

Algorithmen zur Visualisierung der Relativitätstheorie

Studienarbeit im Fach Informatik

von

Oliver Sudmann (6096486)

Paderborn, 6. Februar 2004

Inhaltsverzeichnis

Einleitung und Motivation	5
1 Die spezielle Relativitätstheorie	7
1.1 Relativistische Addition der Geschwindigkeiten	7
1.2 Die Lorentz Transformation	9
1.2.1 Bemerkung zur Lorentz Transformation	10
1.3 Die Längenkontraktion	10
1.3.1 Zahlenbeispiele zur Längenkontraktion	11
1.4 Längenkontraktion bei Myonen	12
1.5 Die Zeitdilatation	12
1.6 Die relativistische Masse	13
1.7 Minkowski Raum-Zeit-Diagramme	13
1.7.1 Relativierung der Gleichzeitigkeit	14
1.7.2 Beschleunigte Bewegung im Minkowski-Diagramm	16
1.8 Allgemeine Relativitätstheorie	16
2 Visuelle Effekte	17
2.1 Effekte der Lorentz Transformation	17
2.2 Auswirkungen der endlichen Lichtlaufzeit	17
2.3 Kombination aus Längenkontraktion und Lichtlaufzeit	19
2.4 Relativistischer Doppler-Effekt	19
2.5 Relativistische Aberration	21
2.5.1 Aberration und Geschwindigkeitsänderungen	21
2.6 Sonstige Konsequenzen	22
2.7 Effekte der Allgemeinen Relativitätstheorie	23

3	Visualisierungsalgorithmen	25
3.1	Ray-Tracing Algorithmen	25
3.1.1	Grundlegendes zum Ray-Tracing	25
3.1.2	Ray-Tracing nach Hsiung und Dunn	26
3.2	Bild-basierte Verfahren	30
3.2.1	Algorithmus von Weiskopf, Korbas und Ruder	31
3.3	Polygon-Rendering Algorithmen	35
3.3.1	T-Buffer Algorithmus von Hsiung, Thibadeau und Wu	35
3.3.2	Erweiterung für eine gekrümmte Beobachter-Weltlinie von Rau, Weiskopf und Ruder	40
3.4	Sonstige Visualisierungsalgorithmen	41
3.4.1	Textur-basierter Algorithmus von Daniel Weiskopf	41
3.4.2	Radiosity	41
4	Ausblicke	43
4.1	Gekrümmte Beobachter- und Objekt-Weltlinien	43
4.2	Simulation nicht visueller Eindrücke	44
4.2.1	„Gefühlte“ Zeit/Relativität der Gleichzeitigkeit	44
4.2.2	Konzepte zur „interaktive“ Steuerung	45
4.2.3	Relativistische Masse/Relativistische Massenträgheit	46
	Literaturverzeichnis	48
	Versicherung	49
	Danksagung	49

Einleitung und Motivation

Die meisten Menschen kennen den Begriff „*Relativitätstheorie*“ und verbinden ihn korrekt mit dem Namen ALBERT EINSTEIN. Dennoch wissen nur wenige, was sich genau dahinter verbirgt. In der Tat scheinen ihre Vorhersagen der natürlichen Intuition auf den ersten Blick zu widersprechen: Die Länge eines Stabes soll sich bei Bewegung reduzieren, Uhren langsamer laufen, Masse zunehmen, Lichtstrahlen sich biegen und Farben verändern. Dennoch sind diese Effekte durch Präzisionsmessungen vollkommen bestätigt; sie werden heutzutage in praktischen Anwendungen wie Atomuhren und Satellitennavigationssystemen (GPS) berücksichtigt und müssen dies auch, um die angestrebten Zeit- und Positionsgenauigkeiten zu erreichen.

Die Informatik kann nun helfen, diese fast 100 Jahre alte doch noch immer neuartige Physik einem breiten Publikum nahe zu bringen. Dies gelingt durch Simulation und Visualisierung der Eindrücke, welche sich für eine Raumfahrer bei hohen Geschwindigkeiten ergeben. Die vorliegende Arbeit bietet einen Überblick über die sich dabei ergebenden Probleme und ihre in der Literatur behandelten Lösungen, hauptsächlich Algorithmen zur Computer-Visualisierung der speziellen Relativitätstheorie.

Diese sind im Hinblick auf realistische Science Fiction Filmproduktionen (vgl. DISNEY's „*Das schwarze Loch*“) interessant. Für rein virtuelle Szenen bieten sich hier Modifikationen der bekannten Ray-Tracing Methoden an (Kapitel 3.1.2), während Bild-basierte Verfahren (Kapitel 3.2.1) gewissermaßen eine ‚relativistische Nach-Korrektur‘ von klassischen Aufnahmen realer Szenen erlauben. Computerspiele hingegen verlangen Interaktivität und hohe Bildraten, die eine in Kapitel 3.3.1 beschriebene Erweiterung des bekannten Z-Buffer Algorithmus ermöglicht. Diese transformiert die raum-zeitliche, d.h. 4-dimensionale Szenenbeschreibung in jedem Schritt (nichtlinear, doch in Linearzeit) in eine 3-dimensionale, welche dann klassisch von handelsüblicher Grafik-Hardware gerendert werden kann.

Die vorliegende Studienarbeit richtet sich insbesondere an — nicht immer physikerfahrene — Informatiker. Somit werden in Kapitel 1 zunächst die nötigen physikalischen Grundlagen dargestellt. In Kapitel 2 werden dann die visuellen Effekte der speziellen Relativitätstheorie erläutert, die wir mit Hilfe von Computer simulieren wollen. Kapitel 3 stellt hierfür verschiedene Techniken und Algorithmen aus der Literatur vor mit Beispielbildern, Vor- und Nachteilen. Der Ausblick in Kapitel 4 zeigt Ansätze für weitere Forschung: Aufheben von Beschränkungen der bekannten Algorithmen sowie Simulation nicht-visueller Eindrücke wie Zeitdilatation, Massenzunahme und (Raumschiff-)Steuerung.

Kapitel 1

Die spezielle Relativitätstheorie

Da sich diese Studienarbeit nicht alleine im Gebiet der Informatik bewegt, sondern auch im Bereich der Physik, sollen in diesem Kapitel zunächst die physikalischen Grundlagen erläutert werden.

Zu Beginn des 20. Jh. suchte man, in Analogie zum Schall, nach einem Medium, in dem sich das Licht ausbreitet. Dies wurde auch mit dem Michelson-Experiment versucht. Statt jedoch einen Nachweis für das gesuchte Übertragungsmedium für Licht zu finden, ergaben die Messungen, dass jeder Beobachter die gleiche Lichtgeschwindigkeit misst, egal ob der Beobachter sich bewegt oder ruht. Dadurch entstanden Probleme, die die bis dahin bekannte Physik in Frage stellten. Die Probleme löste Einstein mit der Relativitätstheorie.

Benutzt man die Relativitätstheorie, so bezeichnet man einen solchen Ansatz meist als relativistischen Ansatz. Die Physik vor der Relativitätstheorie nennt man auch klassische Physik. Streng genommen ist also die klassische Physik falsch. Bei kleinen Geschwindigkeiten gegenüber der Lichtgeschwindigkeit kommt man jedoch meist mit der klassischen Physik aus (s.u.).

1.1 Relativistische Addition der Geschwindigkeiten

Um in die relativistische Denkweise einzuführen, soll im folgenden zunächst ein typisch klassisches Beispiel behandelt werden.

Ein Zug fährt mit der Geschwindigkeit \vec{v} an einem Bahnsteig vorbei. Auf dem Bahnsteig sei Beobachter A, im Zug sei Beobachter B. Beobachter B sieht im Wagon eine Kugel mit der Geschwindigkeit \vec{u} rollen (vgl. Abb. 1.1). Für Beobachter A hat die Kugel aufgrund der Geschwindigkeit des Zuges die Geschwindigkeit $\vec{v} + \vec{u}$.

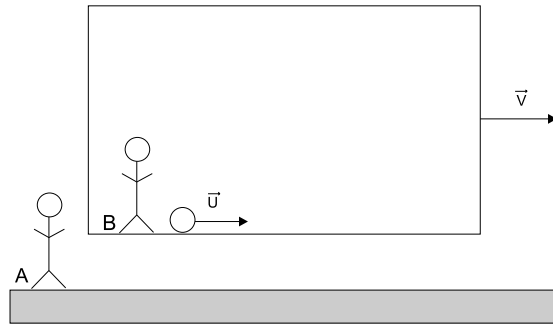


Abbildung 1.1: Kugel mit Geschwindigkeit \vec{u} rollt in einem Wagon mit Geschwindigkeit \vec{v}

Für den Übergang zu einem relativistischen Beispiel ändern wir das Szenario etwas ab: Statt der Kugel verwenden wir nun eine Taschenlampe, die Lichtquanten aussendet. Ein Lichtquant bewegt sich mit der Lichtgeschwindigkeit \vec{c} im Wagon. Der Zug fährt nun mit der Geschwindigkeit $\frac{1}{2} \cdot \vec{c}$ (vgl. Abb. 1.2).

