

Simulation von Lichtszenen in Game Engines für Virtual Reality

Silke Leontopoulos, M. Sc., Dr. Martine Knoop

Technische Universität Berlin, Fachgebiet Lichttechnik

s.leontopoulos@tu-berlin.de

Abstract

Tageslicht auf einer Virtual Reality-Brille könnte in der Zukunft eine Alternative für die Behandlung immobiler Patienten oder die Durchführung lichttechnischer Probandenversuche darstellen. Während die bisherigen Studien zur Licht- und Raumwirkung von Virtual Reality (VR) vorrangig 360°-Panoramen aus lichttechnischen Simulationsprogrammen oder HDR-Kameraaufnahmen verwendet haben, wird beim Einsatz 3D-modellierter Szenen, in denen der Proband mit seiner natürlichen Bewegung navigieren kann, die Lichtsimulation in der Game Engine durchgeführt.

Um eine Einschätzung der daraus entstehenden Limitationen zu geben, werden in diesem Paper zunächst die Voraussetzungen für eine akkurate lichttechnische Visualisierung in Game Engines definiert. Dabei wird vor allem auf die Anforderungen an Lichtquellen sowie Materialien eingegangen. Auf Basis, unter anderem, der CIE 171:2006 werden weiterführende Vorschläge zur detaillierten Validierung von Lichtsimulationen in Game Engines gemacht.

1. Einleitung

Mit wachsender Urbanisierung und immer tieferer Verbauung wird der Zugang zu Tageslicht erschwert. Menschen verbringen mehr Zeit in Innenräumen und künstlicher Beleuchtung. Während es für mobile Menschen immer zu empfehlen ist, nach draußen zu gehen, ist dies für eine steigende Zahl von alten Menschen in Pflegeheimen nicht immer möglich. Dabei würden gerade diese Menschen von den positiven Eigenschaften von Tageslicht, wie der Stabilisierung des Tag-Nacht-Rhythmus oder den stressreduzierenden Effekten von Aussicht profitieren. Eine Alternative wäre es hier, Tageslicht-ähnliches Licht und Ausblick über eine Virtual Reality-Brille anzubieten.

© 2021 by the authors. – Licensee Technische Universität Ilmenau, Deutschland.



This is an **Open Access** article distributed under the terms of the [Creative Commons Attribution ShareAlike-4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/), (<https://creativecommons.org/licenses/by-sa/4.0/>).

Ein zweiter Anwendungsfall ist die Verwendung von Virtual Reality (VR) für lichttechnische Probandenversuche, die zwar nicht die gleiche Relevanz im gesellschaftlichen Maße haben, aber die Forschung vereinfachen und beschleunigen könnten. Insbesondere im Bereich Tageslicht, wo auf Grund des sich ständig verändernden Himmels üblicherweise mit künstlichen Lichtquellen gearbeitet werden muss, würde die Verwendung einer virtuellen Szene eindeutige Vorteile bringen.

Bis die Wirkung von Tageslicht auf einer VR-Brille für Patienten mit Tageslicht-Mangel umfassend untersucht ist, wird noch viel Forschung nötig sein. Das diskrete RGB-Spektrum der Displays, limitierte Auflösung und Gesichtsfeld sowie Leuchtdichte¹, Faktoren wie Flicker und Latenz können die Wahrnehmung verfälschen und zu Unwohlsein führen. Der Einfluss dieser Faktoren muss in genauen physiologischen und psychologischen Untersuchungen bestmöglich separiert und bewertet werden.

2. Stand der Forschung

Seit Anfang der 2000er wurde eine Vielzahl an Studien durchgeführt, um die Verwendung virtueller Szenen für Probandenversuche zu evaluieren. Dabei wurde besonderer Fokus auf Raum- und Lichtwirkung gelegt. In Cauwerts (2013) konnten im Gegensatz zu 2D- und 3D-Bildern nur mit 360°-Panoramen auf einem 2D-Monitor die Parameter *light distribution* und *pleasantness* für alle dargestellten Szenen reproduziert werden. Mit der Kommerzialisierung und zunehmenden Verbreitung von VR-Technologie wurden solche Vergleiche auch zwischen realen Räumen und Simulationen in der VR durchgeführt. Die Untersuchungen von Chen et al. (2019) und HigueraTrujillo et al. (2019) zeigten eine deutliche Annäherung der Raumwirkung und emotionalen Wahrnehmung an die echte Szene für 360°-Panoramen auf einer VR-Brille im Vergleich zu zweidimensionalen Bildern oder Videos. Chamilothoni et al. (2018) und Abd-Alhamid et al. (2019) haben echte und VR-Szenen verglichen und dabei eine gute Übereinstimmung der Raumwirkung bzw. bei letzterer Studie auch der Lichtwirkung festgestellt.

