



Bachelorarbeit
im Studiengang Medientechnik

Eingereicht von: Enis Inep

Matrikelnummer: 7022248

Mail: enis.inep@stud.hs-emden-leer.de

Erstprüfer:

Prof. Dr. rer. nat. Thies Pfeiffer

Zweitprüfer:

Jannik Franssen, M.Eng.

Hamburg, 09 2024

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Relevanz des Themas	1
1.2	Problemstellung: Tiefenwahrnehmung bei Panoramabildern auf 2D Monitoren	1
1.3	Zielsetzung und Forschungsfragen.....	1
1.4	Aufbau der Arbeit.....	1
2	Theoretische Grundlagen	4
2.1	Wahrnehmungspsychologische Grundlagen (Hergovich, 2022a).....	4
2.1.1	Parallax Effect und Tiefenhinweise (monokular vs. binokular)(Wölfel, 2023).4	4
2.1.2	Grenzen der Tiefenwahrnehmung auf 2D-Displays.....	5
2.2	360°-Panoramen als Darstellungsformat.....	5
2.2.1	Equirectangular Projection und Geometrische Verzerrungen.....	6
2.3	Texturbasierte Verfahren der Tiefendarstellung	7
2.3.1	Texture Baking.....	7
2.3.2	Rolle von Depth Maps.....	8
2.3.3	(Überblick Modifikatoren in Blender)	10
2.4	Überblick bestehender Ansätze	10
2.5	Stand der Forschung im Bereich Tiefenwahrnehmung Fehler! Textmarke nicht definiert.	
3	Methodik	13
3.1	Forschungsdesign: Analyse und Bewertung der Szenarien	13
3.2	Definition der Szenarien.....	13
3.2.1	Szenario 1: Baseline-3D-Rendering mit gezielten Tiefenherausforderungen..	13
3.2.2	Szenario 2: Depth Map + 360°-Panorama (<i>Babylon.js</i> , Artefakte, Optimierungsstrategien)	13
3.2.3	Szenario 3: Low-Poly-Rekonstruktion aus Panorama + ggf. Depth Map (Tools wie zForm, HoHoNet).....	14

3.3	Technische Umsetzung	14
3.4	Evaluationsdesign.....	14
3.4.1	Stimulusmaterial.....	14
3.4.2	Auswertungsmethode	14
4	Ergebnisse	15
4.1	Technische Ergebnisse der Szenario-Umsetzungen.....	15
4.2	Beobachtete Optimierungseffekte (z. B. weniger Artefakte, visuelle Glätte).....	15
4.3	Ergebnisse der Nutzerbefragung (Darstellung in Tabellen, Diagrammen).....	15
4.4	Zusammenfassung der Ergebnisse in Bezug auf Forschungsfrage	15
5	Diskussion	16
5.1.1	Interpretation der Ergebnisse im Kontext der Forschungsfrage.....	16
5.1.2	Vergleich mit Literatur(?)	16
5.1.3	Bewertung der Verfahren: Stärken, Grenzen, Anwendungspotenzial	16
6	Fazit und Ausblick	17
6.1	Beantwortung der Forschungsfrage	17
6.2	Beitrag der Arbeit.....	17
6.3	Offene Fragen und Ausblick	17
7	Anhang	XVIII

1 Einleitung

Im Rahmen meines Praktikums bei einem VR/XR-Start-up in Hamburg habe ich ein ausgeprägtes Interesse an der Tätigkeit als 3D Artist entwickelt. Nach Abschluss des Praktikums habe ich mich am Mixed-Reality-Lab der Hochschule Emden/Leer nach einem geeigneten Thema für eine Bachelorarbeit erkundigt. Durch meinen Betreuer, Herrn Dr. Prof. Theis Pfeiffer, wurde ich darüber informiert, dass das Mixed-Reality-Lab regelmäßig mit 360°-Panoramen im Bereich der virtuellen Realität arbeitet. Diese Bilder enthalten keine Tiefendaten und wirken daher zweidimensional. Mithilfe einer Depth Map ist es möglich, ein Tiefeneindruck zu erzeugen, der dem Panorama mehr Tiefe verleiht. Allerdings weist diese Methode auch einige Schwächen auf. Das Verfahren könnte optimiert werden, um diesen Tiefeneindruck zu erzeugen. Im Rahmen der Analyse werden drei Herangehensweisen erörtert und umgesetzt. Darüber hinaus interessieren mich auch Bilder und Computergrafik im Allgemeinen und daher ist das Thema für mich von großem Interesse.