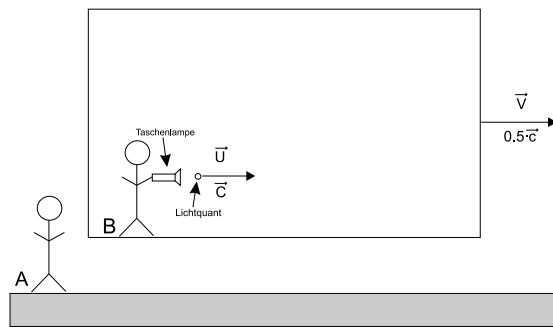


Abbildung 1.2: Taschenlampe sendet Lichtquanten mit Geschwindigkeit \vec{c} in einem Wagon mit Geschwindigkeit $\frac{1}{2} \cdot \vec{c}$

Beobachter B sieht also die Lichtquanten mit der Geschwindigkeit \vec{c} . Was aber sieht A? Nehmen wir zunächst den klassischen Ansatz:

Die Geschwindigkeit eines Lichtquanten ergibt sich dann als Summe der Geschwindigkeit des Zuges und des jeweiligen Lichtquanten:

$$\frac{1}{2} \cdot \vec{c} + \vec{c} = \frac{3}{2} \cdot \vec{c}$$

Das heißt also, dass Beobachter A Lichtquanten sieht, welche schneller als das Licht sind. Dies ist ein Widerspruch zur oben erwähnten Feststellung, dass alle Beobachter die gleiche Lichtgeschwindigkeit messen. An dieser Stelle wird deutlich, dass man die Geschwindigkeiten nicht einfach addieren kann. Für die relativistische Addition der Geschwindigkeiten gilt stattdessen:

$$v_{\text{gesamt}} = \frac{u + v}{1 + \frac{u \cdot v}{c^2}} \quad (1.1)$$

wobei c : Lichtgeschwindigkeit, v : Geschwindigkeit des bewegten Systems, u : Geschwindigkeit des Objekts im bewegten System.

Wendet man Gl. (1.1) auf das zweite Beispiel an, so erhält man:

$$\frac{c + \frac{1}{2} \cdot c}{1 + \frac{c \cdot \frac{1}{2} \cdot c}{c^2}} = \frac{\frac{3}{2} \cdot c}{1 + \frac{\frac{1}{2} \cdot c^2}{c^2}} = \frac{\frac{3}{2} \cdot c}{\frac{3}{2}} = c$$

Also sieht auch A ein Lichtquant mit der Geschwindigkeit \vec{c} .

Das erste Beispiel mit der Kugel müsste natürlich genau genommen auch relativistisch berechnet werden. Da aber die Geschwindigkeiten \vec{v} und \vec{u} klein gegenüber der Lichtgeschwindigkeit \vec{c} sind, erhält man für den Quotienten $\frac{v \cdot u}{c^2} \approx 0$. Somit ist die klassische Geschwindigkeitsaddition eine Näherung, die bei kleinen Geschwindigkeiten recht gut gilt.

1.2 Die Lorentz Transformation

Die zuvor beschriebene relativistische Geschwindigkeitsaddition lässt sich aus der Lorentz Transformation herleiten. Dabei handelt es sich um eine Koordinatentransformation.

Bewegen sich zwei Objekte mit einer relativen Geschwindigkeit \vec{v} zueinander, so befinden sie sich in verschiedenen Koordinatensystemen K und K' . Zur Umrechnung der Koordinaten verwendet man Transformationen. Zur Vereinfachung sollen sich die beiden Objekte nur entlang der z -Achse mit der relativen Geschwindigkeit \vec{v} bewegen (Abb. 1.3). Da es sich um relative Geschwindigkeiten handelt, kann man sich eines der Koordinatensysteme als ruhend (sei dies K) vorstellen und das andere entlang der z -Achse bewegt (K').

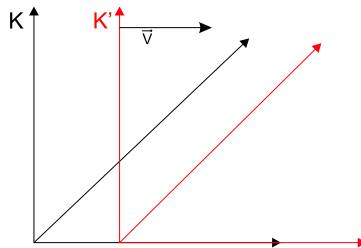


Abbildung 1.3: K -System ruht, K' bewegt sich mit Geschwindigkeit \vec{v} entlang z

Im Alltag verwendet man meist die nicht relativistische Galileo-Transformation:

$$x' = x, \quad y' = y, \quad z' = z - vt, \quad t' = t \quad (1.2)$$

Die Galileo-Transformation lässt sich beispielsweise auf einen mit der Geschwindigkeit \vec{v} fahrenden Zug anwenden. Ein Beobachter A am Bahnsteig befindet sich im System K , der Beobachter B im Zug im System K' . Für den Beobachter B ruht der Zug. Er muss also annehmen, dass sich der Bahnsteig mit der Geschwindigkeit $-\vec{v}$ an ihm vorbeibewegt. Die Galileo-Transformation drückt nun gerade diesen Sachverhalt aus. Ein fester Punkt z im Ruhesystem K ändert nach Gl. (1.2) im System K' seine Position z' in Abhängigkeit von der Zeit t und der Geschwindigkeit \vec{v} des Zuges. Ein ruhender Punkt im K -System ist also im K' -System nicht in Ruhe.

Die Galileo-Transformation ist eine klassische Transformation und gilt nur für kleine Geschwindigkeiten. Sie lässt sich aus der allgemeineren, relativistischen Lorentz-Transformation herleiten. Für

die Lorentz-Transformation gilt:

$$x' = x, y' = y, z' = \frac{z - vt}{\sqrt{1 - \frac{v^2}{c^2}}}, t' = \frac{t - \frac{z}{c} \frac{v}{c}}{\sqrt{1 - \frac{v^2}{c^2}}} \quad (1.3)$$

wobei t : Zeit, v : Geschwindigkeit und c : Lichtgeschwindigkeit

Wegen des häufigen Vorkommen von v und c in den Formeln werden meist folgende Abkürzungen verwendet:

$$\beta = \frac{v}{c}, \gamma = \frac{1}{\sqrt{1 - \frac{v^2}{c^2}}}$$

β gibt also den Anteil an der Lichtgeschwindigkeit an

Die Transformation lässt sich auch in Matrixform schreiben. Mit den gerade eingeführten Abkürzungen erhält man:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \gamma & -\beta\gamma \\ 0 & 0 & -\beta\gamma & \gamma \end{pmatrix} \quad (1.4)$$

Vergleicht man diese Matrix mit einer Rotationsmatrix, so stellt man Ähnlichkeiten fest. Es zeigt sich tatsächlich, dass die Lorentz Transformation eine Rotation um einen imaginären¹ Winkel $i\theta$ ist[8]. Wie bei allen Rotationen, kann man auch die Lorentz Transformation um verschiedene Achsen rotieren. Die Achsen entsprechen in diesem Fall dem Richtungsvektor der relativen Geschwindigkeit \vec{v} der Systeme K und K' .

Wendet man das gleiche Prinzip an, um eine Rotation um eine beliebige Achse im 3-Dimensionalen Raum zu realisieren, so erhält man eine Lorentz Transformationsmatrix für beliebige Richtungen von \vec{v} . Sei nun \vec{n} der Einheitsvektor der Richtung von \vec{v} und $v = |\vec{v}|$, dann erhält man die folgende Matrix[13]:

$$\begin{pmatrix} (\gamma - 1)n_x^2 + 1 & (\gamma - 1)n_x n_y & (\gamma - 1)n_x n_z & -\beta\gamma n_x \\ (\gamma - 1)n_x n_y & (\gamma - 1)n_y^2 + 1 & (\gamma - 1)n_y n_z & -\beta\gamma n_y \\ (\gamma - 1)n_x n_z & (\gamma - 1)n_y n_z & (\gamma - 1)n_z^2 + 1 & -\beta\gamma n_z \\ -\beta\gamma n_x & -\beta\gamma n_y & -\beta\gamma n_z & \gamma \end{pmatrix} \quad (1.5)$$

1.2.1 Bemerkung zur Lorentz Transformation

Bei Geschwindigkeiten grösser als \vec{c} wird γ imaginär, da $1 < \frac{v^2}{c^2}$ ist und die Wurzel negativ wird. In einem solchen Fall würden also Raum und Zeit imaginär. Daher macht es keinen Sinn über Objekte mit Überlichtgeschwindigkeit im Rahmen der speziellen Relativitätstheorie zu sprechen.

1.3 Die Längenkontraktion

Ein Effekt der speziellen Relativitätstheorie, im Hinblick auf die Visualisierung auf dem Computer, ist die sogenannte Längenkontraktion.

¹Der imaginäre Winkel hat in diesem Fall weniger damit zu tun, dass die Koordinaten imaginär sind, sondern kommt vom Sinus-Hyperbolicus (es gilt nämlich: $\cosh \theta = \cos i\theta$ und $i \sinh \theta = \sin i\theta$)[8]

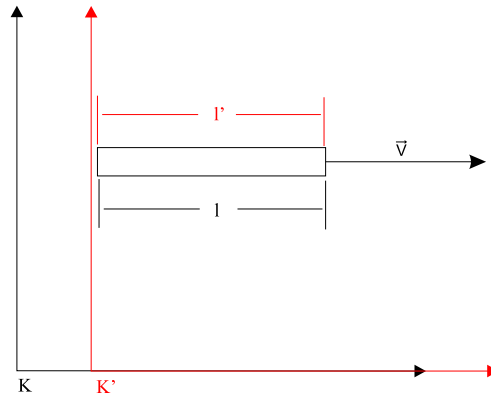


Abbildung 1.4: Längenkontraktion eines Stabes mit Geschwindigkeit \vec{v}

Ein Stab habe in Bewegungsrichtung die Länge l im K -System und die Länge l' im K' -System. Das K' -System bewege sich mit der Geschwindigkeit \vec{v} (Abb. 1.4). Mit Hilfe der Lorentz Transformation lässt sich herleiten:

$$l = l' \cdot \sqrt{1 - \beta^2} \quad (1.6)$$

Das heißt also, dass der bewegte Stab für den Beobachter im K -System kürzer erscheint als für den Beobachter im K' -System. Es sei aber an dieser Stelle schon vorweggenommen, dass man die Längenkontraktion in reiner Form nicht beobachtet. Näheres dazu ist im nächsten Kapitel zu finden.