In fast allen Studien wurden die virtuellen Szenen als 360°-Panoramen erstellt, die eine fixe Position des Probanden erfordern und schon bei Verkipfung des Kopfes nicht mehr akkurat sind. Zudem sind diese Panoramen immer für eine feste Körpergröße erstellt, weshalb sich bei Probanden oftmals das Gefühl zu schweben eingestellt hat. Einzig HigueraTrujillo et al. (2019) hat eine 3D-modellierte Szene erstellt, in der sich Probanden jedoch mithilfe eines Joysticks bewegen mussten. Im Vergleich zur

¹ Das Display des am Fachgebiet verwendeten VR-Headsets besitzt z.B. nur eine maximale Leuchtdichte von $140 \frac{cd}{m^2}$.

360°-Variante gab es eine Annäherung des Bio-Feedbacks (Herzratenvariabilität und elektrodermale Aktivität) zur echten Szene, jedoch in nicht der subjektiven Wahrnehmung. 3D-modellierte Szenen, in denen der Betrachter durch Bewegung seines Körpers navigieren kann, wurden bisher noch nicht bewertet. Dabei würde gerade diese Immersion den Betrachtungsmöglichkeiten eines echten Tageslicht-Erlebnisses am nächsten kommen und somit das Potential der Plattform VR voll ausnutzen.

Zur Darstellung auf einer VR-Brille muss eine Ansteuerungssoftware mit Echtzeit-Rendering eingesetzt werden. Aus diesem Grund werden Game Engines, wie Unreal und Unity, verwendet. Für 360°-Panoramen werden üblicherweise Kameraaufnahmen oder gerenderte Bilder aus einer lichttechnischen Simulationssoftware (wie Radiance) zu einem Panorama zusammengerechnet und das Ergebnis in der Game Engine visualisiert. Im Gegensatz dazu muss für eine 3D-modellierte Szene die gesamte Beleuchtung in der Game Engine vorgenommen werden.

Dies bedeutet zusammengefasst: Die Qualität der lichttechnischen Simulation in den 3D-modellierten Szenen hängt von den Beleuchtungsmodellen der Game Engine ab. Unterster Baustein für den Einsatz dieser echt-immersiven Lichtszenen ist demnach die Betrachtung der Szenen-Erstellung in Game Engines und eine genaue Charakterisierung der lichttechnischen Simulationsmöglichkeiten dieser.

Neben der Szenenerzeugung spielt auch die Darstellung der Szenen eine große Rolle. Es wurden jedoch nur bei Chamilothon et al. (2018) und Abd-Alhamid et al. (2019) die Leuchtdichten von Punkten im Raum mit den Leuchtdichten des VR-Displays verglichen. In Abd-Alhamid et al. (2019) wurde die Gamma-Kurve des Displays vermessen und Darstellungsparameter entsprechend angepasst, allerdings nicht die verschiedenen Farbkanäle betrachtet. Obwohl bei der Betrachtung der Lichtwirkung neben rein photometrischen Aspekten auch farbmimetrische Aspekte abgefragt wurden, wurden die genauen Werte der Farbörter nicht validiert. Eine Methodik zur umfassenden Charakterisierung und Kalibrierung der Leuchtdichten sowie Farbörter von VR-Brillen ist somit dringend notwendig, um aussagekräftige Vergleiche zwischen virtuellen Szenen und echten Szenen sowie anderen Darstellungsmedien anzustellen.

In diesem Artikel werden nach einer kurzen Einführung in Rendering-Techniken die Voraussetzungen für eine akkurate lichttechnische Visualisierung definiert. Im Anschluss an die Darstellung der Visualisierungsverfahren von Game Engines² werden Vorschläge zur Validierung von Lichtsimulationen in Game Engines sowie zur Kalibrierung gemacht.

² Die Angaben zu den Game Engines beziehen sich auf die Versionen Unreal 4 bzw. Unity 2021.

3. Überblick Rendering

Das Rendering eines Bildes kann in zwei Schritte aufgeteilt werden: Im Ersten wird überprüft, welche Objekte überhaupt im Bild sichtbar sind (Verdeckungsrechnung), im Zweiten werden die Farben der einzelnen Pixel durch das Zusammenspiel von Oberflächen und Beleuchtung bestimmt (Shading).

