

Workshop

Raytracing 101

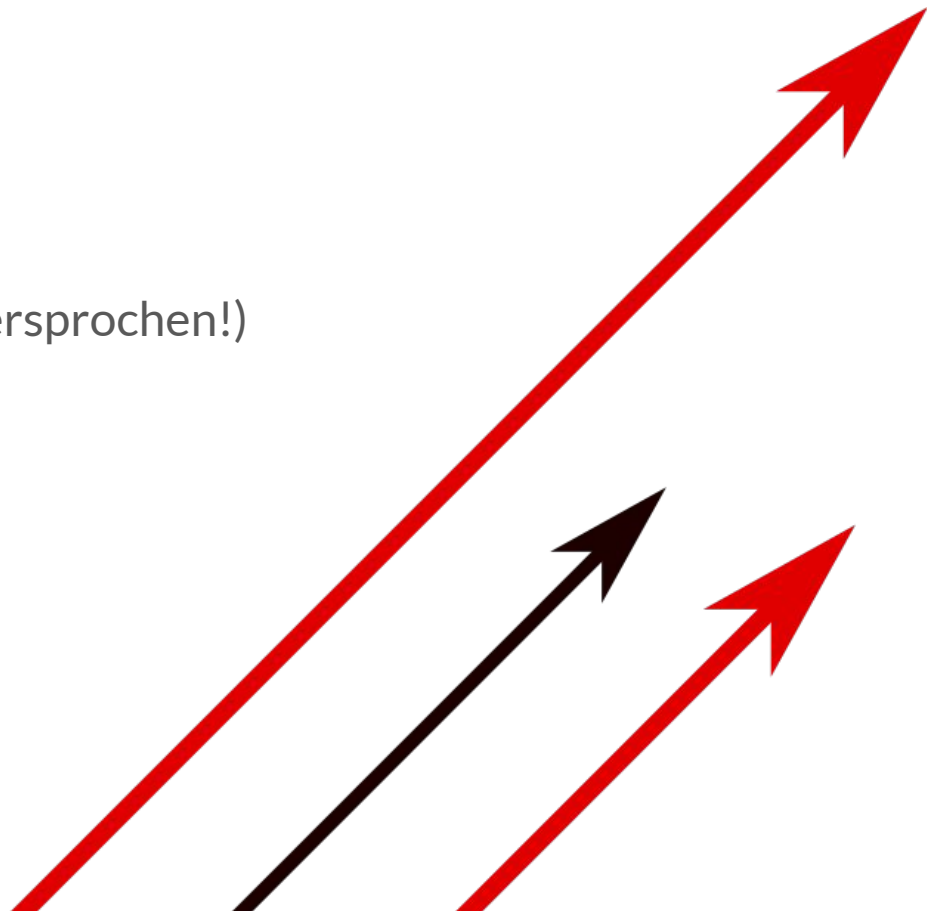
von

Frederik Krumm



Übersicht

1. Kurzvorstellung
2. Was ist Raytracing?
3. RT-Quiz
4. Die Raytracing-Geschichte (nur kurz versprochen!)
5. Ihr schreibt einen Raytracer
6. Diskussion