1.3.1 Zahlenbeispiele zur Längenkontraktion

Wie zu Beginn dieses Kapitels bereits erwähnt, wirkt sich die Relativitätstheorie erst bei besonders hohen Geschwindigkeiten aus. Dies lässt sich mit der Längenkontraktion recht gut verdeutlichen. Die folgende Tabelle wurde mit einem Wert für die Lichtgeschwindigkeit von $c = 2.997925 \cdot 10^8 \text{ms}^{-1}$ ermittelt.

Beschreibung	Geschwindigkeit v	kontrahierte Länge l
Fußgänger	$5 \frac{\text{km}}{\text{h}}$	1 m
Auto	$100 \frac{\text{km}}{\text{h}}$	1 m
Rakete	$30000 \frac{\text{km}}{\text{h}}$	99.99999996 cm
Teilchen im Beschleuniger	$10\% \cdot c$	99.49874371 cm
	$25\% \cdot c$	96.82458366 cm
	$50\% \cdot c$	86.60254038 cm
	$75\% \cdot c$	66.14378278 cm
	$90\% \cdot c$	43.58898944 cm
	$99\% \cdot c$	14.10673598 cm
	$99.9\% \cdot c$	4.471017781 cm

Tabelle 1.1: Längenkontraktion bei einer Länge des Stabes von $l' = 1 \text{ m}$

1.4 Längenkontraktion bei Myonen

Als kleines Beispiel für die Längenkontraktion seien an dieser Stelle die Myonen erwähnt. Myonen sind subatomare Teilchen, die durch die kosmische Strahlung in der Atmosphäre in einer Höhe von ca. 20 km erzeugt werden. Sie sind mit einer Geschwindigkeit von $v = 99.98\% \cdot c$ fast so schnell wie das Licht (Abb. 1.5). Ihre Lebensdauer beträgt jedoch nur $t = 2.2 \cdot \mu\text{s}$.

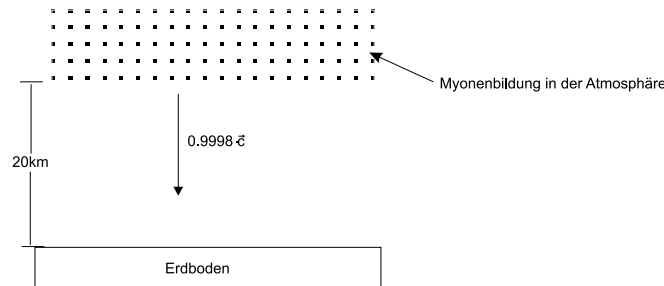


Abbildung 1.5: Myonen auf ihrem Weg zum Erdboden

Die Myonen können in ihrer Lebensdauer eine Strecke von $v \cdot t \approx 659\text{ m}$ zurücklegen. Bis zur Erde sind es jedoch 20 km . Somit sollten keine Myonen auf dem Erdboden ankommen. Messungen haben jedoch ergeben das noch zahlreiche Myonen den Erdboden erreichen.

Ein relativistischer Beobachter auf einem Myon nimmt an, dass sein Myon ruht und sich die Erde auf ihn zu bewegt. Wenn man sich weiter ein Maßband fest mit der Erde verbunden denkt, dann bewegt sich auch dieses Maßband mit der Geschwindigkeit der Erde auf das Myon zu. Wenn das Maßband von der Erde bis zum Myon reicht, muß es etwa $l' = 20\text{ km}$ lang sein. Da es sich aber mit fast Lichtgeschwindigkeit bewegt, ist die Länge des Maßbandes kontrahiert. Nach der Längenkontraktion ergibt sich:

$$l = l' \cdot \sqrt{1 - \beta^2} \approx 400\text{ m}$$

Aus Sicht des Myons sind es also nur 400 m bis zum Erdboden. Da ein Myon durchschnittlich 659 m schafft, kommen noch viele Myonen bis zum Boden. (S.701-702,[4])

Die Myonen sind also ein Beleg für die Richtigkeit der Relativitätstheorie.

1.5 Die Zeitdilatation

Da die Zeitdilatation implizit bei der Visualisierung beachtet werden soll und ausserdem eine grosse Rolle bei der Simulation der speziellen Relativitätstheorie spielt, sei die Zeitdilatation hier kurz erwähnt.

Bewegt sich das System K' mit einer Geschwindigkeit v gegenüber dem System K , so gilt für eine Zeitspanne Δt in K und der entsprechenden Zeitspanne $\Delta t'$ in K' (S.698-699 von [4]):

$$\Delta t' = \Delta t \frac{1}{\sqrt{1 - \beta^2}} \quad (1.7)$$

Aus der Zeitdilatation ergibt sich die berühmte Aussage: Bewegte Uhren gehen langsamer.

1.6 Die relativistische Masse

Neben der Längenkontraktion (Kapitel 1.3) und der Zeitdilatation (Kapitel 1.5) ist die relativistische Masse ein weiteres Ergebnis der speziellen Relativitätstheorie. Ruht eine Masse relativ zu einem Beobachter, so stellt der Beobachter die Ruhemasse m_0 des Objektes fest. Bewegt sich diese Masse nun mit einer Geschwindigkeit v relativ zum Beobachter, so stellt der Beobachter die relativistische Masse von $m(v)$ fest:

$$m(v) = m_0 \frac{1}{\sqrt{1 - \left(\frac{v}{c}\right)^2}} = m_0 \frac{1}{\sqrt{1 - \beta^2}} \quad (1.8)$$

Das Objekt erscheint also aus der Sicht des bewegten Beobachters schwerer.

1.7 Minkowski Raum-Zeit-Diagramme

Eine gute Darstellungsmöglichkeit relativistischer Vorgänge, die auch in den nachfolgenden Kapiteln noch eine Rolle spielen wird, sind die von Minkowski erstmals verwendeten Raum-Zeit-Diagramme. Bei diesen Diagrammen trägt man auf der Abszisse eine Ortskoordinate auf (*hier*: x) und als Ordinate ct (Abb. 1.6 a). (S.78 ff., [14])

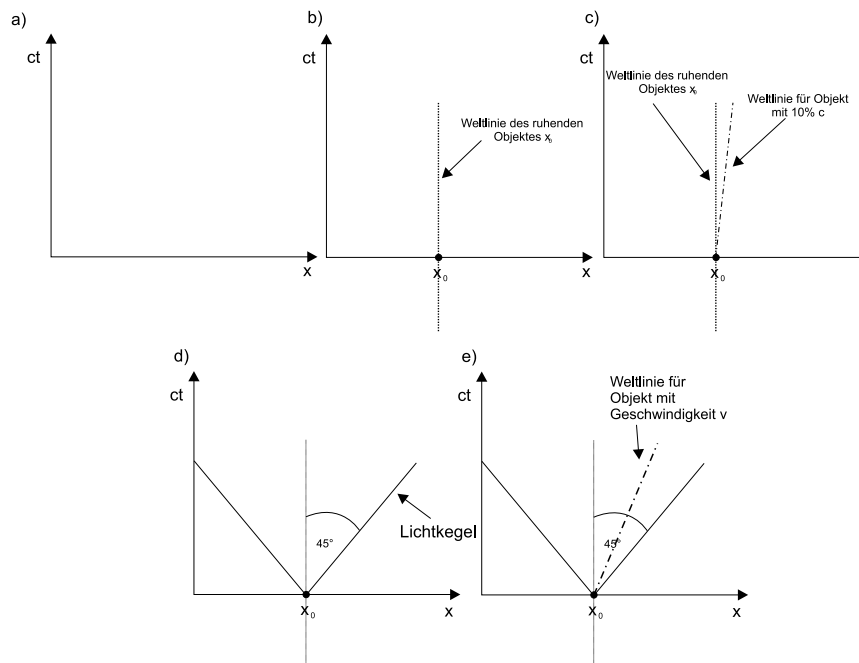


Abbildung 1.6: Minkowski Raum-Zeit-Diagramme

Ein Ereignis an einem Ort x_0 zu einem bestimmten Zeitpunkt t_0 lässt sich als Punkt in einem solchen Diagramm darstellen (Abb. 1.6 b). Ruht ein Objekt, so ändert sich seine Position nicht. Also bleibt die x -Koordinate für alle Zeitpunkte gleich. Daher wird ein ruhendes Objekt als parallele Gerade zur ct -Achse dargestellt. Eine solche Gerade nennt man auch die Weltlinie des Objektes

(Abb. 1.6 b).

Ein sich bewegendes Objekt ändert seine Position zu jedem Zeitpunkt, somit erhält man eine gedrehte Gerade zu einem ruhenden Objekt (Abb. 1.6c). Licht breitet sich in alle Richtungen mit Lichtgeschwindigkeit aus. Somit ändert sich die Ortskoordinate für einen Schritt auf der ct -Achse auch um 1, und zwar einmal nach links und einmal nach rechts. Im Minkowski-Diagramm ergibt sich daher ein Lichtkegel, dessen Geraden um 45° gedreht sind (Abb. 1.6 d).