Ist das Ziel, fotorealistische Bilder zu generieren, wie bei dem in Game Engines verwendeten *Physically based Rendering*, basiert das Beleuchtungsmodell auf der Lösung der Rendergleichung, wie von Kajiya (1986) eingeführt. Die Darstellung in (1) ist nach Ward und Shakespeare (1998).

$$L_r(\theta_r, \phi_r) = L_e + \iint L_i(\theta_i, \phi_i) f_r(\theta_i, \phi_i, \theta_r, \phi_r) |\cos \theta_i| \sin \theta_i d\theta_i d\phi_i \quad (1)$$

Sie besagt, dass ausgesandte Strahlung an einem Punkt L_r gleich der Summe der emittierten Strahlung L_e (Primärstrahler) und der gesamten von der Fläche reflektierten Strahlung (Sekundärstrahler) von diesem Punkt ist. Die reflektierte Strahlung ist dabei aus der empfangenen Strahlung $L_i(\theta_i, \phi_i)$ und der bidirektionalen Reflexionsverteilungsfunktion³ $f_r(\theta_i, \phi_i, \theta_r, \phi_r)$ zusammengesetzt, welche in Unterkapitel 4.2 genauer betrachtet wird.

Eine in fotorealistischen Renderern häufig eingesetzte Technik ist das Raytracing. Für die Beleuchtungssimulation werden Strahlen hierbei entweder von der Quelle zum Oberflächenpunkt verfolgt (Forward-Raytracing) oder vom Oberflächenpunkt zur Quelle (Backward-Raytracing). Eine Implementation von Forward-Raytracing ist deutlich ineffizienter, da die meisten Strahlen außerhalb des Bildpunktes auftreffen und es schwierig vorauszusehen ist, wann alle Bildpunkte mit Informationen versehen sind. Aus diesem Grund ist das Backward-Raytracing die weiter verbreitete Variante.

Soll das Rendering in Echtzeit stattfinden, wird stattdessen Rasterisierung eingesetzt. Hierbei werden alle Elemente in Polygone aufgebrochen und ein Objektpunkt P mit Hilfe einer Projektionsmatrix auf einen Bildpunkt P' umgerechnet. Da bei der Rasterisierung immer nur ein Polygon auf einmal bearbeitet wird, müssen Effekte wie Schatten und Reflexionen, die durch Wechselwirkungen mehrerer Objekte entstehen, in einem zusätzlichen Schritt ergänzt werden.

Insgesamt bringt Raytracing visuell bessere Ergebnisse beim Shading, jedoch zum Preis höherer Berechnungskosten.

³ Im englischen Sprachgebrauch BRDF für *Bidirectional Reflectance Distribution Function* genannt.

In Unity und Unreal wird Rasterisierung verwendet, auch wenn es mit einer kompatiblen Grafikkarte auch die Möglichkeit gibt, Realtime-Backward-Raytracing zu verwenden. Auch Radiance arbeitet mit Backwards-Raytracing; zusätzlich gibt es eine Photon Mapping-Erweiterung, die die Software um Forward-Raytracing ergänzt.

4. Anforderungen für lichttechnisch und farbmétrisch akkurate Visualisierungen

4.1 Abbildung von Primärstrahlern

Die Anforderungen für eine fotorealistische Beleuchtung beginnen bei der Lichtquelle, dem Primärstrahler. Die Angabe der emittierten Strahlung muss einer physikalischen Größe folgen. Hierbei eignet sich je nach Anwendung eine quellenbezogene photometrische Größe, wie der Lichtstrom oder die Leuchtdichte. Unity in der *High Definition Rendering Pipeline (HDRP)* sowie Unreal unterstützen unterschiedliche Größen, je nach ausgewählter Standardquelle (siehe Tab. 1).

Quellen-Typ	Unity	Unreal
Point Light	Lichtstrom, Lichtstärke, Beleuchtungsstärke	Lichtstrom, Lichtstärke
Spot Light	Lichtstrom, Lichtstärke, Beleuchtungsstärke	Lichtstrom, Lichtstärke
Area Light/Rect Light	Lichtstrom, Leuchtdichte	Lichtstrom, Lichtstärke
Directional Light	Beleuchtungsstärke	Beleuchtungsstärke

Tab. 1: Photometrische Größen in Unity und Unreal

In beiden Programmen haben sich die gleichen vier Standardquellen etabliert:

- Das *Point Light* ist eine Punktlichtquelle, die in alle Richtungen abstrahlt.
- Das *Spot Light* strahlt Licht in einem festgelegten Winkel aus, der durch den *Spot Angle* definiert wird. Die Lichtstärke fällt zu den Rändern der Form ab. Die Form kann in Unreal ein Konus, in Unity hingegen auch eine Pyramide oder eine Box sein.
- Das *Area Light* ist ein flächenförmiger Lambertstrahler in Unity, der die Formen Rechteck, Röhre oder Scheibe annehmen kann. In Unreal heißt die vergleichbare Lichtquelle *Rect Light* und muss eine rechteckige Form besitzen.
- Das *Directional Light* ist eine unendlich breit ausgedehnte Flächenquelle im Unendlichen, die den Einfall von parallelen Strahlen simuliert. Da die Quellen-Position nicht erkennbar ist, kann dieses Licht überall in der Szene positioniert

werden. Die Beleuchtungsstärke auf Flächen ändert sich nicht, wenn der Abstand zur Quelle verändert wird. Diese Lichtquelle wird vor allem dafür eingesetzt, um gerichtetes Sonnenlicht zu simulieren.

Diese vier Quellentypen vertreten unterschiedliche Abstrahlcharakteristika, die in der Lichttechnik korrekterweise durch Lichtstärkeverteilungskurven (LVK) definiert werden. Sowohl Unreal als auch Unity erlauben einen Import von LVKs im IES-Format. Zusätzlich ist auch eine korrekte physikalische Ausbreitung des Lichts über die Distanz entscheidend. Da eine unendlich weit strahlende Lichtquelle die Berechnung einer Szene sehr aufwendig macht, wird die Beleuchtungsstärke mit Hilfe von Abklingformeln ab einer gewissen Distanz auf null gesetzt. Der Wert für diese Distanz lässt sich definieren.

Neben der Verwendung von photometrischen Größen und der korrekten Lichtverteilung ist für eine realistische Emission zudem die Beschreibung der Lichtfarbe wichtig, die sich in der Realität aus der spektralen Verteilung des abgestrahlten Lichts ergibt. Hier gibt es mehrere Möglichkeiten, Information in der Simulationssoftware zu hinterlegen: Da der Import von spektral aufgelösten Werten auch in lichttechnischen Simulationstools in der Regel nicht möglich ist, kann an dieser Stelle stattdessen mit einem Farbort oder einer Farbtemperatur gearbeitet werden. Beide Möglichkeiten sind sowohl in Unity als auch Unreal implementiert.

In Radiance können spektrale Verteilungen von Lichtquellen integriert werden, die allerdings in entsprechende RGB-Farbörter umgerechnet werden. Rubbertsberg und Bloj (2008) beschreiben eine Möglichkeit, spektrale Messdaten auf 27 Dateien (d.h. 81 Frequenzbänder) aufzuteilen und die Ergebnisse der 27 einzelnen Berechnungen am Ende zu kombinieren.

Ein letzter Faktor neben der emittierten Strahlung ist die Visualisierung der Lichtquelle selbst. In Game Engines werden Primärstrahler nur durch die Bestrahlung anderer Flächen in der Szene sichtbar und besitzen keine emittierende Oberfläche. In der Vergangenheit wurde der visuelle Eindruck durch eine zusätzliche Punktlichtquelle erzeugt, die die weiße Textur der Primärstrahler-Oberfläche beleuchtet. Heutzutage können in beiden Game Engines Materialien als emittierend definiert werden, womit eine abstrahlende Oberfläche simuliert werden kann, die so eingestellt wird, dass sie kein weiteres Licht in die Szene wirft. Die Visualisierung der Oberflächen einer Lichtquelle ist dann besonders wichtig, wenn beispielsweise Effekte wie Blendung untersucht werden sollen.

4.2 Abbildung von Materialeigenschaften

Bei der Ermittlung der Beleuchtung einer Fläche wird immer zwischen dem Direktanteil durch Lichtquellen in der Szene und dem indirekten Anteil durch Reflexion an anderen Oberflächen unterschieden. Das Zustandekommen des indirekten Anteils wird im Bereich der Computergrafik auch als *Global Illumination* bezeichnet und kann gerade bei stark reflektierenden Materialien einen großen Unterschied in den Beleuchtungsniveaus der Szene ausmachen. Je nach Qualität der Berechnung werden Mehrfachreflexionen bis zu einer gewissen Anzahl unterstützt. Unity und Unreal besitzen beide mehrere *Global Illumination*-Modelle, die entweder nur statische oder auch dynamisch veränderliche Lichtquellen während der Ausführung der Szene zulassen⁴.