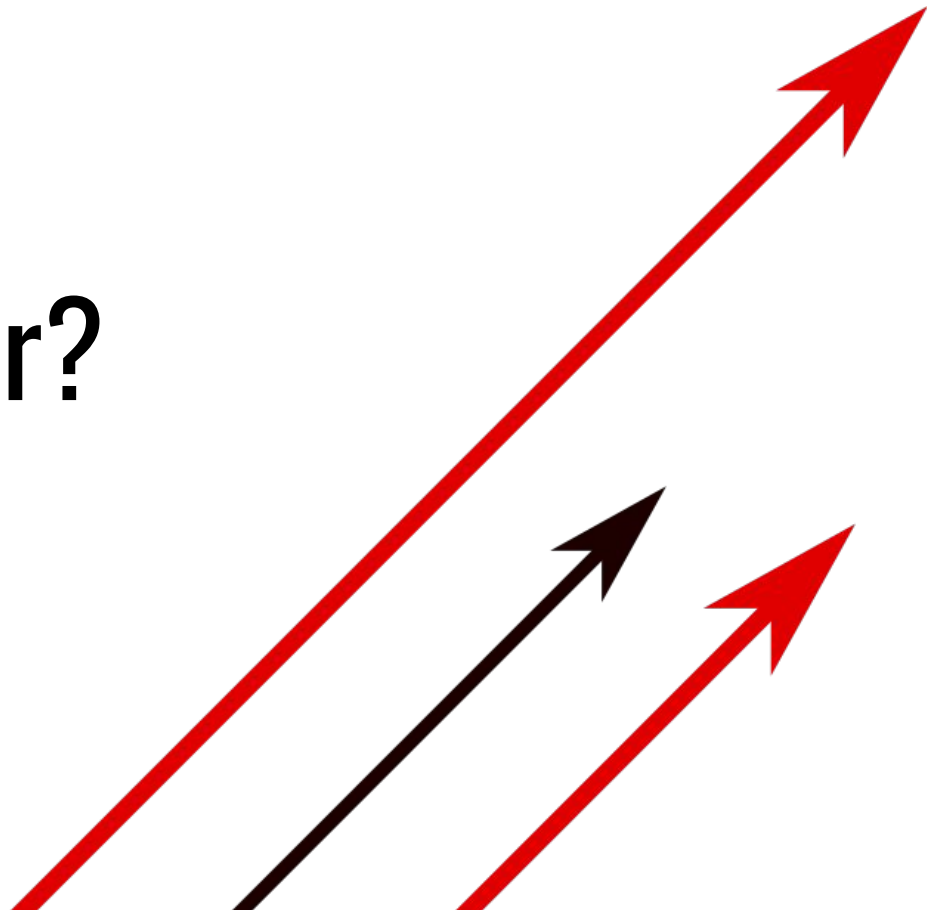


Wer bin ich?

- Studiere im Master Visual Computing & Games Technology
- Arbeite an der HS für das Institute of Visual Computing im Bereich Computergrafik
- Schwerpunkte: Light Transport & Neural Rendering

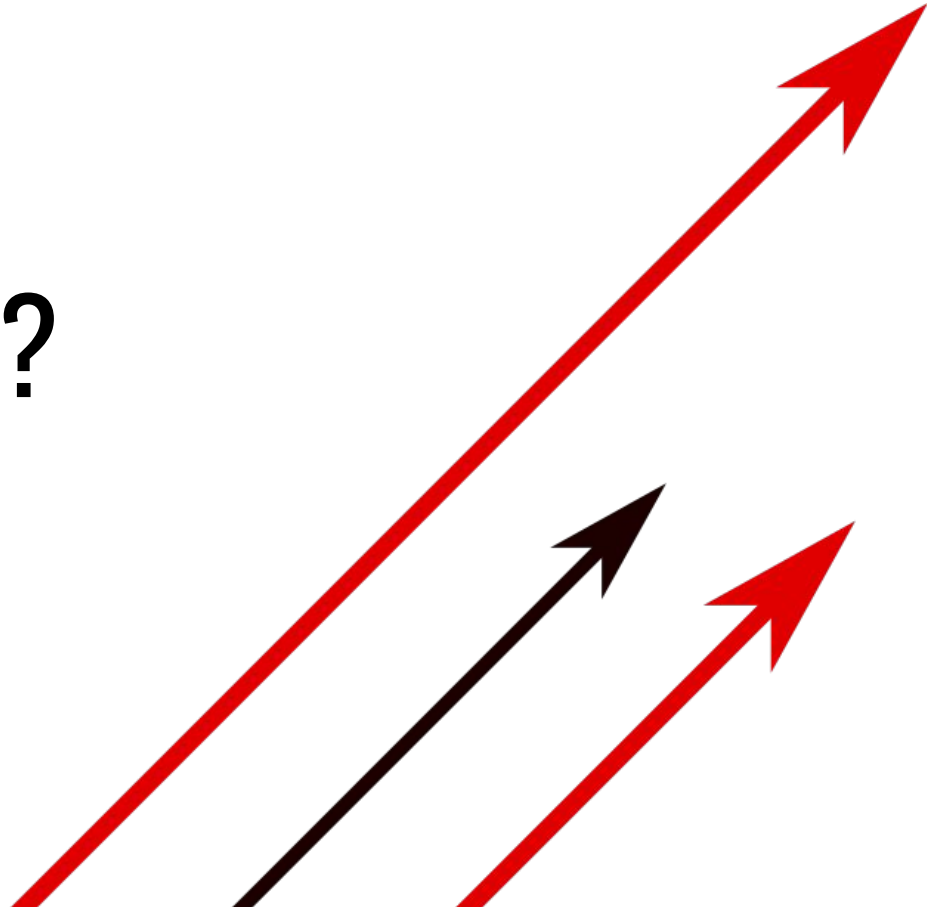


Warum seid ihr hier?