Alle Objekte die zur gleichen Zeit am Punkt x_0 starten, können nur Weltlinien innerhalb dieses Lichtkegels besitzen (also Geraden mit Winkel kleiner gleich 45°). Da ihre Geschwindigkeit ansonsten grösser als die Lichtgeschwindigkeit wäre (Abb. 1.6 e).

Ein bewegtes Objekt besitzt (wie bereits weiter oben beschrieben) ein anderes raum-zeitliches Koordinatensystem als ein ruhendes. Auch dies lässt sich mit Minkowski-Diagrammen darstellen. Wollen wir zum Beispiel das Koordinatensystem K' des Objektes aus Abb. 1.6 c, das sich mit der Geschwindigkeit v fortbewegt, darstellen, müssen wir die ct -Achse und die x -Achse um den Winkel $\alpha = \arccos(\frac{v}{c})$ drehen. Es ergibt sich Abb. 1.7.

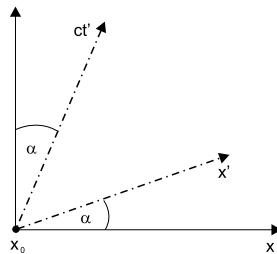


Abbildung 1.7: Ruhendes und bewegtes Koordinatensystem

1.7.1 Relativierung der Gleichzeitigkeit

Neben den bereits erwähnten Effekten ist ein Effekt, den man häufig als Relativierung der Gleichzeitigkeit bezeichnet, besonders gut mit Minkowski-Diagrammen darstellbar. In der klassischen Vorstellung geht man davon aus, dass die Zeitmessung von der Bewegung unabhängig ist und somit Ereignisse, die im ruhenden System gleichzeitig sind, auch im bewegten System gleichzeitig sind. Man geht in der klassischen Physik also von einer global gültigen Zeit aus. Diese Vorstellung ist jedoch nach der Relativitätstheorie falsch. Es gibt in der relativistischen Welt keine globale Zeit.

Dieser doch sehr abstrakte Gedanke soll nun zunächst an Minkowski-Diagrammen veranschaulicht werden. Alle Ereignisse auf einer Parallelen zur x -Achse finden gleichzeitig statt. Es ergeben sich jedoch verschiedene Parallelen in einem ruhenden System K und einem bewegten System K' (Abb. 1.8). Somit sind Ereignisse, die in K' gleichzeitig sind, in K normalerweise nicht gleichzeitig. (S.79, [14])

An dieser Stelle sei angemerkt, dass Ereignisse die zur gleichen Zeit am gleichen Ort stattfinden, in jedem System gleichzeitig sind. Im Minkowski-Diagramm lägen diese Ereignisse auf einem Punkt und somit immer auf einer Parallelen zur x -Achse.

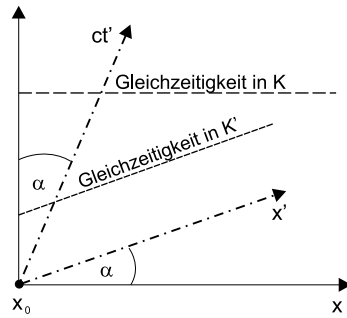


Abbildung 1.8: Relativierung der Gleichzeitigkeit im Minkowski-Diagramm

Da diese Vorstellung nicht besonders intuitiv ist, hier nochmal ein kleines Beispiel:
Wir betrachten wieder einen Zug, der mit der Geschwindigkeit $\frac{1}{2} \cdot c$ am Bahnsteig vorbeifährt. Der Beobachter A sei am Bahnsteig und Beobachter B bewege sich mit dem Zug. Im Zug sind an beiden Enden Blitzlampen angebracht. Nun sollen beide Blitzlampen für Beobachter B im Zug gleichzeitig einen Blitz abgeben (Abb. 1.9).

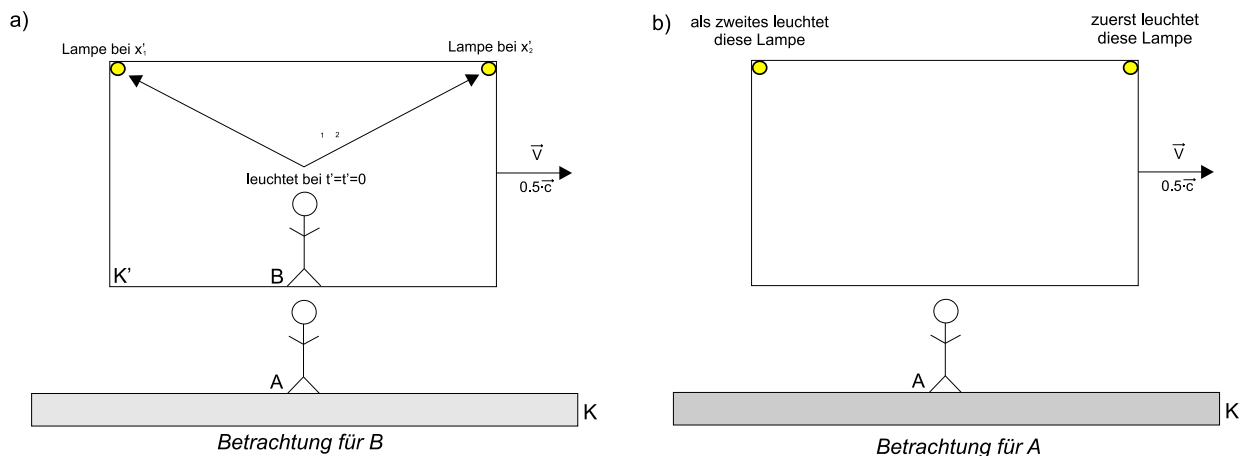


Abbildung 1.9: Relativierung der Gleichzeitigkeit im Zug

Wenn B in der Mitte des Zuges ist, erreichen ihn beide Lichtimpulse zur gleichen Zeit. Nach klassischer Ansicht würde nun auch Beobachter A am Bahnsteig beide Lichtimpulse gleichzeitig sehen. Für die relativistische Betrachtungsweise muss man allerdings mit der Lorentz-Transformation rechnen. Nehmen wir an die Lampen haben in K' die Positionen x'_1 und x'_2 . Da die Lampen in K' gleichzeitig aufleuchten, gilt für die Zeit in K' : $t'_1 = t'_2$ (Abb. 1.9 a). Es gilt nun die Zeiten t_1 und t_2 in K zu ermitteln. Es gilt:

$$t_1 = \frac{t'_1 - \frac{v}{c^2} \cdot x'_1}{\sqrt{1 - \beta^2}}, \quad t_2 = \frac{t'_2 - \frac{v}{c^2} \cdot x'_2}{\sqrt{1 - \beta^2}}$$

Würden die beiden Lampen auch für A gleichzeitig leuchten, so müsste die Differenz 0 ergeben. Es

gilt aber:

$$t_2 - t_1 = \frac{-\frac{v}{c^2} \cdot (x_2' - x_1')}{\sqrt{1 - \beta^2}}$$

Da x_1' und x_2' aber nicht gleich sind, ist diese Gleichung ungleich 0. Somit vergeht eine gewisse Zeit zwischen dem Aufleuchten der beiden Lampen. Fährt der Zug also von links nach rechts, sieht der Beobachter A zuerst die rechte Blitzlampe aufleuchten (Abb. 1.9 b). Fährt der Zug in entgegengesetzter Richtung, sieht A die linke zuerst.

1.7.2 Beschleunigte Bewegung im Minkowski-Diagramm

In der Realität sind unbeschleunigte Bewegungen eher selten. Im Minkowski-Diagramm erkennt man ein beschleunigtes Objekt daran, dass die Weltlinie keine Gerade, sondern eine Kurve ist (Abb. 1.10). Da auf dieses Thema später noch einmal zurückgegriffen wird, wird an dieser Stelle noch auf weitere Einzelheiten verzichtet.

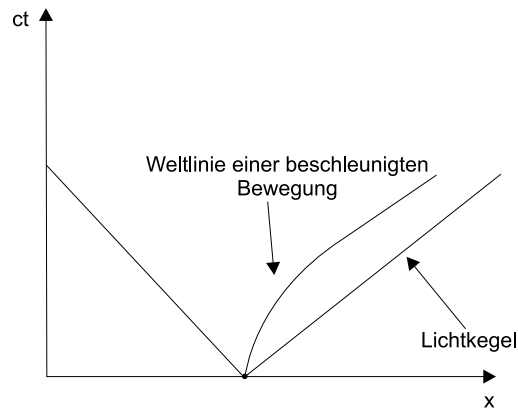


Abbildung 1.10: Beschleunigte Bewegung im Raum-Zeit-Diagramm

1.8 Allgemeine Relativitätstheorie

Die spezielle Relativitätstheorie beinhaltet nur einen Teil der relativistischen Denkweise. In der allgemeinen Relativitätstheorie wird zusätzlich ein Zusammenhang zwischen Gravitation, Masse, Beschleunigung und Zeit hergestellt. Da die Allgemeine Relativitätstheorie aber nicht Gegenstand dieser Studienarbeit sein soll, wird an dieser Stelle auch nicht näher darauf eingegangen.

Kapitel 2

Visuelle Effekte

Die visuellen Auswirkungen der speziellen Relativitätstheorie beschränken sich nicht nur auf die schon im ersten Kapitel angedeuteten Effekte (wie z.B. die Längenkontraktion). Es gibt noch andere wichtige Effekte, die bei der Darstellung schnell bewegter Objekte von Bedeutung sind. Über diese Effekte soll dieses Kapitel einen kleinen Überblick geben.

