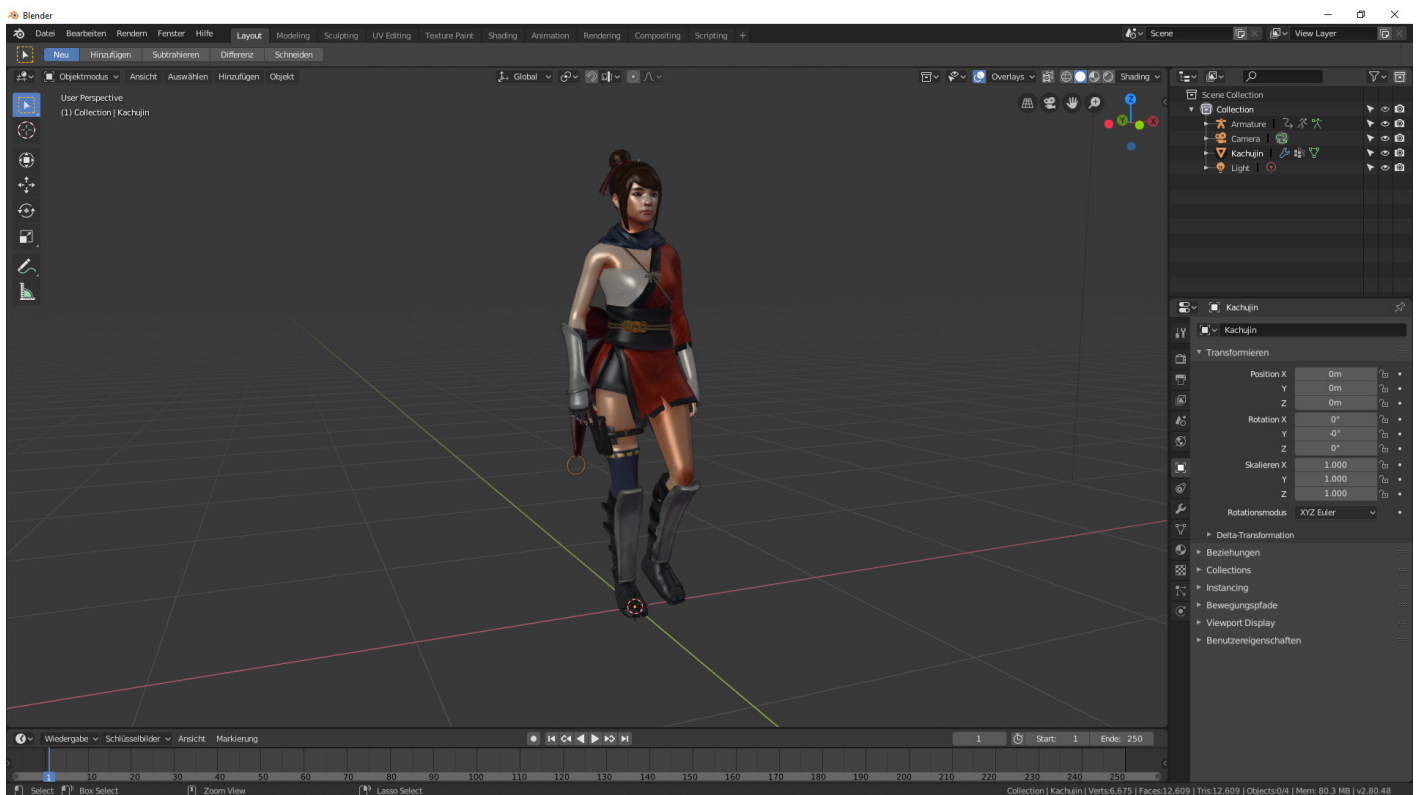


3D Modelle im Web mit three.js

Kategorien : [Open Source](#)

Schlagwörter : [3D](#), [3D-Figuren](#), [glTF](#), [Three.js](#), [WebGL](#)

Datum : 27. Mai 2019



3D-Modelle sind in Hunderten von Dateiformaten verfügbar, die jeweils unterschiedliche Zwecke, verschiedene Funktionen und unterschiedliche Komplexität aufweisen. Obwohl three.js viele Lader zur Verfügung stellt, spart die Wahl des richtigen Formats und Workflows später Zeit und Frustration. Einige Formate sind schwer zu verarbeiten, für Echtzeit-Erlebnisse ineffizient oder werden derzeit einfach nicht vollständig unterstützt.

Dieser Leitfaden enthält einen empfohlenen Workflow und Vorschläge, was Sie versuchen sollten, wenn die Dinge nicht wie erwartet laufen.

Empfohlener Arbeitsablauf

Wenn möglich, empfehlen wir die Verwendung von glTF (GL Transmission Format). Sowohl die .GLB- als auch die .GLTF-Version des Formats werden gut unterstützt. Mit glTF 2.0 wurde ein laufzeitneutrales, offenes Standardformat für die Echtzeit-Abgabe von 3D-Inhalten im Internet veröffentlicht. Zu den Funktionen gehören Netze, Materialien, Texturen, Skins, Skelette, Morph-Ziele, Animationen, Lichter und Kameras.

