Karlsruhe Institute of Technology

Aufbau einer modernen Rendering-Pipeline

Proseminar-Ausarbeitung von

Jonas Heinle

An der Fakultät für Informatik
Institut für Visualisierung und Datenanalyse,
Lehrstuhl für Computergrafik

9. Mai 2019

Inhaltsverzeichnis

1. Abstract

In unser heutigen hoch technologisierten Welt steigt stetig und rasant die Leistungsfähigkeit moderner Grafikhardware. Um heutzutage Grafik auf einem modernen Computer darzustellen sind mittlerweile eine Vielzahl von Zwischenschritten nötig. Diese Arbeit beschäftigt sich um die einzelnen Stufen dieser Kette, deren jeweilige Aufgabe, ihren Datentransfer untereinander, ihre Reihenfolge sowie ihren Arbeitskontext. Beim Arbeitskontext werden unter anderem die unterschiedlichen Koordinatensysteme untersucht. Dabei soll nicht nur konkret auf die Arbeitsweise der einzelnen Stufe eingegangen werden, sondern auch im Speziellen auf die Zusammenarbeit und Kommunikation. Exemplarische Fragestellungen die behandelt werden sind Folgende: Können in dieser Art der Abarbeitung Flaschenhälse entstehen und wie werden Sie umgangen bzw. bekämpft. Welchen Einfluss hat der Programmierer auf die Pipeline bzw. welche Schritte kann er selber implementieren und welche Schritte werden rein von der Hardware übernommen und können nicht von ihm modifiziert werden. Diese Arbeit macht den Aufbau einer modernen Rendering-Pipeline verständlich und vermittelt den Begriff Rasterisierung Zusätzlich wird es bei dieser Abhandlung, dank des stetigen technologischen Fortschritts, ein Ausblick auf zukünftige Entwicklungen und Neuerungen gegeben, die zukünftig moderne Rendering-Pipelines beherrschen wird.

2. Einleitung

3. Moderne Rendering-Pipeline

3.1 Anwendung

...

3.2 Geometrie

...

3.2.1 Vertex Shader

...

3.2.2 Primitive Assembly

...

3.2.3 Tessellation

...

3.2.4 Geometry Shader

...

3.2.5 Clipping

...

3.2.6 Viewport Transform

...

3.3 Rasterization

...

3.4 Fragment Shader

...

3.5 Per Fragment Operations

...

3.5.1 Multisample Fragment Ops

...

3.5.2 Stencil Test

...

3.5.3 Occlusion Query

...

3.5.4 Blending

...

3.5.5 Logical Operations

...

3.6 Framebuffer und Buffer Objekte

...

3.7 GPU Memory

...

3.8 Compute Shader

...

4. Unterschied moderne und klassische Rendering-Pipeline

4.1 Freie Programmierung oder reines Konfigurieren

...

4.2 Vertex Arrays, Index Buffers, ..

...

5. Ausblick

5.1 Raytracing Unterstützung

...

Literaturverzeichnis

[Hil18] Tomas Akenine Möller; Eric Haines; Naty Hoffman; Angelo Pesce; Michał Iwanicki; Sébastien Hillaire: *Real-time rendering*, Band 4. 2018.

[Nat08] Akenine Möller Tomas; Haines Eric; Hoffman Nathaniel: *Real-time rendering*, Band 3, Seiten 11–28. 2008.

[Ste09] Shirley Peter; Marschner Steve: *Fundamentals of computer graphics*, Band 3, Seiten 161–184. 2009.

Erklärung

Ich versichere, dass ich die Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet habe. Die Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt und von dieser als Teil einer Prüfungsleistung angenommen.

Karlsruhe, den 9. Mai 2019

(Jonas Heinle)