Die Reflexionen werden in der Computergrafik durch bidirektionale Verteilungsfunktionen beschrieben⁵. Diese können von echten Materialien durch eine bidirektionale Messanlage gewonnen werden oder durch ein analytisches Modell berechnet werden. Die BRDF wird für das analytische Modell in zwei Teile aufgeteilt, einen diffusen, für den meist das Lambert'sche Modell direkt oder in einer Variation verwendet wird, sowie einen spiegelnden Anteil. Spektrale Informationen können als zusätzliche Information in eine BRDF aufgenommen werden, was in der Praxis jedoch selten umgesetzt wird.

Unreal und Unity bieten keine Möglichkeit Messdaten als BRDF zu importieren, sondern nutzen rein analytische Modelle^{6,7} zur Berechnung der Oberflächeneigenschaften. Das *Physically-based-Rendering* der beiden Game Engines verwendet in den Standard-Shadern den *Metallic-Workflow*, bei dem metallische und nicht-metallische Oberflächen separat behandelt werden. Diese Unterscheidung basiert auf der unterschiedlichen Erscheinung von Reflexionen und somit u.a. der wahrgenommenen Oberflächenfarbe. Während sich die Glanzlichter auf einer nicht-metallischen Oberfläche an der Lichtfarbe orientieren, sind diese bei einer metallischen Oberfläche eingefärbt. Die rein diffuse Reflexion einer nicht-metallischen, opaken Oberfläche entspricht in der Regel dem Farbeindruck der Oberfläche, während metallische Oberflächen

⁴ Zum jetzigen Zeitpunkt gibt es keine Implementierung eines dynamischen *Global Illumination*-Modells in der *HDRP* von Unity.

⁵ Analog zur BRDF gibt es die BTDF (*Birectional Transmission Distribution Function*). Die Summe aus beiden ergibt die BSDF (*Bidirectional Scattering Distribution Function*).

⁶ Unity: Abwandlung von Lambert nach Disney (Brent 2021), GGX-Mikrofacetten-Modell nach Walter et al. (2007)

⁷ Unreal: Lambert, GGX-Mikrofacetten-Modell (Karis 2012)

keine diffusen Reflexionen besitzen und der Farbeindruck nur durch die spiegelnden Reflexionen entsteht.

Zur Simulation der Oberflächeneigenschaften werden dem Nutzer vier Parameter zur Verfügung gestellt, um das Aussehen und die Interaktion eines Objekts mit Licht zu spezifizieren: *Base Color/Albedo*, *Roughness/Smoothness*, *Metallic* und *Specular* (nur in Unreal).

- Die Funktion von *Base Color* (Unreal) oder *Albedo* (Unity) unterscheidet sich, je nachdem, ob es sich um eine metallische oder nicht-metallische Oberfläche handelt. Für eine nicht-metallische Oberfläche werden die RGB-Farbwerte der rein diffusen Reflexion hinterlegt. Bei einer metallischen Oberfläche sind in diesem Parameter die RGB-Werte der spiegelnden Reflexion gespeichert. In beiden Fällen wird somit der übergeordnete Farbeindruck durch den Parameter beschrieben.
- *Roughness* (Unreal) oder *Smoothness*⁸ (Unity) bestimmt die Lichtverteilung bei der Reflexion. In den verwendeten Mikrofacetten-Modellen der Game Engines wird eine große Anzahl kleiner, spiegelnder Flächen simuliert, durch deren individuelle Ausrichtung und Abschattung das Licht insgesamt gerichtet oder gestreut abgestrahlt wird. Die Abstraktion auf diese Mikrofacetten erlaubt dabei eine einfachere mathematische Beschreibung als die reale mikroskopisch raue Oberfläche. Als Resultat kann die Lichtverteilung durch diesen Parameter zwischen ideal spiegelnd und ideal diffus gesetzt werden. Je nach Einstellung erscheinen Glanzlichter größer und diffuser bzw. kleiner und schärfer (siehe Abb. 1).
- *Metallic* legt fest, ob es sich um eine metallische oder nicht-metallische Oberfläche handelt. Werte außer 0 und 1 sollten vermieden werden, auch wenn in der Praxis Zwischenwerte z.B. für Staub oder Rost auf Metall eingesetzt werden. Falls die spiegelnden Reflexionen einer nicht-metallischen Oberfläche zusätzlich abgeschwächt oder verstärkt werden sollen, kann in Unreal der Parameter *specular* verwendet werden. Dies ist allerdings bei den allermeisten Materialien nicht notwendig.