Die folgenden Tools beinhalten den glTF-Export:

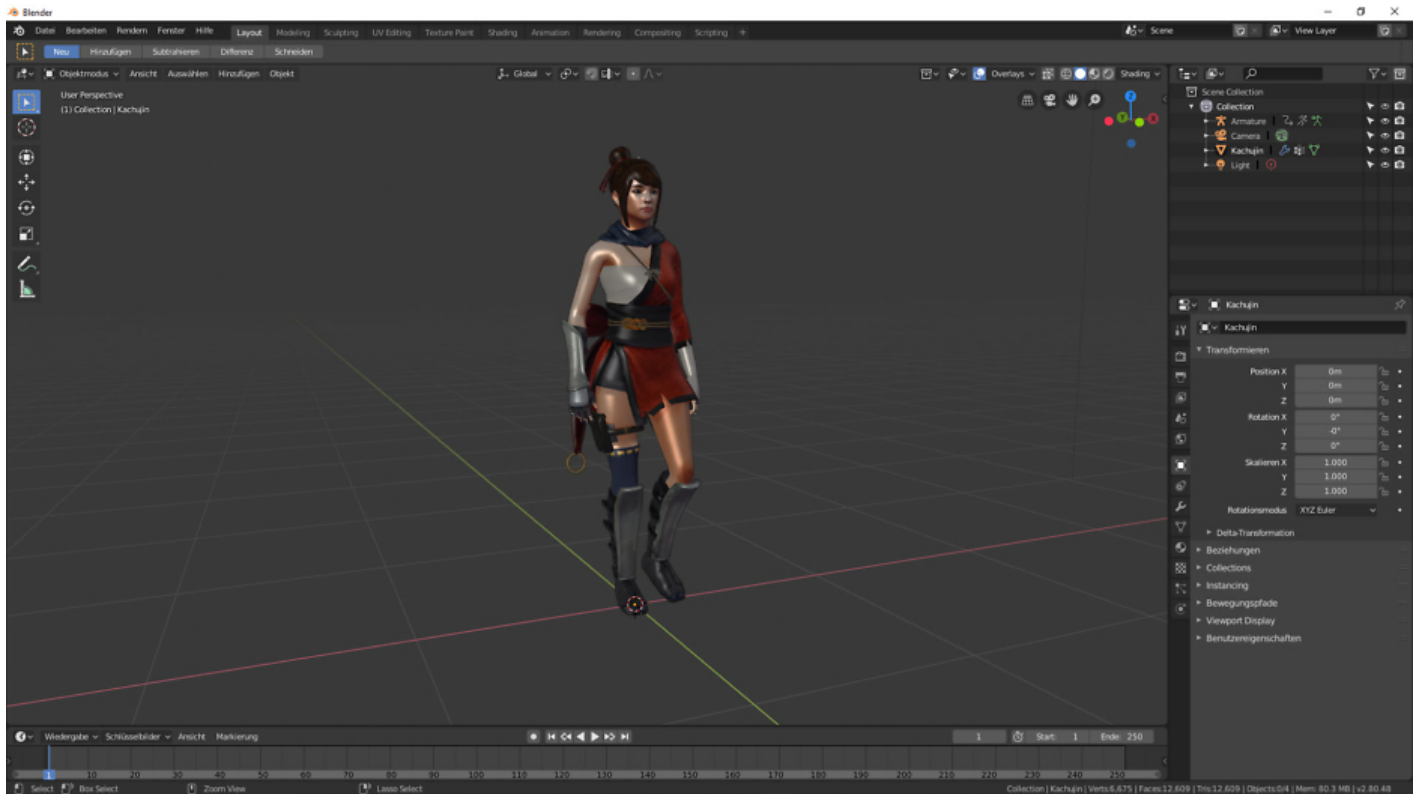
- [glTF-Blender-IO](#) by the Khronos Group
- [COLLADA2GLTF](#) by the Khronos Group
- [FBX2GLTF](#) by Facebook
- [OBJ2GLTF](#) by Analytical Graphics Inc
- ...und [weitere](#)

Für Blender 2.79 stellen wir eine [Installations-Anleitung](#) zur Verfügung. In Blender 2.80 steht der Import und Export von .GLB- als auch die .GLTF-Version zur Verfügung.

Wenn glTF keine Option ist, sind auch gängige Formate wie FBX, OBJ oder COLLADA verfügbar und werden regelmäßig gepflegt. Wir verwenden in unseren Beispielen das glTF (GL Transmission Format).

glTF (GL Transmission Format) ist eine offene Formatspezifikation für die effiziente Bereitstellung und das Laden von 3D-Inhalten. 3D Modelle können entweder im JSON (.gltf) oder im binären

(.glb) Format bereitgestellt werden. Externe Dateien speichern Texturen (.jpg,.png) und zusätzliche binäre Daten (.bin). Ein glTF-Asset kann eine oder mehrere Szenen liefern, einschließlich Netze, Materialien, Texturen, Skins, Skelette, Morph-Ziele, Animationen, Lichter und Kameras.