2.1 Effekte der Lorentz Transformation

Im ersten Kapitel wurde bereits die Längenkontraktion erklärt. Diese ist sicherlich ein darstellbarer relativistischer Effekt. An dieser Stelle sei jedoch schon erwähnt, dass die Längenkontraktion, so wie im ersten Kapitel, niemals zu sehen sein wird, da sich dieser Effekt mit anderen Effekten (vor allem der Lichtlaufzeit, s.u.) überlagert.

2.2 Auswirkungen der endlichen Lichtlaufzeit

Obwohl bereits im 17. Jh. bekannt war, dass die Lichtgeschwindigkeit endlich ist, haben die Auswirkungen auf schnell bewegte Objekte erst Mitte des 19. Jh. Beachtung gefunden. Sogar Einstein selbst hat den Fehler gemacht und nur die Längenkontraktion für die Beschreibung schnell bewegter Objekte herangezogen. Dies liegt vermutlich daran, dass lange Zeit kein Unterschied zwischen Messen und Beobachten gemacht worden ist. Während beim Messen nur die Distanz zwischen Punkten im jeweiligen Bezugskordinatensystem des Messenden zu ermitteln ist, ist beim Beobachten entscheidend wann welches Lichtquant und von wo es beim Beobachter eintrifft. Somit sind die Effekte der Lichtlaufzeit, wie im folgenden genauer dargestellt werden soll, genauso wichtig für die Darstellung von Objekten mit fast Lichtgeschwindigkeit, wie die Längenkontraktion.

Bei kleinen Geschwindigkeiten kann man, aufgrund der hohen Lichtgeschwindigkeit, davon ausgehen, dass das Licht von allen Punkten des Objekts gleichzeitig ankommt. Bei Geschwindigkeiten nahe der Lichtgeschwindigkeit muss man jedoch die verschiedenen Zeiten, die das Licht von den verschiedenen Punkten des Objekts braucht, um zum Beobachter zu gelangen, mit einrechnen. Aufgrund der Wichtigkeit dieses Effektes für die Visualisierung von schnell bewegten Objekten, soll auf die Auswirkungen der endlichen Lichtlaufzeit anhand eines kleinen Beispiels näher eingegangen

werden.

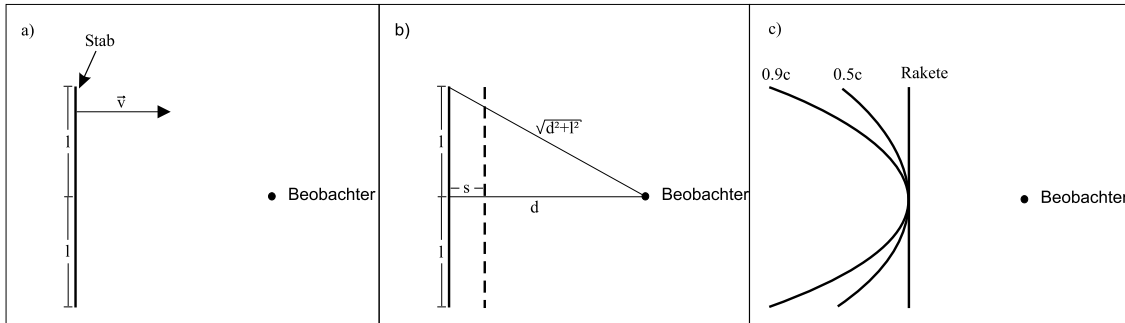


Abbildung 2.1: Aussehen eines schnellen Stabes bei Einbeziehung der Lichtlaufzeit

Man stelle sich einen Stab mit der Länge $2 \cdot l$ vor. Dieser habe eine Geschwindigkeit \vec{v} , die senkrecht zur Stablänge verläuft. Der Beobachter B sieht den Stab mittig auf sich zu kommen (vgl. Abb. 2.1a).

Zum Verständnis der Auswirkungen der Lichtlaufzeit mache man sich zunächst klar, dass ein Lichtquant von der Mitte des Stabes zum Auge eine kleinere Strecke zurücklegen muss, als ein Lichtquant von irgendeinem anderen Punkt des Stabes. Wie in Abb. 2.1 b zu sehen ist, muss ein Lichtquant in der Mitte des Stabes eine Strecke d zurücklegen, während ein Lichtquant am oberen Ende des Stabes die längere Strecke $\sqrt{d^2 + l^2}$ zurücklegen muss. Da sich beide Lichtquanten mit der gleichen Geschwindigkeit c bewegen, muss das aus der Mitte des Stabes kommende Lichtquant früher beim Beobachter ankommen, als das Lichtquant vom oberen Ende des Stabes.

Während sich das Lichtquant vom oberen Ende des Stabes zum Beobachter bewegt, bewegt sich der Stab ebenfalls weiter. Somit erreicht den Beobachter beim Eintreffen des Lichtes vom oberen Ende, Licht von einem um die Strecke s nach rechts bewegten Stab aus der Mitte. Da der Beobachter beide Lichtquanten zur gleichen Zeit empfängt, muss die Zeit t für das Lichtquant vom oberen Ende bis zum Beobachter gleich der Zeit, der Bewegung des Stabes um s und der Zeit des Lichtquanten von der neuen Position zum Beobachter, sein. Es muss daher gelten:

$$\frac{\sqrt{d^2 + l^2}}{c} = \frac{s}{v} + \frac{d - s}{c}$$

Durch Umformung nach s erhält man:

$$s = \frac{\sqrt{d^2 + l^2} - d}{\frac{c}{v} - 1} \quad (2.1)$$

Die Gleichung (2.1) beschreibt die Verzerrung des Stabes durch die Lichtlaufzeit. Variiert man die Länge l , so erhält man eine Hyperbel. Der Stab sieht also für den Beobachter bei hohen Geschwindigkeiten gekrümmt aus (Abb. 2.1 c). (S.70-71, [14])

Diese Betrachtung zeigt, dass die Lichtlaufzeit für die Betrachtung schnell bewegter Objekte nicht zu vernachlässigen ist. Obwohl dies eigentlich kein relativistischer Effekt ist, sondern bereits eine Erkenntnis aus der klassischen Physik darstellt, können die Auswirkungen für die Darstellung von schnell bewegten Objekten nicht vernachlässigt werden.

2.3 Kombination aus Längenkontraktion und Lichtlaufzeit

Zur Darstellung der speziellen Relativitätstheorie ist zunächst das geometrische Aussehen am interessantesten. Da die Form eines Objektes sowohl durch die Längenkontraktion als auch durch die Lichtlaufzeit beeinflusst wird, hat das Zusammenspiel beider Effekte eine besondere Bedeutung.

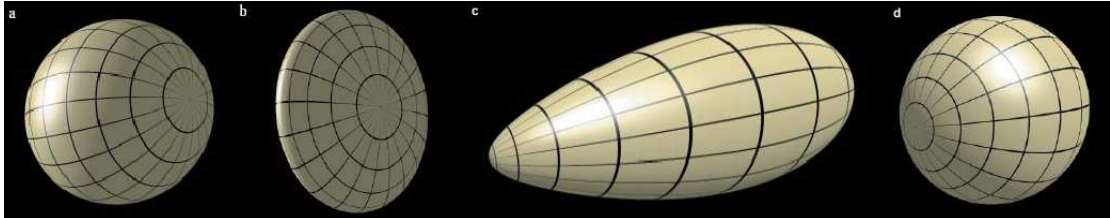


Abbildung 2.2: Darstellung einer Kugel unter Berücksichtigung verschiedener Effekte[9]

Am Beispiel einer Kugel soll nun das wahre geometrische Aussehen schnell bewegter Objekte erläutert werden. In Abb. 2.2 a ist eine Kugel in Ruhe zusehen. Unter Berücksichtigung der Längenkontraktion erscheint die Kugel bei 95% der Lichtgeschwindigkeit als Ellipsoid (Abb. 2.2 b). Eine gegensätzliche Wirkung hat die endliche Lichtlaufzeit. Berücksichtigt man nur den Effekt der Lichtlaufzeit, erscheint die Kugel gestreckt und ähnelt einem Zepplin (Abb. 2.2 c). Desweiteren ist bemerkenswert, dass die Kugel aus einem anderen Blickwinkel zu sehen ist. Sie erscheint gedreht.

Nimmt man beide Effekte zusammen, so gleichen sich in diesem Fall die geometrischen Verzerrungen der Längenkontraktion und der Lichtlaufzeit gerade aus. Die Drehung bleibt aber. Somit erscheint die Kugel nur gedreht (Abb. 2.2 d). [9]

Dass sich die Längenkontraktion und die Lichtlaufzeit hier (Abb. 2.2) gerade ausgleichen ist reiner Zufall. Im Allgemeinen gilt für ein Objekt, dass auf den Beobachter zukommt (S.72-73, [14]):

$$l = l' \cdot \sqrt{\frac{1 + \beta}{1 - \beta}} \quad (2.2)$$

2.4 Relativistischer Doppler-Effekt

Der Doppler-Effekt ist wohl am bekanntesten aus dem Bereich der Akustik. Die Tonhöhe eines sich nähernden Martinshorn ist zunächst erhöht; ist das Fahrzeug mit dem Martinshorn am Beobachter vorbei, so sinkt die Tonhöhe.

Da Licht eine elektromagnetische Welle ist, lässt sich dieser Effekt auf Licht übertragen. Bei einer Relativbewegung zwischen einer Lichtquelle und einem Beobachter sieht der Beobachter eine andere Wellenlänge λ als die ursprünglich ausgesandte Wellenlänge λ_0 .