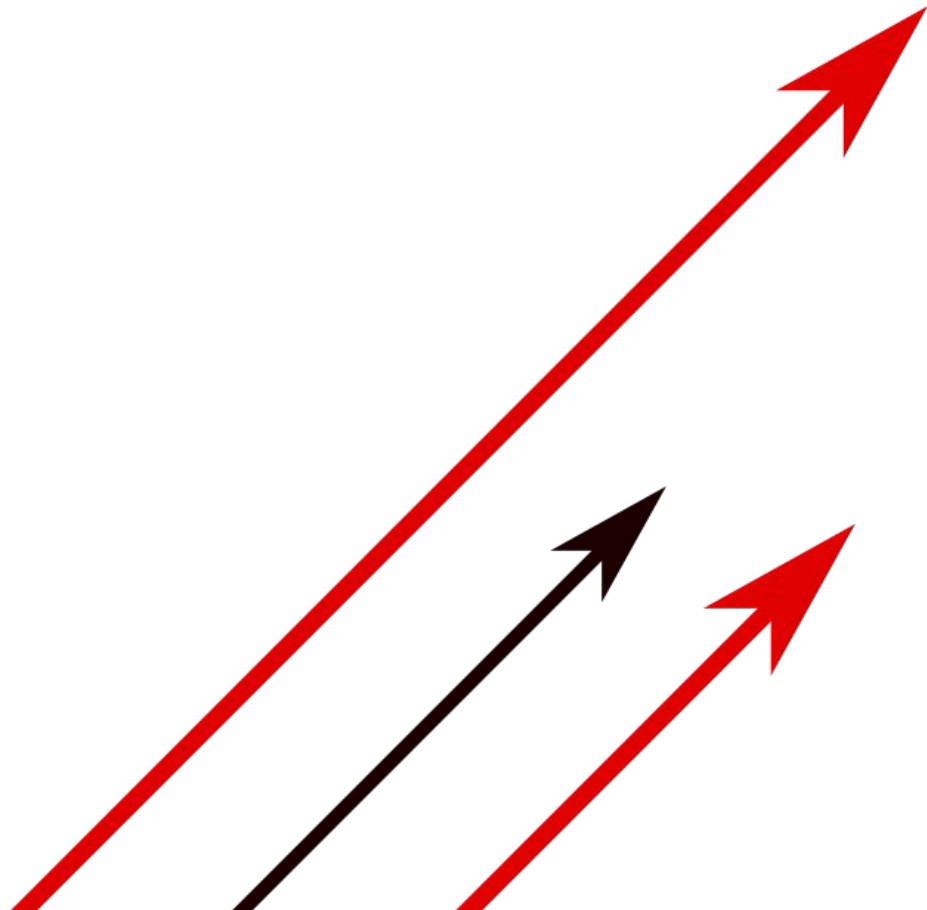


Was ist Raytracing?

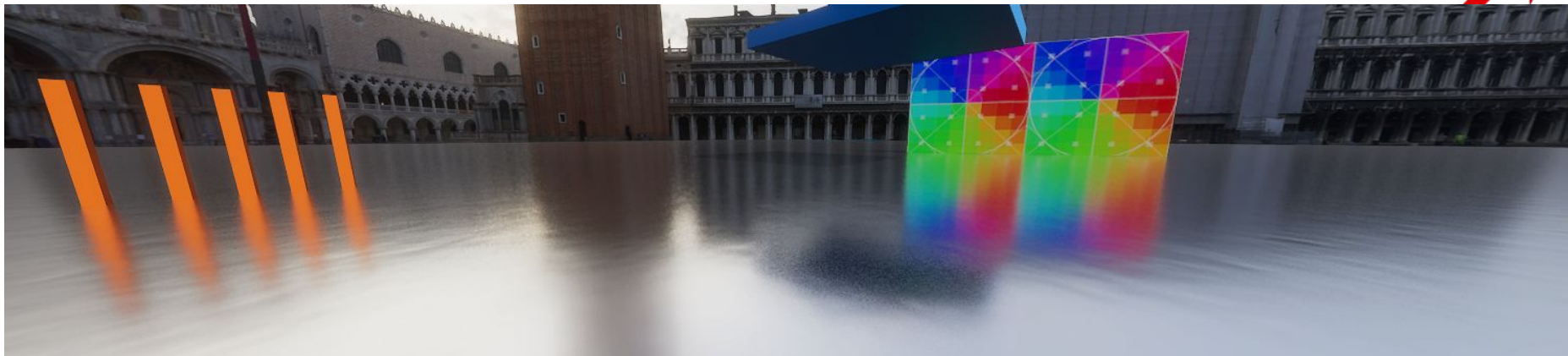


RT-Quiz

Nein, es gibt keine Preise!