3D Modell von mixamo.com

Laden

Code Beispiel

// Globales Skript

Nachdem Sie den Lader importiert haben, können Sie Ihrer Szene ein Modell hinzufügen.

```
// Instantiate a loader var loader = new THREE.GLTFLoader(); // Optional
: Provide a DRACOLoader instance to decode compressed mesh data THREE.DRA
COLoader.setDecoderPath( 'js/libs/draco' ); loader.setDRACOLoader( new TH
REE.DRACOLoader() ); // Optional: Pre-fetch Draco WASM/JS module, to sa
ve time while parsing. THREE.DRACOLoader.getDecoderModule(); // Load a g
LTF resource loader.load( // resource URL 'models/gltf/untitled.glb',
// called when the resource is loaded function ( gltf ) { scene.add(
gltf.scene ); gltf.animations; // Array gltf.scene; // THREE.Scene
gltf.scenes; // Array gltf.cameras; // Array gltf.asset; // Object
}, // called while loading is progressing function ( xhr ) { consol
e.log( ( xhr.loaded / xhr.total * 100 ) + '% loaded' ); }, // called w
hen loading has errors function ( error ) { console.log( 'An error ha
ppened' ); } );
```

Bevor wir aber den 3D Inhalt im glTF (GL Transmission Format) laden fügen wir Draco hinzu.

Auszug aus unserem obigen Beispiel

```
// Optional: Provide a DRACOLoader instance to decode compressed mesh dat
a THREE.DRACOLoader.setDecoderPath( 'js/libs/draco' ); loader.setDRACOLoa
der( new THREE.DRACOLoader() ); // Optional: Pre-fetch Draco WASM/JS mo
dule, to save time while parsing. THREE.DRACOLoader.getDecoderModule();
```

Draco ist eine Bibliothek zum Komprimieren und Dekomprimieren von geometrischen 3D-Netzen und Punktwolken. Ziel ist es, die Speicherung und Übertragung von 3D-Grafiken zu verbessern.

Draco wurde für die Effizienz und Geschwindigkeit der Kompression entwickelt und gebaut. Der Code unterstützt Komprimierungspunkte, Verbindungsinformationen, Texturkoordinaten, Farbinformationen, Normalen und alle anderen allgemeinen Attribute, die mit der Geometrie verbunden sind. Mit Draco können Anwendungen mit 3D-Grafik deutlich kleiner sein, ohne die Bildtreue zu beeinträchtigen. Für den Anwender bedeutet dies, dass Apps nun schneller heruntergeladen werden können, 3D-Grafiken im Browser schneller laden können und VR- und AR-Szenen nun mit einem Bruchteil der Bandbreite übertragen und schnell gerendert werden können.

Website: <https://google.github.io/draco/>

GitHub: <https://github.com/google/draco>

Verwandeln Sie Ihren Commerce mit AR und 3D-Produktvisualisierung!

Bei uns geht es um Techniken, die es schaffen, das Produkt zum Erlebnis zu machen. Virtual & Augmented Reality, 360 Grad-Videos, Darstellungen in 3D, virtuelle Showrooms. Die Besucher:innen sollen eintauchen in die Welt des Unternehmens mit immersiven Technologien.



Sie können uns mit der Erstellung von individuellen 3D-Visualisierungen beauftragen. Jeder kann 3D-Visualisierungen bei unserem Kreativservice bestellen - unabhängig davon, ob Sie nur ein einzelnes 3D-Modell benötigen oder viele.

Wir unterstützen Sie bei der Umsetzung Ihres Augmented Reality (AR) oder Virtual Reality (VR) Projektes! Egal ob [Produktfotografie](#), [3D Scan Service](#), [3D-Visualisierung](#) oder fertige [3D Modelle für AR/VR](#) – wir beraten Sie persönlich und unverbindlich.

Wo kann ich Anregungen, Lob oder Kritik äußern?

Ihre Meinung ist uns wichtig! Schreiben Sie uns, was Ihnen in Bezug auf unser Angebot bewegt. info@r23.de

R23 — Ihre Digitalagentur für Virtual Reality und interaktive Markenerlebnisse

Wünschen Sie ein individuelles Angebot auf Basis Ihrer aktuellen Vorlagen, nutzen Sie einfach unser [Anfrageformular](#).

3D Modelle im Web mit three.js



<https://blog.r23.de/3d-modelle-im-web-mit-three-js/>

Besuchen Sie uns auch auf [Facebook](#) und [Twitter](#).

r23
Thüringenstr. 20
58135 Hagen
Deutschland
Telefon: 02331 / 9 23 21 29

E-Mail: info@r23.de

Ust-IdNr.:DE250502477