1.1 Motivation und Relevanz des Themas

Wir sind motiviert, weil wir motiviert sind

1.2 Problemstellung: Tiefenwahrnehmung bei Panoramabildern auf 2D Monitoren

1.3 Zielsetzung und Forschungsfragen

Das zentrale Ziel dieser Arbeit ist der Versuch, mithilfe von drei unterschiedlichen Darstellungsverfahren ein visuell überzeugendes Ergebnis für die Wahrnehmung von Tiefe auf einem 2D-Monitor zu erzielen. Die zugrunde liegende Forschungsfrage lautet daher: Wie lassen sich drei Verfahren zur Darstellung von Panoramabildern auf 2D-Monitoren technisch korrekt und visuell optimiert umsetzen, um eine möglichst überzeugende Tiefenwahrnehmung zu erzielen, und welche Grenzen zeigen sie dabei. Somit könnte ein Einstieg in die Weiterführende Optimierung vom Tiefeneindruck, für Panoramabilder geschafft werden.

1.4 Aufbau der Arbeit

Die drei Herangehensweisen werden in einer 3D-Umgebung erstellt und jedes Szenario wird individuell optimiert.

Szenario 1: S1

Das Basisszenario S1 ist ein 3D-Rendering, in dem Objekte bewusst so angeordnet sind, dass Grenzen der Tiefenwahrnehmung ausgereizt werden. Die Herausforderung besteht darin, dass Nutzer:innen bei der Betrachtung Schwierigkeiten haben können, Tiefe, Abstände und Relationen korrekt einzuschätzen. Damit dient S1 als Testszene, um systematisch zu prüfen, welche Probleme bei der Wahrnehmung von Tiefe auftreten.

Aufbauend auf S1 werden weitere Szenarien erzeugt. Diese stellen verschiedene technische Lösungsansätze dar, um die Wahrnehmung von Tiefe zu verbessern. Ein Highpoly-Rendering wie in S1 zeigt zwar hohe Detailtreue, bringt aber Herausforderungen hinsichtlich Performance und Dateigröße mit sich, insbesondere in einer Web-Umgebung.

Szenario 2: S2

Szenario 2 S2 erweitert die Basisszene S1 durch die Fusion einer Depth Map mit dem 360°-Panorama. Mithilfe von Babylon.js wird auf einer unterteilten Sphere ein Tiefeneindruck erzeugt. Jedes Face erhält den Tiefenwert seines Mittelpunktes, wodurch die Szene räumlich dargestellt wird.

Bei der Analyse von S2 treten jedoch neue Herausforderungen auf. Insbesondere an Spitzen, Ecken und Außenkanten entstehen Verzerrungen, wenn sich die Nutzer:innen im Raum bewegen oder drehen. Dies liegt daran, dass die Kanten der Faces nach hinten abknicken und in der Mitte nadelartige Spitzen entstehen, die den Gesamteindruck der Szene beeinträchtigen.

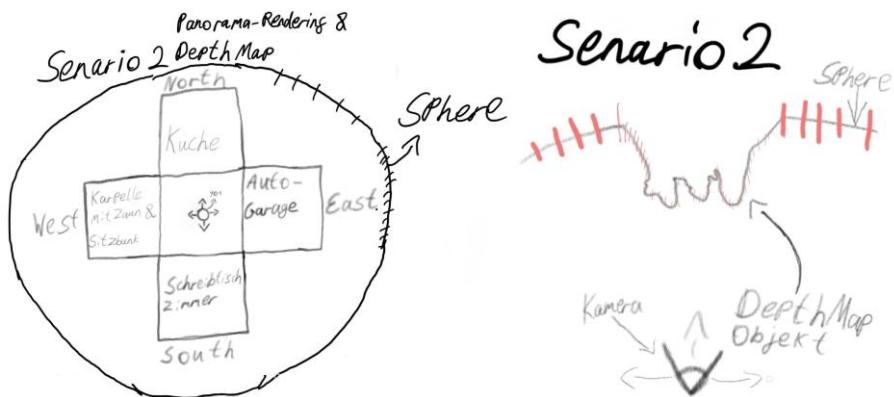


Abbildung 1 Grobe Skizze Szenario 2

Szenario 3 S3 verfolgt den Ansatz, ein Lowpoly-Mesh aus der Basisszene S1 zu erzeugen, das mit gebackenen Texturen versehen werden kann. Ziel ist es, ein visuell zufriedenstellendes Ergebnis zu erreichen, ohne ein neues Mesh manuell zu modellieren. Im Unterschied zu S2 soll das Mesh hier nicht zwingend als durchgehendes, geschlossenes Modell umgesetzt werden. Vielmehr bietet sich die Möglichkeit, Objekte im Raum vom Hintergrund zu trennen, sodass unnötige Flächen insbesondere hinter dem eigentlichen Mesh eingespart werden können.

Da die konkrete Herangehensweise für S3 noch nicht endgültig festgelegt ist, besteht eine Herausforderung darin, eine geeignete Methode zur Lowpoly-Erzeugung zu identifizieren. Dabei müssen einerseits visuelle Qualität und Tiefeneindruck gewahrt bleiben, andererseits dürfen Performance und Dateigröße nicht aus dem Ruder laufen. Die Trennung von Objekten und Hintergrund erfordert zudem eine saubere Segmentierung, um Darstellungfehler zu vermeiden. Für eine fundierte Bewertung wird es sinnvoll sein, die Ergebnisse online oder in Form von Videodemonstrationen zu präsentieren, damit der Parallax-Effekt während der Beurteilung berücksichtigt werden kann.