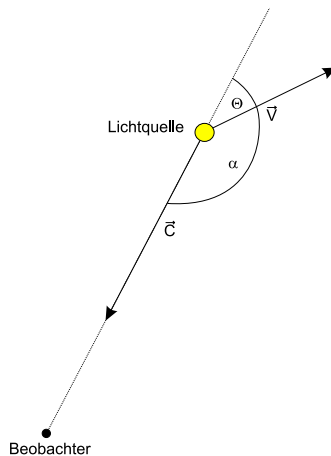


Abbildung 2.3: Relativistischer Doppler-Effekt

Eine Lichtquelle bewege sich, relativ zu einem Beobachter, mit der Geschwindigkeit \vec{v} in eine bestimmte Richtung. Für den Beobachter sind nur solche Lichtquanten interessant, die auch bei ihm ankommen. Die Ausbreitungsrichtung dieser Lichtquanten entspricht der Richtung von der Lichtquelle zum Beobachter. Die Ausbreitungsrichtung und die Richtung der Relativbewegung \vec{v} schließen den Winkel α ein (Abb. 2.3). Wegen $\Theta = \pi - \alpha$ erhält man für den relativistischen Doppler-Effekt [16]:

$$\lambda = \frac{1 - \beta \cos \Theta}{\sqrt{1 - \beta^2}} \lambda_0 \quad (2.3)$$

Ein Astronaut, der mit fast Lichtgeschwindigkeit an der Sonne vorbeifliegt, würde die Sonne also nicht in einer Farbe sehen, sondern in unterschiedlichen Farben, die abhängig vom Winkel zum Lichtaussendungsort sind (Abb. 2.4).



Abbildung 2.4: Vorbeiflug mit $\beta = 99,9\%$ der Lichtgeschwindigkeit an der Sonne unter Berücksichtigung des Doppler-Effektes[9]

Ähnlich wie bei der Längenkontraktion ist jedoch der reine Doppler-Effekt, so wie in Abbildung 2.4 gezeigt, niemals zu sehen. Der Doppler-Effekt wird nämlich durch die Aberration überlagert (s.u.). Dadurch kommen zu den Farbänderungen noch Helligkeitsänderungen, die im Falle des Vorbeifluges an der Sonne, das Bild dominieren würden (vgl. Abb. 2.5). [9]

2.5 Relativistische Aberration

Unter der Aberration versteht man, dass eine Lichtquelle an einer anderen Position stehend erscheint. Sendet eine Lichtquelle einen Lichtstrahl unter dem Winkel Θ aus, so empfängt ein Beobachter ihn unter dem Winkel Θ' . Mit Hilfe einer hier nicht weiter dargestellten relativistischen Rechnung ergibt sich der Zusammenhang:

$$\cos \Theta' = \frac{\cos \Theta + \beta}{1 + \beta \cos \Theta} \quad (2.4)$$

Das Licht erscheint also mit zunehmendem β immer mehr abgelenkt, bis bei Lichtgeschwindigkeit, alles von einem Punkt zu kommen scheint. Durch die Aberration wird also die räumliche Verteilung des Lichtes (und damit auch dessen Intensität) für den Beobachter verändert. Dies geht aber auch einher mit einer Veränderung der Helligkeitsverteilung. [16]

Bei Berücksichtigung dieser Erscheinung, beim Vorbeiflug an der Sonne, ergibt sich somit ein dunkel zu hell Übergang, der den Doppler-Effekt fast vollständig überdeckt (Abb. 2.5). [9]

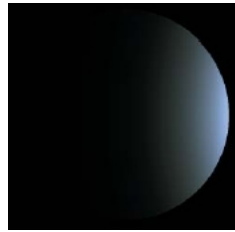


Abbildung 2.5: Vorbeiflug mit $\beta = 99,9\%$ der Lichtgeschwindigkeit an der Sonne unter Berücksichtigung des Doppler-Effektes und der Aberration[9]

2.5.1 Aberration und Geschwindigkeitsänderungen

Während die bisherigen Effekte bereits bei konstanter Geschwindigkeit sichtbar sind, gibt es auch Effekte die nur bei Geschwindigkeitsänderungen zu sehen sind. Ändert sich die Geschwindigkeit, so muss sich nach Gl. (2.4) auch der Winkel ändern. Dies führt dazu, dass die Szene für einen beschleunigenden Beobachter bei hohen Beschleunigungen ruhend oder gar rückwärts bewegt erscheint.

Um diesen Effekt besser verständlich zu machen, stelle man sich eine einfache Szene, bestehend aus einem Würfel und einem Beobachter vor (Abb. 2.6).

Würfel und Beobachter bewegen sich mit einer Geschwindigkeit v aufeinander zu. Zu Beginn sei $v = 0$. Dann ändert sich auch der Winkel der Lichtstrahlen nicht. Der Beobachter sieht also einen Würfel, der sich am Rande seines Sichtfeldes befindet (Abb. 2.6a). Beschleunigt der Beobachter nun auf $v = 0.5c$, so ändert sich scheinbar die Position des Würfels. Aus Gl. (2.4) lässt sich berechnen, dass der Würfel, der vorher unter einem 45° Winkel zu sehen war, nun unter einem Winkel von ca. 27° zu sehen ist (Abb. 2.6b). Der Würfel rückt also mehr in das Sichtfeld hinein.

Betrachtet man nun noch die einzelnen Eckpunkte des Würfels, so müssen diese näher zusammenrücken. Dies kann man sich gut vorstellen, wenn man annimmt, dass der Würfel und andere mit ihm bewegte Objekte bei einer Geschwindigkeit von $v = 0$ im Sichtfeldbereich von 0° bis 45° Platz finden. Erhöht man nun aber die Geschwindigkeit auf $v = 0.5c$, so muss der Würfel und alle anderen Objekte des alten Sichtbereiches in einen Sichtfeldbereich von 0° bis 27° passen. Somit

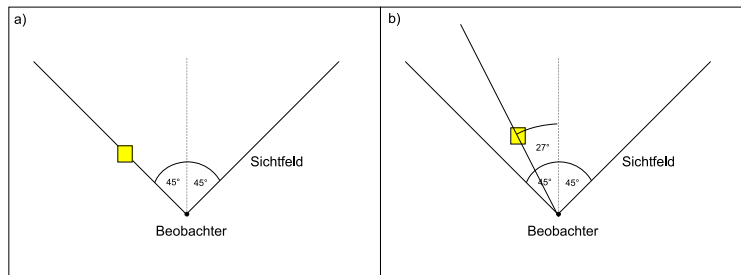


Abbildung 2.6: Auswirkungen der Aberration bei Geschwindigkeitsänderungen

müssen also die Eckpunkte zusammenrücken und das Objekt (der Würfel) kleiner werden, damit es (er) in den neuen Sichtbereich passt.

Ein menschlicher Beobachter, der sich an seine sonst klassische Umwelt gewöhnt hat, kennt ein sich ins Sichtfeld bewegendes Objekt, dass kleiner wird, nur von der Rückwärtsbewegung und interpretiert den Verlauf der Szene somit nicht als Beschleunigung aufeinander zu, sondern als Bewegung voneinander weg.

Im obigen Beispiel wurde nur die Aberration betrachtet. Streng genommen ändert sich der Winkel nicht nur durch die Aberration, sondern auch durch die beschleunigte Bewegung. Wählt man jedoch weit genug entfernte Objekte, so ist die Winkeländerung durch die Bewegung sehr klein gegenüber der Winkeländerung durch die Aberration.

2.6 Sonstige Konsequenzen

Neben den bereits erwähnten Effekten gibt es natürlich noch zahlreiche andere Konsequenzen für schnell bewegte Objekte.

Durch die endliche Lichtlaufzeit ergeben sich zum Beispiel Besonderheiten bei Schatten schnell bewegter Objekte. Ein mit 93% der Lichtgeschwindigkeit schnelles Rad, rollt beispielsweise seinem Schatten voraus (blaues Rad in Abb. 2.7). [9]

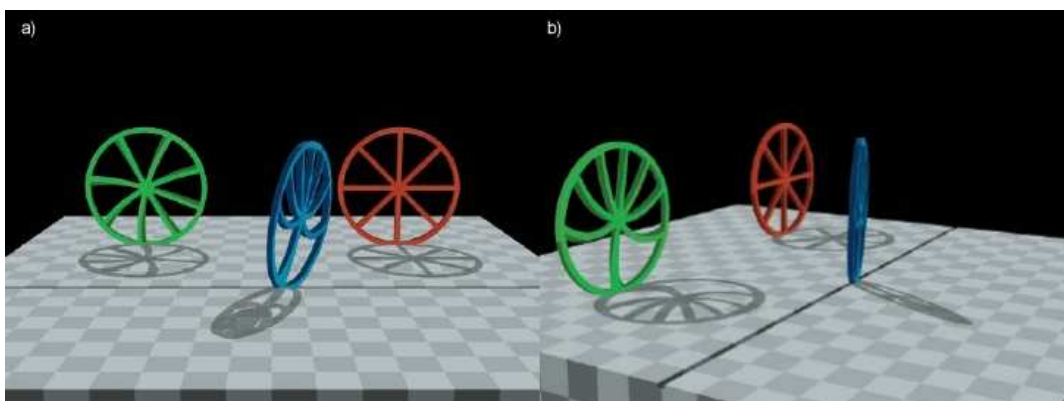


Abbildung 2.7: Rollende Räder bei fast Lichtgeschwindigkeit[9]

Ähnlich wie auch bei der Betrachtung schnell bewegter Objekte geometrische Verzerrungen auftreten, gibt es diese auch für die Schatten. Wobei die Verzerrungen des Schattens nicht mit denen des Betrachters übereinstimmen müssen, da der Schatten einer ganz anderen Betrachtungsposition entspricht, wie der Position des Betrachters. Dadurch entstehen Schattenbilder, die nicht immer dem Bild des Objektes entsprechen müssen (grünes Rad in Abb. 2.7 b). [9]

2.7 Effekte der Allgemeinen Relativitätstheorie

Die Darstellung der Allgemeinen Relativitätstheorie kann und soll nicht das Thema dieser Studienarbeit sein, da sie den Rahmen bei Weitem sprengen würde. Dennoch soll, hier am Beispiel von Gravitationslinsen, der Einfluss der Allgemeinen Relativitätstheorie auf das Licht und somit auf eine korrekte Visualisierung kurz dargestellt werden.