⁸ *Smoothness* funktioniert hierbei komplementär zu *Roughness*, d.h. für ein gleiches Ergebnis wie für *Roughness* = 1 muss *Smoothness* auf 0 gesetzt werden.

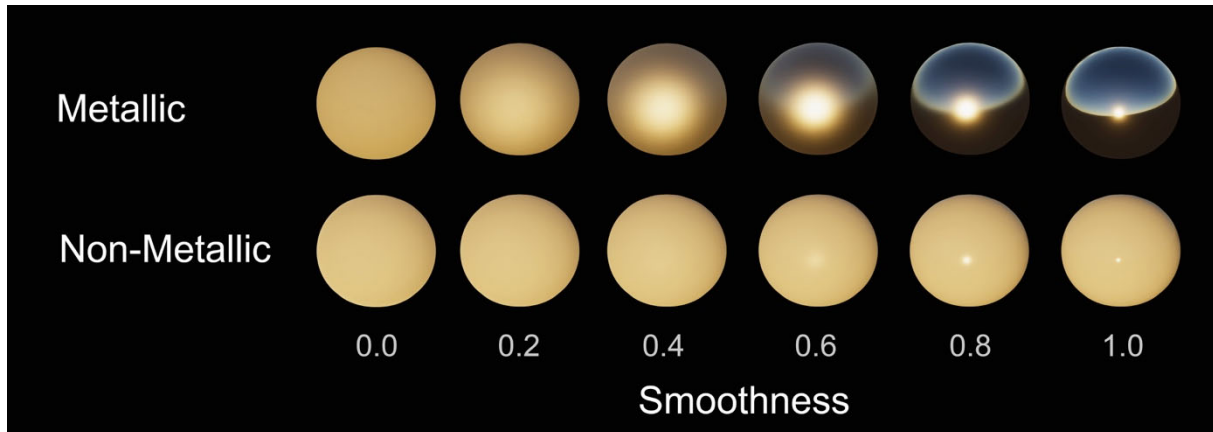


Abb. 1: Verschiedene Smoothness-Einstellungen von 0 – 1 bei einem Metallic-Wert von 1 (oben) und 0 (unten)

Neben opaken isotropen und anisotropen Oberflächen werden auch transluzente und transparente Oberflächen unterstützt. Hier kommen noch weitere Auswahlparameter, wie z.B. die Brechzahl des Materials hinzu.

Genau wie bei den Primärstrahlern werden keine spektral aufgelösten Daten unterstützt. In Abb. 2 kann man in Unity bei aktiver *Global Illumination* trotzdem sehen, wie durch das *Albedo* der Wand die neutrale Oberflächenfarbe der Kugel beeinflusst wird.

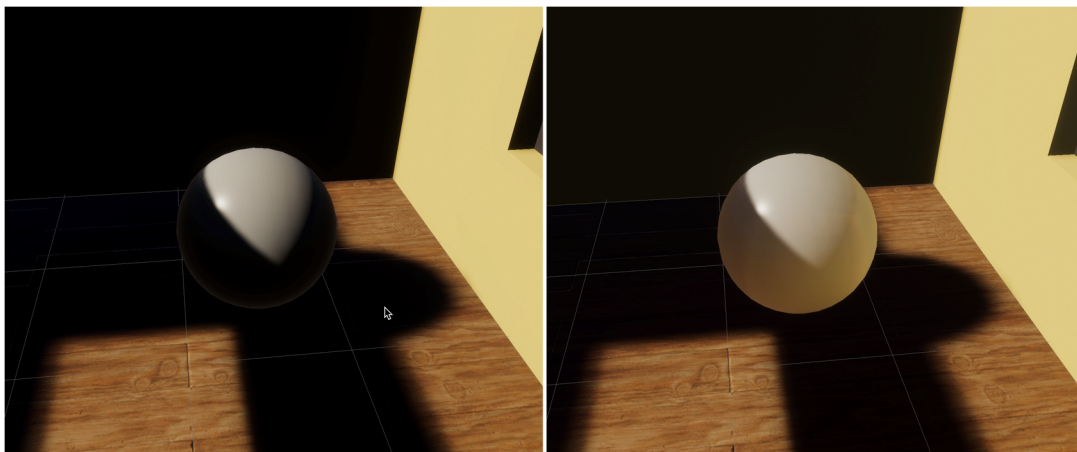


Abb. 2: Ohne *Global Illumination* (links), mit *Global Illumination* (rechts)