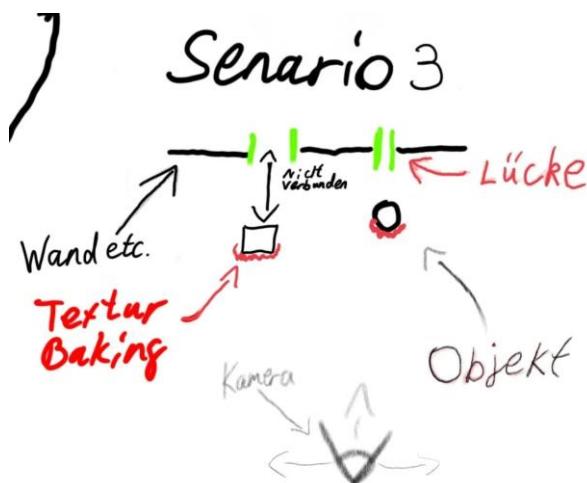


Abbildung 2 Grobe Skizze Szenario 3

2 Theoretische Grundlagen

Das Thema der Untersuchung lautet: "Untersuchung dreier Verfahren zur Darstellung von Panoramabildern mit visuell optimierter Tiefenwahrnehmung und ihren Grenzen". Für die Untersuchung ist ein gewisses Maß an Verständnis der Wahrnehmungspsychologie erforderlich. Es stellt sich die Frage, wie Tiefe auf einem 2D-Monitor wahrgenommen wird, wann genau Tiefe erkannt wird, was der Parallaxe-Effekt ist und wozu er wichtig ist. Zudem wird erörtert, wozu DepthMaps dienen und welche Funktionen sie bieten.

Im Folgenden erfolgt eine Definition des Begriffs *360°-Panorama*. Ein 360°-Panorama liegt in equirektangularer Projektion vor. Dabei wird die sphärische Umgebung in eine zweidimensionale, rechteckige Fläche mit einem Seitenverhältnis von 2:1 transformiert. Dieses Format kann in 3D-Anwendungen wieder auf eine Kugelgeometrie projiziert werden, wodurch eine immersive Rundumsicht entsteht. In der vorliegenden Arbeit wird ein direkt anschließendes, umlaufendes Bild als 360°-Panorama bezeichnet.

Kommentiert [E11]: Passt das hier? Und beleg fehlt!

2.1 Wahrnehmungspsychologische Grundlagen (Hergovich, 2022a)

Unnötiges Kapitel

Kommentiert [E12]: Kann weg? Kann Weg!

2.1.1 Parallax Effect und Tiefenhinweise (monokular vs. binokular)(Wölfel, 2023)

Wie bereits in den „Theoretischen Grundlagen“ angedeutet, setzt dieses Kapitel grundlegende Kenntnisse der Wahrnehmungspsychologie voraus. Im Folgenden wird daher die Tiefenwahrnehmung in ihren monokularen und binokularen Formen näher erläutert. Ein anschauliches Beispiel hierfür ist die Wahrnehmung von Objekten entlang einer Bahnstrecke.

„Schauen wir während der Fahrt aus dem Zug, rauschen nahe Objekte schnell an uns vorbei, ohne dass wir sie scharf wahrnehmen können, während Objekte am Horizont nahezu unverändert bleiben“ (Hergovich, 2022b, S. 76). Dieses Phänomen wird als Bewegungsparallaxe (*Parallax Effect*) bezeichnet und gilt als monokularer Tiefenhinweis, da bereits ein Auge genügt, um relative Tiefenunterschiede wahrzunehmen (Hergovich, 2022b, S. 76).

Kommentiert [E13]: Hab ich das Richtig direkt Zitiert?

Im Rahmen der binokularen Tiefenwahrnehmung kommt es aufgrund der leichten Versetzung der Augen zum Auftreten von zwei verschiedenen Bildern, die in der Fachliteratur als Querdisparation bezeichnet werden. Das Gehirn gleicht diese beiden Bilder ab und unterscheidet, inwiefern sich die Positionen der Objekte voneinander versetzt sind (Schönhammer, 2013, S. 196–197). Die daraus resultierende Tiefenwahrnehmung übertrifft die

eines einzelnen Auges (Hibbard et al., 2023, S. 7–8). Um die Thematik zu veranschaulichen, kann der Versuch unternommen werden, einen Faden durch ein Nadelöhr zu fädeln, zunächst mit einem geschlossenen Auge und anschließend mit beiden Augen.