1



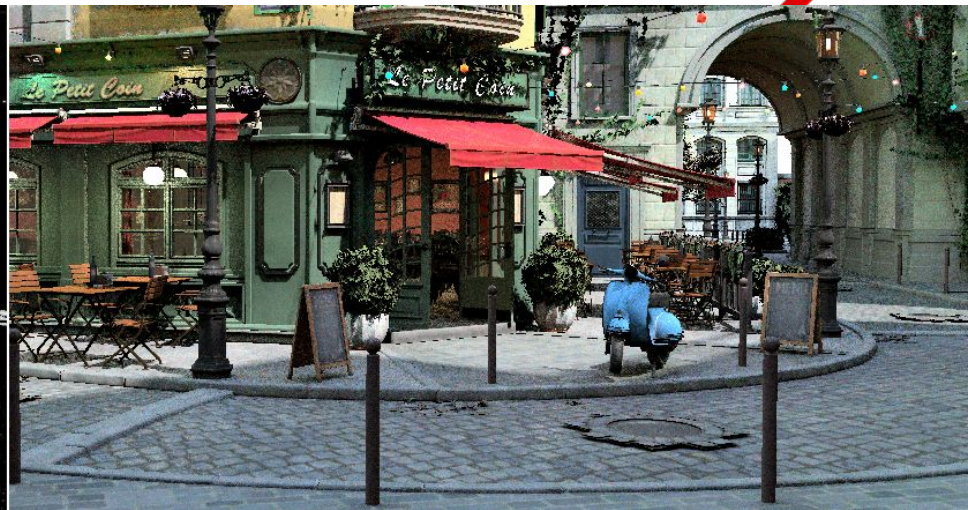
2



3



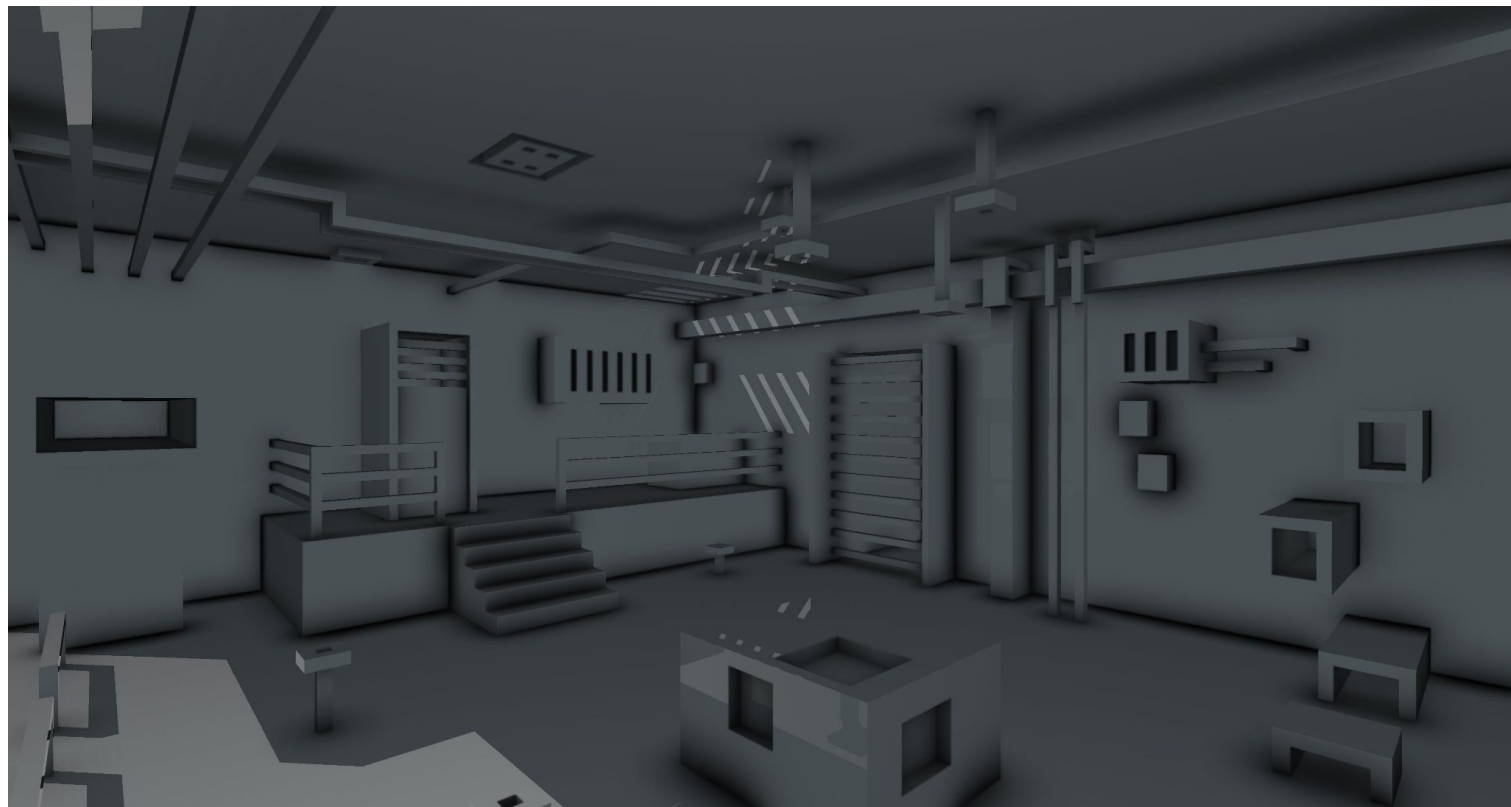
3 - Beispiel



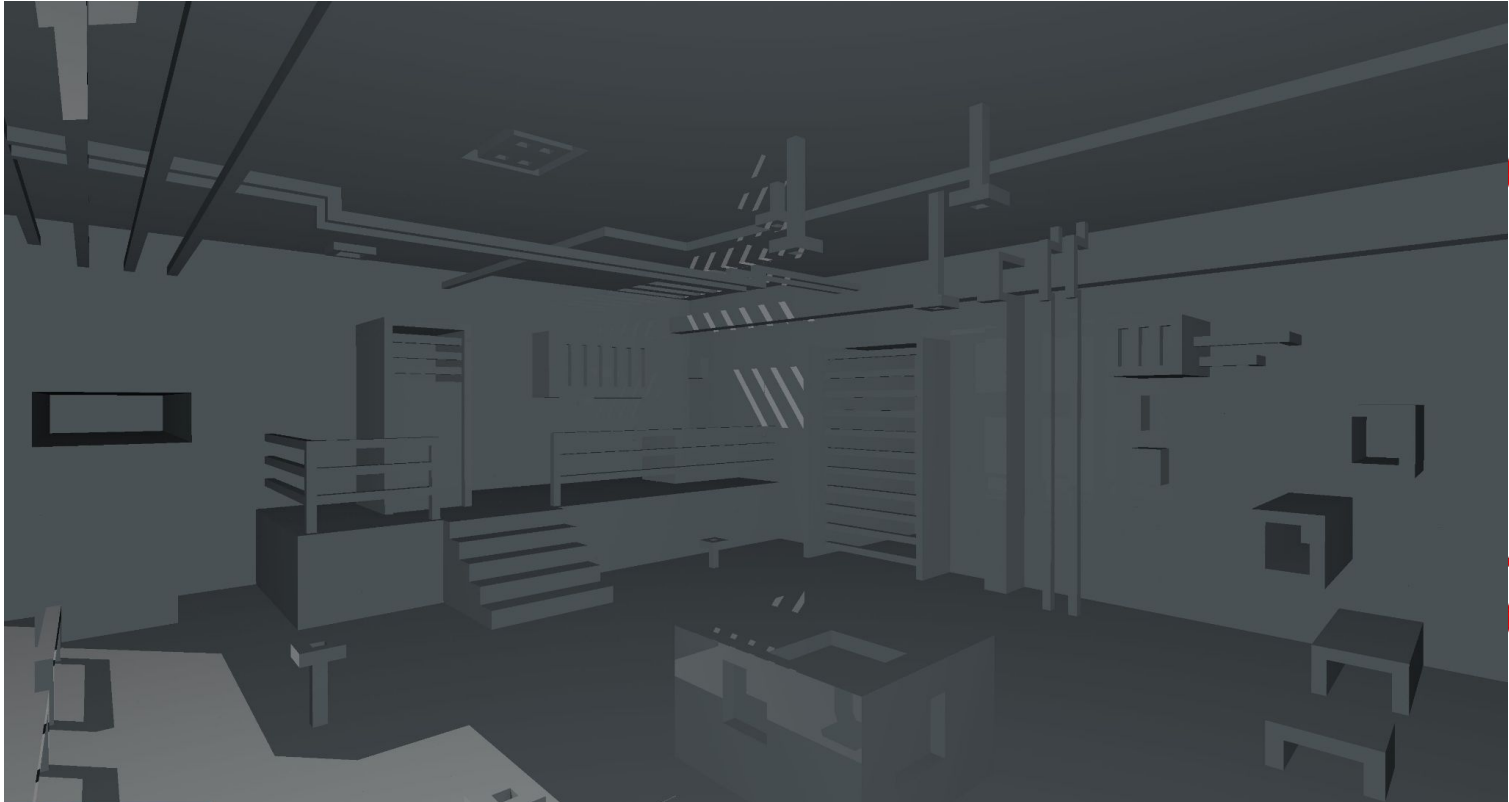
4



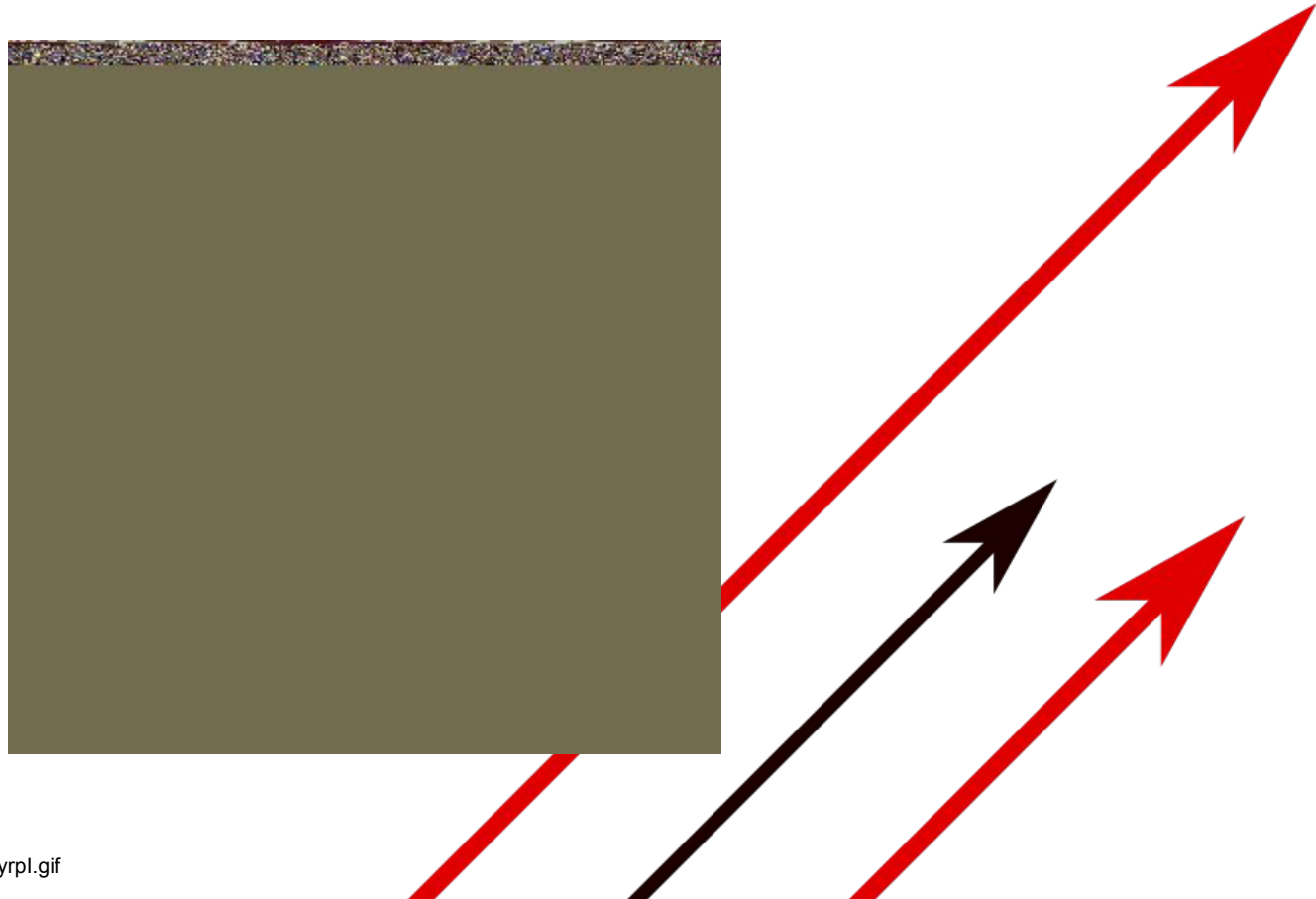
5



6



7



7



Die Geschichte von Raytracing/Raycasting



Kurzgeschichte

- Das Konzept der Beschreibung von Licht durch Strahlen ist schon von Albrecht Dürer (16. Jhdt.) für die Bildsynthese genutzt worden