Nach der Allgemeinen Relativitätstheorie wird die Raum-Zeit durch grosse Massen (z.B. die Sonne oder Galaxien) gekrümmt. Somit wird auch ein Lichtstrahl in diesem Raumabschnitt gekrümmt. Somit haben grosse Massen ähnliche Auswirkungen wie Linsen in der Optik.

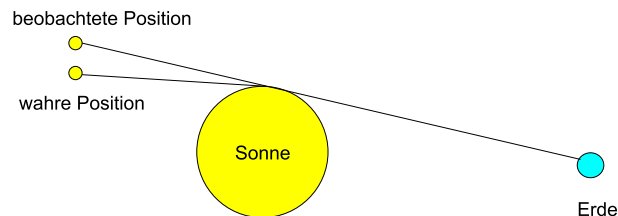


Abbildung 2.8: Sonne als Gravitationslinse

Eine solche Gravitationslinse ist beispielsweise die Sonne. Durch die Linsenwirkung der Sonne werden Sterne sichtbar, die von der Erde aus eigentlich nicht zu sehen wären (Abb. 2.8). [12]

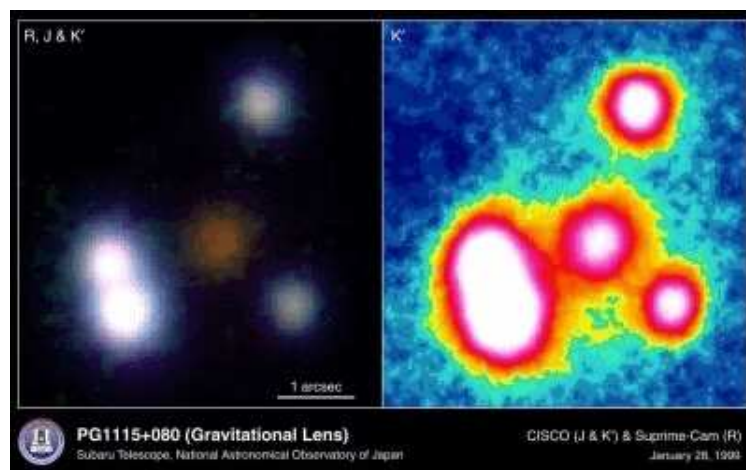


Abbildung 2.9: Quasar der durch die Linsenwirkung einer Galaxie vierfach erscheint[10]

Ein weiterer Effekt ist die Vervielfachung des Bildes eines Sterns. Unter bestimmten Umständen

wird das Licht durch die Gravitation derart abgelenkt, dass verschiedene Bilder eines Sternes beim Beobachter zu sehen sind. Diese Bilder erscheinen nicht nur an verschiedenen Positionen, sondern sind sogar Bilder von verschiedenen Zeitperioden des Sterns (Abb. 2.9). [3]

Kapitel 3

Visualisierungsalgorithmen

Für die Visualisierung relativistischer Effekte stehen bereits einige Algorithmen zur Verfügung. Die Wichtigsten sollen im folgenden näher erläutert werden. Dazu soll zunächst das Prinzip des entsprechenden Algorithmus grob erklärt werden und dann auf die wichtigsten algorithmischen Aspekte näher eingegangen werden. Erst dann soll der eigentliche Algorithmus dargeboten werden. Dabei soll insbesondere auf die geometrischen Effekte eingegangen werden. Obwohl der Doppler-Effekt und der Suchscheinwerfer-Effekt ebenfalls wichtige visuelle Effekte sind, sollen diese Effekte nur eine Nebenrolle spielen.

Die bekannten klassischen Visualisierungsalgorithmen lassen sich erstaunlicherweise meist derart erweitern, dass sie auch den relativistischen Fall behandeln können. Dazu wird die Szene in verschiedene Bezugssysteme aufgeteilt, so dass es für jedes Objekt ein System gibt, in dem das Objekt ruht. Für die Visualisierung werden die Koordinaten dann mit der Lorentz-Transformation zwischen den Systemen transformiert. Dieses Prinzip lässt sich sowohl für relativistisches Ray-Tracing und Polygon-Rendering, wie auch für ein relativistisches Bild-basiertes Verfahren anwenden, wie in den folgenden Abschnitten gezeigt wird.

3.1 Ray-Tracing Algorithmen

3.1.1 Grundlegendes zum Ray-Tracing

Bevor auf das relativistische Ray-Tracing näher eingegangen wird, hier zunächst eine kurze Einführung ins klassische Ray-Tracing.

Beim Ray-Tracing wird auf eine Bildebene (vgl. Abb. 3.1) ein 2-dimensionales Bild von Objekten projiziert. Dazu wird aus einem Projektionszentrum durch die Punkte der Bildebene ein Lichtstrahl in den Objektraum zurückverfolgt. Man kann somit einen einfachen Ray-Tracing Algorithmus wie folgt beschreiben:

1. Berechne Strahl.
2. Suche Objekt, dass als erstes vom Strahl getroffen wird.
3. Setze Farbe des Bildpunktes, auf die Farbe des Objektes; wird kein Objekt getroffen, verwende Hintergrundfarbe.

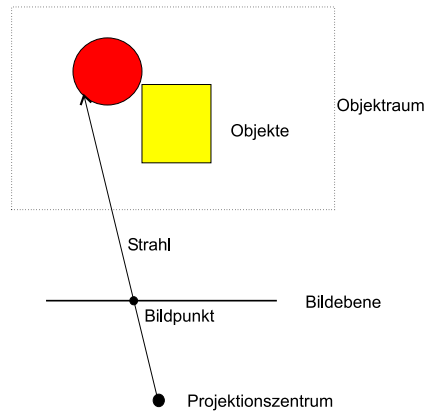


Abbildung 3.1: Prinzip des Ray-Tracing

In den meisten Fällen wird dieser einfache Algorithmus noch dadurch erweitert, dass statt des 3-ten Punktes der Strahl in 3 Teile aufgespalten wird: Farbanteil des Objektes, transparenter Anteil und Reflektionsanteil. Der transparente Anteil und der Reflektionsanteil wird durch zwei neue Strahlen ermittelt.

Rückverfolgung des Lichtstrahls

Um den Lichtstrahl verfolgen zu können, muss dieser in Form einer Geradengleichung vorliegen:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = P_Z + t\vec{a} \quad (3.1)$$

Der Punkt P_Z entspricht dem Projektionszentrum (vgl. Abb. 3.1), den Richtungsvektor \vec{a} erhält man, indem man die Differenz der 3-dimensionalen Koordinaten eines Bildpunktes und der Koordinaten des Projektionszentrums (P_Z) bildet (die Richtung des Strahls ist somit genau umgekehrt zu der eines realen Lichtstrahls). Beim klassischen Ray-Tracing entspricht t einem Parameter und nicht etwa der Zeit, wie später beim relativistischen Ray-Tracing. Für $t = 1$ erhält man, bei nicht normiertem Richtungsvektor \vec{a} , gerade den Schnittpunkt mit der Bildebene. Somit liegen alle Punkte mit $t < 1$ hinter der Bildebene und müssen bei der Berechnung der Schnittpunkte übergangen werden.

3.1.2 Ray-Tracing nach Hsiung und Dunn

Die meisten relativistischer Ray-Tracer basieren auf dem von HSIUNG UND DUNN 1989 vorgestellten Algorithmus zur Visualisierung relativistic effects in spacetime (REST) [5].

In ihrem Algorithmus gibt es einen Beobachter, der in seinem S -Koordinatensystem ruht. Die Objekte ruhen in ihrem System S' . Das S' -System bewegt sich relativ zum S -System des Beobachters mit der Geschwindigkeit \vec{v} entlang der x -Achse (Abb. 3.2).