2.1.2 Grenzen der Tiefenwahrnehmung auf 2D-Displays

Der Versuch, Tiefe auf einem 2D-Monitor darzustellen, auf den beide Augen fokussiert sind, führt nicht zur Querdisparation, da das Bild lediglich auf einer flachen Oberfläche betrachtet wird (vgl. Lehn et al., 2022, S. 422). Die Tiefenwahrnehmung erfolgt daher ausschließlich monokular, auch wenn beide Augen genutzt werden. Um dennoch einen Eindruck von Tiefe zu erzeugen, können verschiedene Gestaltungsmittel eingesetzt werden. So lassen sich durch perspektivische Fluchtpunkte oder Größenunterschiede von Objekten räumliche Effekte hervorrufen. Auch eine gezielte Nutzung von Licht und Schatten oder Hierarchie der Objekte, wenn im Vordergrund stehende Objekte, die im Hintergrund verdecken, können Tiefe im Bild schaffen. (vgl. Hergovich, 2022b, S. 77).

Darüber hinaus kann Bewegung eine Bewegungsparallaxe erzeugen und die Tiefenwahrnehmung verstärken. Auch externe Faktoren beeinflussen die Wahrnehmung: Untersuchungen zeigen, dass sowohl der Betrachtungsabstand als auch die Displaygröße eine Rolle spielen. Je näher der Betrachter am Monitor sitzt und je größer das Display ist, desto stärker wirken Fluchtpunkte und Größenunterschiede, wodurch die Tiefe intensiver wahrgenommen wird (vgl. Miyashita et al., 2022, S. 5:2–5:5). Es ist wichtig auf diese Methoden zuzugreifen, um mehr Tiefe auf 2D-Monitoren zu erzeugen, denn auf Standbildern bleibt der Tiefeneindruck eher flach (Miyashita et al., 2022, S. 5:10–5:11).

2.2 360°-Panoramen als Darstellungsformat

Monitore und Bildformate besitzen feste Grenzen und beschränken somit den sichtbaren Bildausschnitt (Hoffmann, 2025, S. 10–11). Sie unterliegen normierten und standardisierten Größenverhältnissen, die sich historisch aus technischen Entwicklungen ergeben haben (Hoffmann, 2025, S. 19–20). Um den wahrnehmbaren Darstellungsbereich über diese Festgelegten Grenzen zu erweitern, werden Panoramen eingesetzt. Diese ermöglichen eine erweiterte bildliche Wahrnehmung sowohl auf zweidimensionalen Bildschirmen oder im VR-Bereich. (Hoffmann, 2025, S. 11–12).

Dabei kann man unter verschiedenen Arten von Panoramabildern unterscheiden, am bekanntesten sind 180°-Darstellungen. Für diese Bachelorarbeit werden jedoch 360°-

Panoramen betrachtet, da diese im Kontext von 3D-Visualisierung und Tiefenwahrnehmung relevanter sind. Diese Panoramen decken das gesamte Sichtfeld ab, das sich über 360° horizontal und 180° vertikal erstreckt. (Hoffmann, 2025, S. 23–25). Praktisch nutzt man 360°-Panoramen oft in VR-Anwendungen oder im Architekturbereich zur Simulation, in den anwendungsfällen wirken sie besonders Immersiv.

Diese Darstellungsform ist besonders hilfreich, um seine Umgebung von einem Zentralen Punkt aus wahrzunehmen. In der Regel sehen wir unser Sichtfeld um uns herum, ein Panorama bildet dieses Sichtfeld auf eine Fläche ebene ab und erzeugt damit eine Verzerrung (Hoffmann, 2025, S. 21–24).

Kommentiert [EI4]: Eigentlich gute Überleitung für nächstes Thema lol

2.2.1 Equirectangular Projection und Geometrische Verzerrungen

Wir betrachten die Welt in einer sphärischen Geometrie unser gesamtes Sichtfeld bildet eine 360° Kugeloberfläche. Jeder Punkt in diesen Raum lässt sich durch Richtungsvektoren beschreiben oder Sphärischen Koordinaten beschreiben. Sobald man diese Umgebung auf eine 2D Fläche überträgt, in Form eines Bilds oder ähnliches wird im Oberen und unterem Bereich das Bild verzerrt. Das 360°-Panorama wird an der Oberen und Unteren Bildfläche gestaucht und in die Ecken des Equirectangulars „abgewickelt“, dass Breitengrade linear auf die y-Achse und Längengrade linear auf die x-Achse gemappt werden. Mathematisch Beschrieben:

$$x = \lambda, y = \phi$$

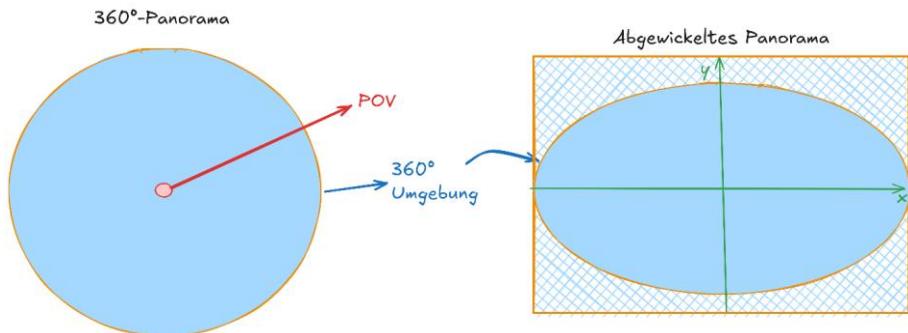
Längengrad $\lambda \in [-180, 180]$ und Breitengrad $\phi \in [-90, 90]$.