- In der Computergrafik wurde es erstmals von Arthur Appel 1968 eingesetzt [1]
- Bereits 1979 wurden komplexe Lichtphänomene durch Raytracing beschrieben [2] und schon in den 80ern wurden erste interaktive Raytracer entwickelt
- In “Offline” Methoden (z.B. bei Pixar, Blender Cycles) von lange Standard, in Games erst wirklich vermehrt seit dem RTX Hype 2018 [3]

[1] A. Appel, “Some techniques for shading machine renderings of solids,” in Proceedings of the April 30–May 2, 1968, spring joint computer conference, 1968, pp. 37–45.

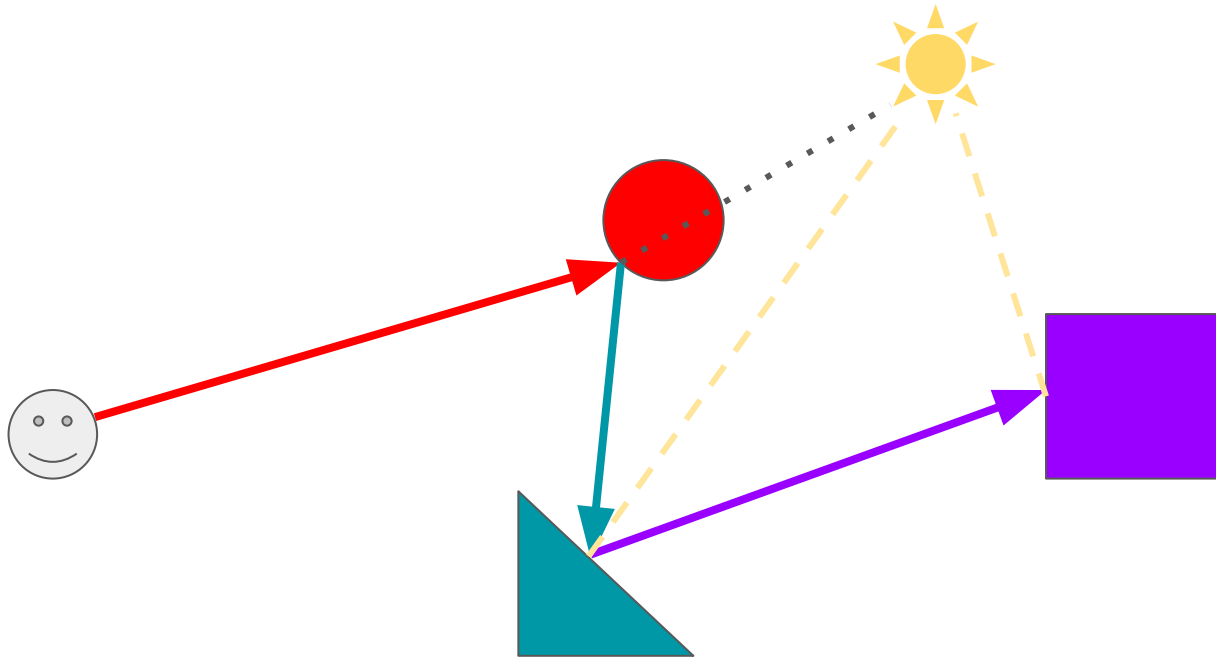
[2] Whitted T. (1979) An Improved Illumination Model for Shaded Display. Proceedings of the 6th annual conference on Computer graphics and interactive techniques

[3] N. Oh. “Nvidia announces rtx technology: Real time ray tracing acceleration for volta gpus and later.” (Mar. 2018), [Online]. Available: <https://www.anandtech.com/show/12546/nvidia-unveils-rtx-technology-real-time-ray-tracing-acceleration-for-volta-gpus-and-later> (visited on 08/25/2022).