Auch Radiance verwendet analytische Modelle⁹ und unterscheidet bei seinen Materialtypen zwischen metallischen- und nicht metallischen Oberflächen, die sowohl in isotropen als auch anisotropen Varianten vorliegen. Im Gegensatz zu den Game Engines gibt es die Möglichkeit Messdaten einer BSDF zu importieren. Auch spektralabhängige

⁹ Lambert und Ward Shading Modell

Reflexionsgrade können eingelesen werden, allerdings werden diese zu einem einzelnen RGB-Farbot umgerechnet. Soll eine spektral aufgelöste Simulation durchgeführt werden, müssen auch die Materialdaten auf verschiedene Frequenzbänder aufgeteilt werden, die dann mit den entsprechenden Bändern der Primärstrahler einzeln nacheinander berechnet werden.

5. Untersuchung der lichttechnischen und farbmtrischen Genauigkeit von Game Engines

Zur Abschätzung der Genauigkeit bei der Visualisierung von Licht-Szenen sollen unterschiedliche Szenarien implementiert und mit Referenzwerten verglichen werden. Da zurzeit noch nicht abgeschätzt werden kann, inwiefern zuverlässige photometrische Werte in den Game Engines gewonnen werden können, ist auch ein Vergleich von RGB-Werten zu einer identischen Implementierung in Radiance möglich.

Der erste Vergleich soll durch die dokumentierten Testszenarien der *CIE 171:2006: – Test Cases to Assess the Accuracy of Lighting Computer Programs* vorgenommen werden. Diese enthalten Szenarien-Typen, deren Referenzwerte auf Messungen oder auf analytischen Berechnungen basieren.

Die erste Kategorie umfasst verschiedene Lichtquellen in einem einfachen Raum mit grauen oder schwarzen Wänden. Die zweite Kategorie bietet Vorschläge für die Simulation von Punktlichtquellen und Flächenleuchten. Zudem werden wichtige Aspekte der Wechselwirkungen zwischen Quelle und Oberflächen im Raum in einzelnen Szenarien betrachtet, so wie die Erhaltung des Lichtstroms, die Transmission durch Glasflächen bzw. die Reflexion an diffusen Flächen. In dieser Kategorie werden auch Testszenarien für Tageslichtszenen vorgestellt, die sich auf die CIE Sky Types beziehen. Diese Himmelsmodelle müssen zur Validierung zunächst in den Game Engines implementiert werden.

Zusätzlich zur Betrachtung der genannten Testszenarien sollten auch die Empfehlungen für weitere Versuche mit diffus-transmittierenden und spiegelnd-reflektierenden Oberflächen umgesetzt werden. Hierfür soll entweder ein Referenzszenario als Messung aufgebaut oder mit einer bereits validierten Simulationssoftware wie Radiance implementiert werden. Zur weiteren Untersuchung richtungsabhängiger Materialien sollen zusätzlich auch Mischformen diffuser und spiegelnder Reflexion, wie sie beispielsweise bei satiniertem Aluminium oder Emaille vorkommen, betrachtet werden.

Veränderungen der spektralen Verteilung und somit des resultierenden Farbortes des Lichts durch stark spektral abhängige Reflexions- und Transmissionsgrade von Materialien werden in der CIE 171:2006 nicht behandelt. Auch hier soll ein

Referenzszenario entweder als Messung aufgebaut werden oder mithilfe der in Unterkapitel 4.1 genannten spektral aufgelösten Simulationen in Radiance erstellt werden.

Alle genannten Untersuchungen sollen sowohl für Rasterisierung als auch für Raytracing durchgeführt werden. Letztere können in den Game Engines auch für VR-Szenen verwendet werden, allerdings ist es selbst mit aktueller Hardware schwierig die benötigten Bildwiederholraten zu erzeugen. Eine Alternative wäre die Berechnung der Beleuchtung und Speicherung dieser in den Oberflächen der Szene (*Lightbaking*), sodass dieser rechenintensive Schritt während der Echtzeit-Wiedergabe entfällt.

6. Korrekte Darstellung auf dem Head-Mounted-Display

Neben den beschriebenen Voraussetzungen für eine Erzeugung von Szenen muss sichergestellt werden, dass diese auch korrekt auf einem Display wiedergegeben werden. Dies betrifft die Leuchtdichten der einzelnen Farbkanäle über die volle Aussteuerung von 0 bis 255, aus denen sich die einzelnen Farbörter mischen.