Der Maßstab M wächst mit der breite:

$$M = \frac{1}{\cos(\phi)}$$

Am Äquator ($\phi = 0^\circ$) ist $M = 1 \rightarrow$ keine Verzerrung

An den Polen ($\phi = \pm 90^\circ$) geht $M \rightarrow \infty \rightarrow$ maximale Verzerrung



2.3 Texturbasierte Verfahren der Tiefendarstellung

2.3.1 Texture Baking

Texture Baking ist ein Verfahren, bei dem Rendering-Ergebnisse eines High-Poly Modells auf eine zweidimensionale Bitmap Textur übertragen werden, um diese anschließend auf ein Low-Poly Modell anzuwenden (Zhang et al., 2023, S. 647–648). Texture Baking wird oft für Spiele oder VR/AR Anwendungsfälle verwendet, da es Speicher und Rechenressourcen sparen kann. Andernfalls müssen beispielsweise für ein 3D-Modell einer Schreibtischlampe mehrere Texturen in hoher Auflösung gerendert werden. (Adobe, 2025)

Der Schirm der Lampe ist grünlich transparent und matt, der Lampenfuß besteht aus braunem Holz, und die Beine sind aus glänzendem Metall. Das wären 3 aufwändige Texturen für ein 3D

Modell, die viel Speicherplatz benötigen. Mit Texture Baking könnte man die 3 verschiedenen Texturen zu einer kombinieren (Knodt et al., 2023). So werden beispielsweise Farben (Base Color), Roughness map, Normal Map und Metallizität (Metallic) in einem „Atlas“ gespeichert (Blender Foundation, 2025a). Das reduziert den Speicherbedarf und auch den Rechenaufwand bei der Laufzeit. Um Texture Baking in Blender anzuwenden, braucht das 3D-Modell eine UV-map und eine „Image Texture Node“ um die Ergebnisse zu speichern (Blender Foundation, 2025a). Der Prozess ist ähnlich ab wie ein Rendering verfahren, nur mit dem Unterschied, dass das Ergebnis nicht mit einem Framebuffer sondern auf eine Texture gespeichert wird.

In Blender kann man zwischen zwei grundlegenden Bake-Verfahren unterscheiden. Im ersten Verfahren wird von einem Quellobjekt, gängig wird hier ein High Poly verwendet, auf ein Lowpoly oder das gewünschte Objekt gebacken. Dieses Vorgehen ist die gängigste Methode, da sich damit beliebige Texture Maps wie Normal, Ambient Occlusion oder Roughness vom detaillierten Modell auf ein vereinfachtes übertragen lassen. (Blender Foundation, 2025b)

Im zweiten Verfahren wird direkt auf dasselbe Mesh gebacken. Dieses Verfahren wird verwendet, um Material- oder Beleuchtungsinformationen innerhalb eines einzelnen Modells zu konsolidieren, etwa wenn mehrere Shader oder Materialien zu einer einheitlichen Textur zusammengeführt werden sollen. Dadurch kann man komplexe Node-Strukturen reduzieren oder statische Lichtinformationen, beispielsweise Lightmaps, direkt in das Objekt integrieren. (Blender Foundation, 2025b)

Das Verfahren erfordert eine saubere Topologie und korrekte UV-Koordinaten, da jede geometrische Inkonsistenz (z. B. überlappende Faces, inverted Normals oder non-manifold Edges) zu Fehlberechnungen führt. Wichtig ist hierbei die Formgebenden Edges im Low-Poly beibehalten werden, um so genannte Fehler zu auszuschließen.

2.3.2 Rolle von Depth Maps

Eine Depth Map ist ein Bild, das Tiefeninformationen einer Szene speichert (Jähne, 2024, S. 271–272). Sie gehört wie eine Normal Map oder eine Bump Map zur Kategorie der Textur Maps. Gespeichert wird dabei der Abstand zwischen der Kamera und dem jeweils nächstgelegenen 3D Punkt der Szene (Jähne, 2024, S. 283–289). Die Darstellung erfolgt in der Regel über einen Farbverlauf, meist zwischen Schwarz und Weiß. Helle Bereiche stehen für nahe Objekte, während dunkle Bereiche größere Entfernung kennzeichnen. Grundsätzlich sind auch andere Farbschemata möglich, die Monokular-Depth Graustufenabbildung gilt jedoch als Standard (Divyanth et al., 2023, S. 2–3). Wir wissen, dass Farben in Bildern, durch

Kommentiert [E15]: Quellen Aktualisieren und Prüfen.