**Genug Geschichte,
lasst uns programmieren!**



Was sollen wir denn programmieren?

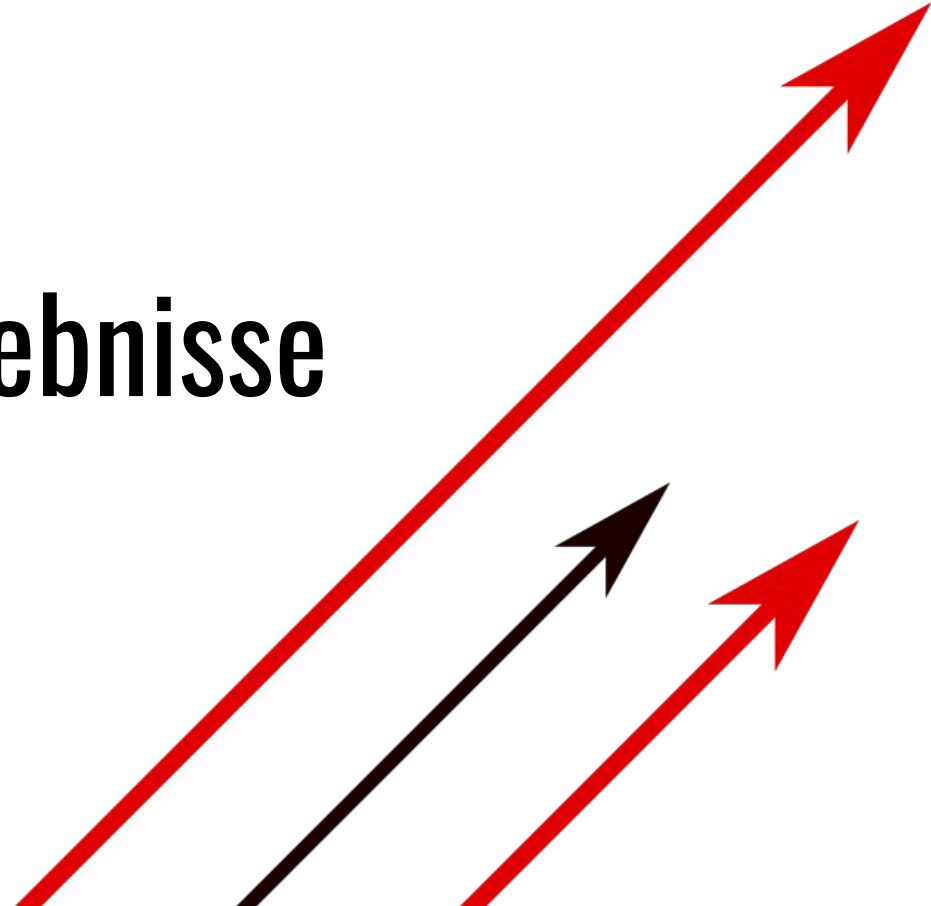


Die Aufgabe

- In Teams von 2-3 Leuten für die, die Bock haben, Rest zu mir ins Plenum
- Schreibt in Pseudocode einen Raytracer
- Der Raytracer soll in der Lage sein **Reflektionen mit einer Tiefe von $n = 3$** darzustellen
- Zusätzlich kann der Raytracer sagen ob er **Licht empfängt** (von einer Lichtquelle)
- Das Ergebnis des Algorithmus ist die beleuchtete Darstellung der Oberfläche mit Reflektionen
- Ihr habt **20 Minuten** Zeit (es ist nicht schlimm, wenn ihr nicht fertig werdet!)



Diskussion der Ergebnisse



Weitere Ressourcen

1. Raytracing Tutorial (C++):
<https://raytracing.github.io>
2. YouTuber mit interessanten Inhalten (Englisch):
<https://www.youtube.com/c/TheChernoProject>