Zurzeit wird am Fachgebiet ein Messaufbau zur Charakterisierung und Kalibrierung entwickelt, an dem neben Leuchtdichtekamera und Spektroradiometer auch ein konventionelles Displaykalibriergerät in Spektrofotometertechnik eingebunden werden kann. Denn gerade für Forschergruppen außerhalb der Lichttechnik, für deren Forschung die akkurate Reproduktion von Lichtszenen trotzdem wichtig ist, könnte dies eine preisgünstige Alternative darstellen, sollte sich die Kalibrierung als hinreichend akkurat herausstellen.

Im Kontext der Kalibrierung muss auch die limitierte Leuchtdichte betrachtet werden. Wird der native Weißpunkt des VR-Displays (entstehend durch die Wellenlängen der OLEDs) durch eine Kalibrierung verändert, sinkt die maximal erreichbare Leuchtdichte. Ziel dieses Schrittes ist es demnach sowohl eine höchstmögliche Leuchtdichte zu erzielen, als auch einen Zusammenhang zwischen Pixelwerten und Leuchtdichten herzustellen, dass zumindest Leuchtdichten bis $140 \frac{cd}{m^2}$ exakt reproduziert werden können.

7. Fazit und Ausblick

Für die Verwendung von voll-immersiven 3D-modellierten VR-Szenen ist eine ganz neue Form von Content Creation notwendig. Die Möglichkeiten und Limitationen der Lichtsimulation in Echtzeit-fähigen Game Engines wie Unity und Unreal müssen dabei detailliert betrachtet werden. Mit Hilfe der in diesem Paper vorgeschlagenen

Testszenarien, die zum einen auf der CIE 171: 2006 basieren, zum anderen weiterführende Untersuchungen empfehlen, sollen in Zukunft die Einsatzmöglichkeiten von Game Engines definiert und validiert werden.

Quellen

- Abd-Alhamid, F. et al. (2019). "Developing an Innovative Method for Visual Perception Evaluation in a Physical- Based Virtual Environment". In: *Building and Environment* 162, p. 106278. doi: 10.1016/j.buildenv.2019.106278.
- Burley, B. (2012). "Physically-Based Shading at Disney", part of "Practical Physically Based Shading in Film and Game Production". In: *SIGGRAPH 2012 Course Notes*.
- Chamilothori, K., Wienold, J. und Andersen, M. (2018). "Adequacy of Immersive Virtual Reality for the Perception of Daylit Spaces: Comparison of Real and Virtual Environments". In: *LEUKOS* 15.2-3, pp. 203–226. doi: 10.1080/15502724.2017.1404918.
- Cauwerts, C. (2013). "Influence of presentation modes on visual perceptions of daylight spaces". Doktorarbeit. Université catholique de Louvain.
- Chen, Y., Cui, Z., und Hao, L. (2019). "Virtual reality in lighting research: Comparing physical and virtual lighting environments". In: *Lighting Research & Technology* 51.6, pp. 820– 837. doi: 10.1177/1477153518825387.
- CIE – International Commission on Illumination (2006). "Test Cases to Assess the Accuracy of Lighting Computer Programs". CIE Technical Report 171
- Higuera-Trujillo, J. L., López-Tarruella Maldonado, J. und Llinares Millán, Ca. (2017). "Psychological and physiological human responses to simulated and real environments: A comparison between Photographs, 360° Panoramas, and Virtual Reality". In: *Applied Ergonomics* 65, pp. 398–409. doi: 10.1016/j.apergo.2017.05.006.
- Kajiya, J. T. (1986). "The rendering equation". In: *SIGGRAPH Comput. Graph.* 20, pp. 143–150. doi: 10.1145/15886.15902
- Karis, B. (2013). "Real Shading in Unreal Engine 4". In: *SIGGRAPH 2013 Presentation Notes*.
- Ruppertsberg, A. I. und Bloj, M. (2008), "Creating physically accurate visual stimuli for free: Spectral rendering with RADIANCE". In: *Behavior Research Methods* 40(1), pp. 304–308. doi: 10.3758/BRM.40.1.304
- Walter, B., Marschner, S. und Li, H. und Torrance, K. E (2007). "Microfacet Models for Refraction through Rough Surfaces". In: *Eurographics Symposium on Rendering*. 195-206. 10.2312/EGWR/EGSR07/195-206.

Ward Larson, G. und Shakespeare, R. (1998). "Rendering with radiance: the art and science of lighting visualization". Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.