die Kombination von drei Farbkanälen (RGB) entsteht. Jedes dieser Kanäle enthält einen Helligkeitswert der Farbe, in einem Bereich von 0 bis 255, das Mischen der 3 Farbkanäle ermöglicht das additive Farbmischen. Eine Depth Map besteht nur aus einem 8-Bit Graustufenbild, mit einem Graukanal ebenfalls von 0 bis 255. (*What Is Depth Anything V2, 2025*)

Kommentiert [E16]: Keine Wissenschaftliche Quelle

Depth Maps können auf unterschiedliche Weise erzeugt werden. Zu den gängigen Methoden zählen Stereovision, Time-of-Flight, Structured Light oder monokulare Schätzungen mithilfe von KI (Jähne, 2024, S. 283–289). Im hierfür wurde Blender verwendet, wobei über die Compositing-Nodes ein sogenannter Z-Depth-Pass aus dem Render extrahiert wurde. Blender kann die Tiefeninformationen direkt aus der 3D-Szene berechnen, indem für jedes Pixel der Abstand zur Kamera entlang der Z-Achse bestimmt wird. Diese Werte werden im sogenannten Z-Buffer gespeichert und ergeben in ihrer Gesamtheit die Depth Map (Blender Foundation, 2025a).

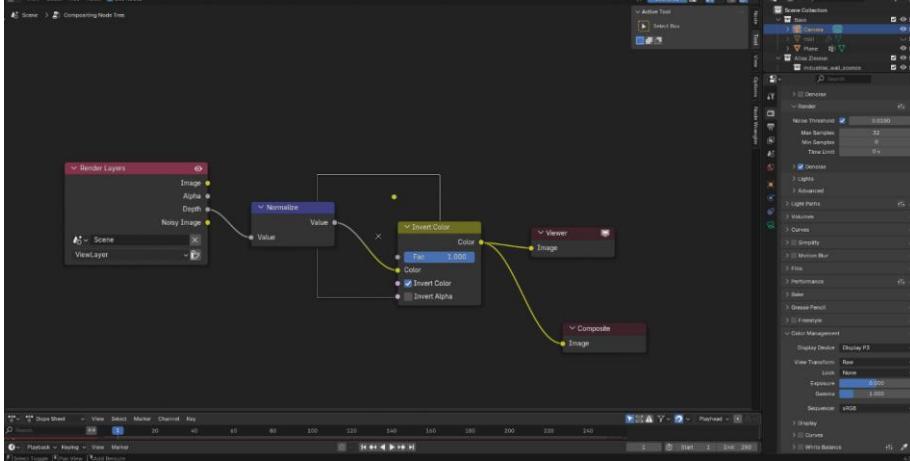
Gängige Einsatzfelder von Depth Maps sind beispielsweise die Hinderniserkennung in virtuellen Umgebungen oder die Modellierung von Szenen. Darüber hinaus werden sie im sogenannten Depth-Image-Based Rendering (DIBR) genutzt, um neue Kameraperspektiven aus vorhandenen Tiefeninformationen zu berechnen (Schmeing & Jiang, o. J., S. 279–280) (Lipski et al., 2014, S. 943–944). Auch in modernen Anwendungen, etwa im autonomen Fahren oder in der Robotik, dienen Depth Maps als Grundlage für die Erstellung topologischer Karten zur Erfassung von Hindernissen und Entfernung (Saha et al., 2022, S. 1).

Kommentiert [E17]: Jahr

Schwierigkeiten zur Tiefeneinschätzung hat eine Depth Map mit spiegelnden Oberflächen, zudem fassen sie nur die Vorderfläche von einer Szene auf und liefern dadurch eine eingeschränkte Sicht. Außerdem sind sie häufig verrauscht und enthalten oft Schätzungsfehler, wenn man sie mithilfe einer KI erstellt (Divyanth et al., 2023, S. 6–7).

Mit all diesen Möglichkeiten kann man weiterführend viel mit Depth Maps machen, kombiniert Depth Maps aus verschiedenen Blickwinkeln können zu einer dichten 3D-Rekonstruktion verhelfen (Izadi et al., 2011, S. 1). Neuerdings kann man mit Open-Source Frameworks, wie zum Beispiel NeRF (Neural Radiance Fields), realistische 3D-Szenen mit ihnen erstellen (Ito et al., 2024, S. 5). (In dieser wird das Szenario 3 nach dem Prinzip erstellt und ist daher für die Theorie wichtig.)

- Relative und Absolute Depth Maps?



2.3.3 (Überblick Modifikatoren in Blender)

2.4 Überblick bestehender Ansätze

2.4.1 Allgemeine Verfahren der Tiefendarstellung in 2D-Panoramen

Im Rahmen der Literaturrecherche zeigte sich, dass bereits verschiedene Verfahren existieren, um Tiefe in zweidimensionalen Panoramen darzustellen. Eine Studie von 2007 zeigt wie man Layer verwenden kann, um Tiefe in Panorama Bildern zu erstellen, dafür wurden mehrere Bilder von einer Szene aufgenommen, um gezielt den Parallaxeneffekt zu erzeugen. Das Konzept heißt Layered Depth Panorama (LDP) und ist eine Erweiterung des klassischen 2D-Panoramas. Während herkömmliche Panoramen nur ein einziges Tiefenniveau pro Pixel abbilden, speichert das LDP mehrere Tiefenschichten (Layer) pro Blickrichtung ähnlich wie ein „Layered Depth Image“ (LDI), aber in zylindrischer Projektion. (Zheng et al., 2007)

Hier wurde aber nur ein 180-220° Panorama verwendet und hinter den Layern befindet sich keine 3 dimensionale tiefe, der Parallaxe Effekt erzeugt dennoch ein Gefühl von tiefe, aber falls man sich den Layern nährt, bemerkt man das diese Flach sind.

