

WebGPU Shader-Generator für PBR-Material

Studiengang: BSc in Informatik
Vertiefung: Computer Perception and Virtual Reality
Betreuer: Prof. Marcus Hudritsch
Experte: Dr. Harald Studer

OpenGL, eine weit verbreitete Grafik-API, deren erste Version im Jahre 1992 erschienen ist, wird nicht mehr weiterentwickelt. Durch die alte Architektur der API ist diese nicht für die modernen Grafikkarten geeignet und hat keine Zukunft mehr. Mehrere moderne Grafik-APIs (z.B. Vulkan, Metal, DirectX12) versuchen, die Lücke, welche OpenGL in nächster Zukunft hinterlassen wird, zu füllen. Ein Anwärter für diese Nachfolge ist WebGPU.

Einleitung

WebGPU wurde ursprünglich als Ersatz für WebGL im Webbrowser entwickelt, bietet jedoch auch eine native Implementation für Plattformen an. Die Architektur von WebGPU basiert darauf, die native Grafik-API des unterliegenden Systems (OS) zu nutzen, um mit der Grafikkarte zu kommunizieren. Somit dient WebGPU als ein Abstraktions-Layer für moderne native Grafik-APIs wie Vulkan, Metal und DirectX12. Diese Architektur hat den Vorteil, dass sie plattform-unabhängig ist. Weiter ist die Nutzung der WebGPU-API wesentlich weniger komplex im Vergleich zu tieferliegenden modernen Grafik-APIs wie Vulkan. Allgemein ist es jedoch für eine 3D-Applikation nicht optimal, eine Grafik-API direkt zu verwenden, da für das Zeichnen einer einfachen Box bereits mehrere Ressourcen (Buffer, Texturen usw.) erstellt und verwaltet werden müssen. Eine 3D-Applikation will diese Box nur als ein Objekt sehen und rendern können, ohne mit den dahinterliegenden Ressourcen der Grafik-API zu hantieren. Um dies zu ermöglichen, gibt es Scene-Graph-Libraries (oder Game-Engines).



Damaged Helm (öffentliches glTF-Modell) mit Image-based lighting (IBL)

Ziele

Das Ziel dieser Bachelor-Thesis war es, eine Scene-Graph-Library für WebGPU zu entwickeln. Mit dieser Library soll es möglich sein, eine hierarchische Struktur abzubilden und Objekte in einem Raum zu rendern. Zusätzlich soll die Scene-Graph-Library PBR-Materialien (Physically Based Rendering) unterstützen. Der Shader-Code, welcher benötigt wird, um die PRB-Materialien zu rendern und die Lichtberechnungen durchzuführen, soll dynamisch generiert werden. Dies ermöglicht es, dass der Shader-Code auf die nötigen Eigenschaften des Materials zugeschnitten wird, ohne dass für jede Kombination einer Materialeigenschaft ein eigener Shader-Code von Hand geschrieben werden muss. Allgemein soll sich diese Bachelor-Thesis möglichst an den offiziellen glTF-Standard halten.

Ergebnisse

Es wurde eine Scene-Graph-Library aufbauend auf WebGPU entwickelt, welche die gesetzten Ziele dieser Bachelor-Thesis erfüllt. Es ist möglich, eine Szene hierarchisch aufzubauen und diese zu verwalten. Die Scene-Graph-Library enthält ein Shader-Generator, welcher dynamisch Shader-Code zusammensetzt und darauf zugeschnittene Pipelines erstellt. Dieser Shader-Generator unterstützt zum Abschluss dieser Thesis folgende 3D-Computer-Grafik-Techniken:

- Textur-Mapping (Albedo-Textur)
- Normal-Mapping
- Roughness / Metallic (Cook-Torrance)
- Roughness- / Metallic-Mapping
- Ambient-Occlusion Mapping
- Emission Mapping
- Clearcoat-Mapping
- Image-based lighting
- Alpha-Blending
- Direkte Lichtquellen (in dynamischer Liste)



Kilian Wampfler
kilian.wampfler@gmx.ch