computer Graphics and Applications* **6** (11), 21-29.
- HAEGLER, S., MÜLLER, P. & VAN GOOL, L., 2009: Procedural Modeling for Digital Cultural Heritage. *EURASIP Journal on Image and Video Processing*, Volume 2009, 7.
- LOESCH, B., CHRISTEN, M., WÜEST, R. & NEBIKER S., 2015: Geospatial Augmented Reality – Lösungsansätze mit natürlichen Markern für die Kartographie und die Geoinformationsvisualisierung im Aussenraum. *Publikationen der DGPF, Band 24*, T. Kersten (Hrsg.), 35. Wissenschaftlich-Technische Jahrestagung der DGPF, 16.-18. März 2015 an der Universität zu Köln, Tagungsband auf CD-ROM, 89-97.

- LERNER, M.R., 2014: At the forge: flask. Issue of Linux Journal, July 2014, Belltown Media, Houston, TX.
- MONGODB, 2015: The MongoDB 3.0 Manual, <http://docs.mongodb.org/manual/> (letzter Zugriff: 20.11.2015).
- MÜLLER, P., VEREENOOGHE, T., WONKA, P., PAAP, I. & VAN GOOL, L.J., 2006: Procedural 3D Reconstruction of Puuc Buildings in Xkipché. Proceedings of the 7th International Symposium on Virtual Reality, Archeology and Cultural Heritage VAST, 139-146.
- NEBIKER, S., BLEISCH, S. & CHRISTEN, M. 2010: Rich point clouds in virtual globes – A new paradigm in city modeling? Computers, Environment and Urban Systems **6** (34), 508-517.
- OPEN GEOSPATIAL CONSORTIUM, 2015: OGC seeks public comment on candidate 3D Portrayal Service Standard, <http://www.opengeospatial.org/pressroom/pressreleases/2165> (letzter Zugriff: 19.11.2015).
- PARISH, Y.I.H. & MÜLLER P., 2001: Procedural modeling of cities. Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, 301-308.
- POVRAY, 2015: Persistence of Vision Renderer „POV-Ray“. <http://povray.org> (letzter Zugriff: 20.11.2015).
- RENDERMAN, 2015: Pixar Renderman Pro Server. <http://renderman.pixar.com> (letzter Zugriff: 20.4.2015).
- SAITO, T. & TAKAHASHI, T., 1990: Comprehensible rendering of 3-D shapes. ACM SIGGRAPH Computer Graphics **24** (4), 197-206.
- SALDAÑA, M., & JOHANSON, C., 2013: Procedural Modeling for Rapid-Prototyping of Multiple Building Phases. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences **40** (5/W1), 205-210.
- SCHAUB, M., 2014: Das Bronzemedell von Augusta Raurica: vom Stadtplan zum Stadtmodell. Augusta Raurica Magazin, 2014(1), 7-9.
- WÜST, T., NEBIKER, S. & LANDOLT, R., 2004: Applying the 3D DIS DILAS to Archaeology and Cultural Heritage Projects – Requirements and First Results. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **35** (Part 5), 407-412.