2.4.2 Spezifische Verfahren im Kontext 360°-Panoramen

Eine bereits weit erforschte Variante im Indoor-Bereich ist die Rekonstruktion von 3D-Umgebungen aus einem oder mehreren 360°-Panoramen unter Verwendung von überwachtem Deep Learning (He et al., 2022, S. 5173–5174; Silveira et al., 2023, S. 1–2; Wang et al., 2023, S. 1–2). Dabei werden die Panoramabilder in ein neuronales Netz eingespeist, das zuvor mit

zahlreichen Panoramen und deren zugehörigen Tiefeninformationen trainiert wurde (He et al., 2022, S. 5174). Diese sogenannten Ground-Truth-Informationen unterscheiden sich je nach Studie und können beispielsweise Depth Maps, Normalenvektoren oder die Kamerahöhe umfassen (He et al., 2022, S. 5173–5174; Rey-Area et al., 2022, S. 1–2). Durch dieses Training lernt das neuronale Netz, die Tiefenstruktur einer Szene aus einem in manchen Fällen auch aus mehreren Panoramabildern zu rekonstruieren (Silveira et al., 2023, S. 11)

Die Ergebnisse werden typischerweise in Form einer Depth Map, einer Point Cloud oder eines 3D-Meshes ausgegeben, aus denen sich die räumliche Struktur ableiten lässt (Silveira et al., 2023, S. 6–8). Solche Verfahren finden bislang vor allem Anwendung in Bereichen wie Innenarchitektur, Schulungsumgebungen oder Visualisierungsanwendungen, da sie zwar sehr detaillierte Geometrien erzeugen, diese jedoch häufig große Datenmengen verursachen oder sich primär auf Innenräume beschränken (Zeng et al., 2020, S. 1–2; Silveira et al., 2023, S. 9–10). Für Anwendungen, die auf eine kompakte und effiziente Tiefendarstellung abzielen etwa zur visuellen Unterstützung in interaktiven Interfaces oder mobilen Systemen sind diese Verfahren daher nur bedingt geeignet. Gezielt wird eine Ressource sparsame Variante gesucht, die Webend gut aussieht (Bro ich bin tod was heißt gut aussieht?) und Performen kann. Die meisten Verfahren sind für innenraum szenarien gedacht oder erzeugen nur ein

Kommentiert [EI8]: Kann weg oder in die nächste spalte?

Calibrating Panoramic Depth Estimation for Practical Applications (2024)

- „Calibrating Panoramic Depth Estimation for Practical Applications“ (2024) [arXiv](#)
- Relevanz:** Untersucht die Genauigkeit von Tiefen-Schätzungen bei Panorama-Darstellungen und zeigt auf, wo Limitationen sind — gut für deinen Abschnitt „Einschränkungen und Forschungsbedarf“.
- Nutzen:** Du kannst daraus zitieren, dass aktuelle Verfahren zwar Fortschritte machen, aber noch unsicher sind – z. B. bei großen Bereichsverschiebungen oder reduzierter Verlässlichkeit der absoluten Tiefenwerte.

2.4.3 Einschränkungen und Forschungsbedarf

Im vorherigen Kapitel wurden Verfahren vorgestellt, die sich mit der Tiefenschätzung aus 360°-Panoramen befassen. Dabei zeigte sich, dass nahezu alle aktuellen Arbeiten ihren Fokus auf die Erzeugung von Depth Maps legen. Hier soll nun erläutert werden, wieso das so ist und warum die Forschung den Schritt umgeht, mithilfe der erstellten Depth Map ein 3D-Modell zu erstellen. Diese Größe lässt sich mithilfe von einem Vergleichswert messen, zum Beispiel misst Root Mean Squared Error (RMSE), wie weit die vorhergesagte Tiefe vom echten (Ground-Truth) Wert entfernt ist. Zudem kann auch der Durchschnittlicher absoluter Fehler Mean Absolute Error (MAE) gemessen werden, dieser misst den Durchschnitt aller Fehler, egal ob zu groß oder zu klein. Zuletzt kann man auch messen, wie viele Pixel ungefähr richtig sind. Das führt dazu, dass Tiefe eine Eindeutige, metrisch Bewertbare Zielgröße ist. Man könnte also sagen, dass Depth Maps standardisierte vergleichbare Größen sind, die sich Optimieren lassen. (Silveira et al., 2023, S. 7–8).

Da es sich bei Depth Maps um zweidimensionale Bilder handelt, ist für die vollständige Erstellung eines 3D-Modells eine weitere Bearbeitung erforderlich. Silveira et al. betonen hierzu ausdrücklich: „Per-face depth estimation tends to produce discontinuities along face edges [87, 119, 32], and post-processing is required to obtain the full 3D model of the scene.“

(Silveira et al., 2023, S. 20)

Diese Nacharbeitungen kann je nach Verfahren, zu einem hochauflösenden High-Poly Mesh oder Point Cloud, die für Performance Zwecke im Web problematisch werden könnten. Ich möchte hier darauf hinweisen, dass es prinzipiell nicht unmöglich ist, aus einer Depth Map eine visuell ansehnliches Low Poly zu erzeugen. Diese Nachbearbeitung könnte aufwendig werden, in diesem Fall arbeiten wir mit einem 360°-Panorama. Daher schätze ich das speziell der aufwand Groß ist ein Optimierte LP zu erstellen, da du hier viel mehr Bildinformationen hast als sonst mit einem normalen Bild.

Kommentiert [EI9]: Quelle nach 2022

Der Fokus dieser Arbeit liegt daher nicht auf der Erzeugung immer präziserer Depth Maps, der Fokus dieser Arbeit liegt daher nicht auf der Erzeugung immer präziserer Depth Maps, sondern auf der Entwicklung eines ressourcenschonenden, visuell überzeugenden 3D-Modells.

Kommentiert [EI10]: Ich bin mir nicht Sicher ob ich hier gerade eine These geschrieben habe, die ich nicht belegen kann.

2.5 Limitationen

Ja es gibt viel. 😊

3 Methodik

3.1 Forschungsdesign: Analyse und Bewertung der Szenarien

- Wir machen ein Vergleich
- Bewertet wird sowas wie

3.2 Definition der Szenarien

Hier erkläre ich das ich die 3 Szenarien nun beschreiben werde und wie der Aufbau umgesetzt wurde. Skizzen für jedes Szenario erstellt, um ein Verständnis darüber zu bekommen, was ich mir gedacht habe.

3.2.1 Szenario 1: Baseline-3D-Rendering mit gezielten Tiefenherausforderungen

Zur Erstellung des Baseline-Szenarios wurde in Blender ein 3D-Modell entwickelt. Dabei kam das Add-on *Poly Haven Asset Browser* (*Poly Haven Asset Browser*, 2025) zum Einsatz, das eine integrierte Bibliothek hochauflösender, texturierter 3D-Modelle bereitstellt, die per Drag-and-drop in die Szene eingefügt werden können. Einige Assets wurden von der Plattform Fab heruntergeladen und in die Szene eingefügt. (*Fab*, 2025) Andere wurden selbständig modelliert und ebenfalls eingebaut. Nach der Platzierung erfolgten Optimierungen, und der Szene wurden Lichter hinzugefügt. Anschließend wurden alle Modelle zusammengeführt und als .glb-Datei exportiert, das Format eignet sich besonders für 3d Modelle, die man im Web-Visualisieren möchte (*glTF - Runtime 3D Asset Delivery*, 2020). Die Datei wurde anschließend auf Babylon.js eingebettet und dort dargestellt. Hierbei wurde darauf wertgelegt das die Szene den Bewertungsansprüchen entspricht.

Kommentiert [EI11]: Absichtlich noch schwammig, da ich noch nicht konkret weiß wie bewertet wird.

3.2.2 Szenario 2: Depth Map + 360°-Panorama (Babylon.js, Artefakte, Optimierungsstrategien)

- Anschließend wurde ein 360° Panorama gerendert und eine Depthmap mit Blender erstellt(wie genau das gemacht wurde kann ein eigenes Unterkapitel werden.)
-

3.2.3 Szenario 3: Low-Poly-Rekonstruktion aus Panorama + ggf. Depth Map (Tools wie zForm, HoHoNet)

3.3 Technische Umsetzung

3.4 Evaluationsdesign

3.4.1 Stimulusmaterial

3.4.2 Auswertungsmethode

4 Ergebnisse

- 4.1 Technische Ergebnisse der Szenario-Umsetzungen**
- 4.2 Beobachtete Optimierungseffekte (z. B. weniger Artefakte, visuelle Glätte)**
- 4.3 Ergebnisse der Nutzerbefragung (Darstellung in Tabellen, Diagrammen)**
- 4.4 Zusammenfassung der Ergebnisse in Bezug auf Forschungsfrage**

5 Diskussion

5.1.1 Interpretation der Ergebnisse im Kontext der Forschungsfrage

5.1.2 Vergleich mit Literatur(?)

5.1.3 Bewertung der Verfahren: Stärken, Grenzen, Anwendungspotenzial

6 Fazit und Ausblick

6.1 Beantwortung der Forschungsfrage

6.2 Beitrag der Arbeit

6.3 Offene Fragen und Ausblick

7 Anhang