Da Lichtstrahlen auch im Sinne der speziellen Relativitätstheorie durch Geraden im Raum beschrieben werden können, kann man, wie auch in einem klassischem Ray-Tracer, den Strahl in

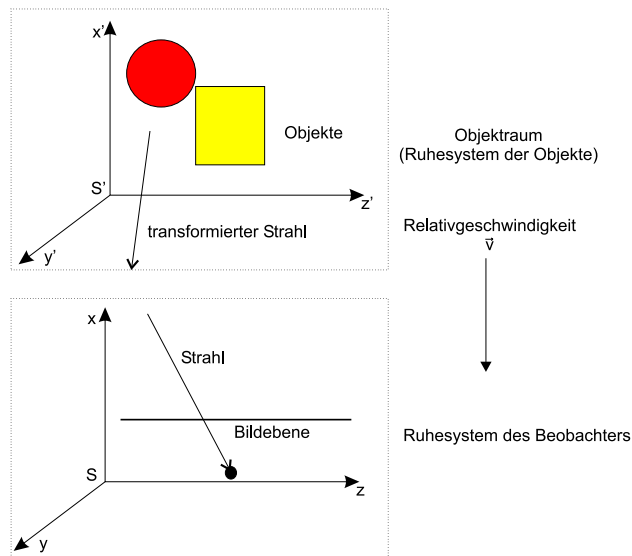


Abbildung 3.2: Modifiziertes Ray-Tracing nach HSIUNG UND DUNN

Form einer Geradengleichung ausdrücken (vgl. Gl. (3.1)). In einem klassischen Ray-Tracer können nun Schnittpunkte zwischen Objekten im Objektraum und dem Strahl berechnet werden. In einem relativistischen Ray-Tracer befinden sich Strahl und Objekte jedoch zunächst in zwei verschiedenen Systemen. In HSIUNGS UND DUNNS Ray-Tracer wird der Strahl daher ins S' -System transformiert. Dies hat den Vorteil, dass alle Objekte gegenüber dem Strahl ruhen und daher der Effekt der Lichtlaufzeit nicht kompliziert über die Schnittpunkte berechnet werden muss. Diesem Effekt wird bereits bei der Transformierung des Strahls indirekt Rechnung getragen. Befindet sich der Strahl erstmal im Objektraum, so können, nach den Verfahren des klassischen Ray-Tracing, die Schnittpunkte berechnet werden. HSIUNG UND DUNN erweitern also den klassischen Ray-Tracing Algorithmus um einen Schritt:

1. Berechne Strahlgleichung im S -System.
- 2. Transformiere Strahlgleichung in das S' -System**
3. Suche Objekt, dass als erstes vom Strahl getroffen wird.
4. Setze Farbe des Bildpunktes, auf die Farbe des Objektes; wird kein Objekt getroffen, verwende Hintergrundfarbe.

Transformation der Strahlgleichung vom S -System in das S' -System

Um die Strahlgleichung transformieren zu können muss die Gl. (3.1) etwas ausführlicher dargestellt werden:

$$P_Z = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}, \vec{a} = \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} \Rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} + c(t_0 - t) \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} \quad (3.2)$$

Um die folgenden Ausdrücke zu vereinfachen normieren HSIUNG UND DUNN die Lichtgeschwindigkeit auf $c = 1$. Somit wird der Bruchteil der Lichtgeschwindigkeit $\beta = \frac{v}{c}$ gleich der relativen Geschwindigkeit v ($\beta = v$). Ausserdem sei der Richtungsvektor \vec{a} normiert. Im Gegensatz zu Gl. (3.1) haben HSIUNG UND DUNN den Lichtstrahl in seiner realen Richtung ausgedrückt. Daher ist hier der Parameter nicht $+t$ sondern $-t$. Dies drückt zudem physikalisch korrekt aus, dass der Strahl in der Zeit zurückverfolgt wird (t entspricht hier auch der Zeit). Dabei ist der Lichtstrahl zum Zeitpunkt t_0 gerade am Projektionszentrum (P_Z) angekommen. Zur Vereinfachung wird im folgenden von $t_0 = 0$ ausgegangen.

Da sich der Objektraum und die Bildebene relativ zueinander in x-Richtung bewegen, muss auf die Geradengleichung des Strahls die Lorentz-Transformation angewendet werden. Für die Transformation von Koordinaten zur Zeit t_0 bzw. t'_0 der Systeme S und S' (siehe auch Kapitel 1.2) gilt:

$$\begin{pmatrix} x'(t'_0) \\ y'(t'_0) \\ z'(t'_0) \\ t'_0 \end{pmatrix} = \begin{pmatrix} \gamma(x_0 - \beta t_0) \\ y_0 \\ z_0 \\ \gamma(t_0 - \beta x_0) \end{pmatrix} \quad (3.3)$$

Mit den zuvor gemachten Annahmen ($t_0 = 0$ und $c = 1$) lässt sich somit der erste Teil der Geradengleichung transformieren:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \gamma x_0 \\ y_0 \\ z_0 \end{pmatrix} - (\gamma \beta t_0 + t') \begin{pmatrix} a'_x \\ a'_y \\ a'_z \end{pmatrix} \quad (3.4)$$

Da die Geradengleichung Lichtstrahlen beschreibt, kann man zur Transformation des Richtungsvektors die Aberrationsgleichung zur Hilfe nehmen (siehe auch Kapitel 2.5). Dies ist bedeutend einfacher als die Transformation mit Hilfe der Lorentz-Transformation durchzuführen. Man erhält für den Richtungsvektor \vec{a} :

$$\begin{pmatrix} a'_x \\ a'_y \\ a'_z \end{pmatrix} = \frac{1}{\gamma(1 + \beta a_x)} \begin{pmatrix} \gamma(a_x + \beta) \\ a_y \\ a_z \end{pmatrix} \quad (3.5)$$

Setzt man den transformierten Vektor \vec{a} in Gl. (3.4) ein, so erhält man die komplett transformierte Geradengleichung:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \gamma x_0 \\ y_0 \\ z_0 \end{pmatrix} - (\gamma \beta t_0 + t') \begin{pmatrix} \frac{a_x + \beta}{1 + \beta a_x} \\ \frac{a_y}{\gamma(1 + \beta a_x)} \\ \frac{a_z}{\gamma(1 + \beta a_x)} \end{pmatrix} \quad (3.6)$$

Laufzeit des Ray-Tracing

Für die Laufzeitanalyse mache man sich zunächst klar, dass Ray-Tracing maßgeblich abhängig ist von der Bildauflösung ($\text{width} \times \text{height}$), da für jeden Bildpunkt eine Farbe berechnet werden muss. Zur Berechnung der Farbe muss ein Lichtstrahl zurückverfolgt werden. Dieses Problem nennt man auch Ray-Casting. Die Laufzeit für das Ray-Casting ist von der Anzahl n der Objekte in der Szene abhängig. Diese Struktur ergibt sich sowohl für das klassische als auch für das relativistische Ray-Tracing. Allgemein lässt sich daher die Laufzeit wie folgt darstellen:

$$\text{Ray}_{\text{Laufzeit}} = O(f(n) \cdot (\text{width} \times \text{height})) \quad (3.7)$$

Für das klassische Ray-Casting liefert die Algorithmische Geometrie verschiedene Lösungsansätze. Die Laufzeit dieser Lösungsansätze spiegelt sich in der Funktion f wieder und ist abhängig von der Anzahl n der Objekte. Ob die Funktion f für das klassische und das relativistische Ray-Casting gleich ist, ist noch nicht untersucht worden. Somit ist die Frage, ob relativistisches Ray-Casting ein komplexeres Problem ist als klassisches Ray-Casting, noch offen.

Obwohl noch einige Fragen zur Laufzeit offen sind, wird in Gl. (3.7) deutlich, dass dieses Verfahren, wegen der hohen Abhängigkeit von der Auflösung, sehr langsam ist.

Vor- und Nachteile dieses Verfahrens

Vorteile

- Da nur wenige Änderungen am klassischen Ray-Tracing-Algorithmus durchgeführt werden müssen, ist die Implementierung einfach.
- Da alle Berechnungen weiterhin mit 3-dimensionalen Operationen durchgeführt werden, sind auch alle Optimierungen zur Schnittpunktberechnung von klassischem Ray-Tracing einsetzbar.
- Da alle Strahlen unabhängig voneinander berechnet werden können, lässt sich Ray-Tracing leicht parallelisieren.

Nachteile

- Bei der vorgestellten Methode kann man nur eine Relativbewegung in x-Richtung simulieren.²
- Es wird nur ein Unterschied zwischen zwei Koordinatensystemen (Beobachter und Objekte) gemacht, daher können keine Objekte mit verschiedenen Geschwindigkeiten simuliert werden.²
- Die Lichtquellen müssen im Ruhesystem der Objekte sein.
- Doppler- und Suchscheinwerfer-Effekt werden bei diesem Verfahren noch nicht betrachtet.³
- Der Algorithmus ist zunächst nur auf eine konstante Geschwindigkeit ausgelegt. Geschwindigkeitsänderungen oder Beschleunigungen können noch nicht simuliert werden.

Beispiele für relativistisches Ray-Tracing

In Abbildung 3.3 sind Ergebnisse des relativistischen Renderings dargestellt. Die Abbildungen (a) und (b) sind genau nach dem oben aufgeführten Algorithmus von HSIUNG UND DUNN erstellt worden und sind auch in der selbigen Arbeit [5] aufgeführt. In Bild (a) ist die Szene in Ruhe zu sehen, während sich in (b) der Beobachter mit $\beta = 0.9$ auf die Szene zu bewegt.

Die Bilder in (a) und (b) wurden aus einem Druck nachträglich digitalisiert und weisen daher nur eine geringe Qualität auf. Die Bilder (c) und (d) stammen aus einer neueren Arbeit von WEISKOPF ([20]) und geben das relativistische Ray-Tracing in weit besserer Qualität wieder. Ausserdem wurde hier bereits der Suchscheinwerfer-Effekt mit einbezogen. Bild (c) zeigt die Szene in Ruhe, während in Bild (d) eine relative Geschwindigkeit von Beobachter und Szene von $\beta = 0.8$ dargestellt worden ist.

²Für diesen Punkt wird in [5] ein Lösungsansatz angegeben.

³Dieser Teil wird in [6] erläutert.

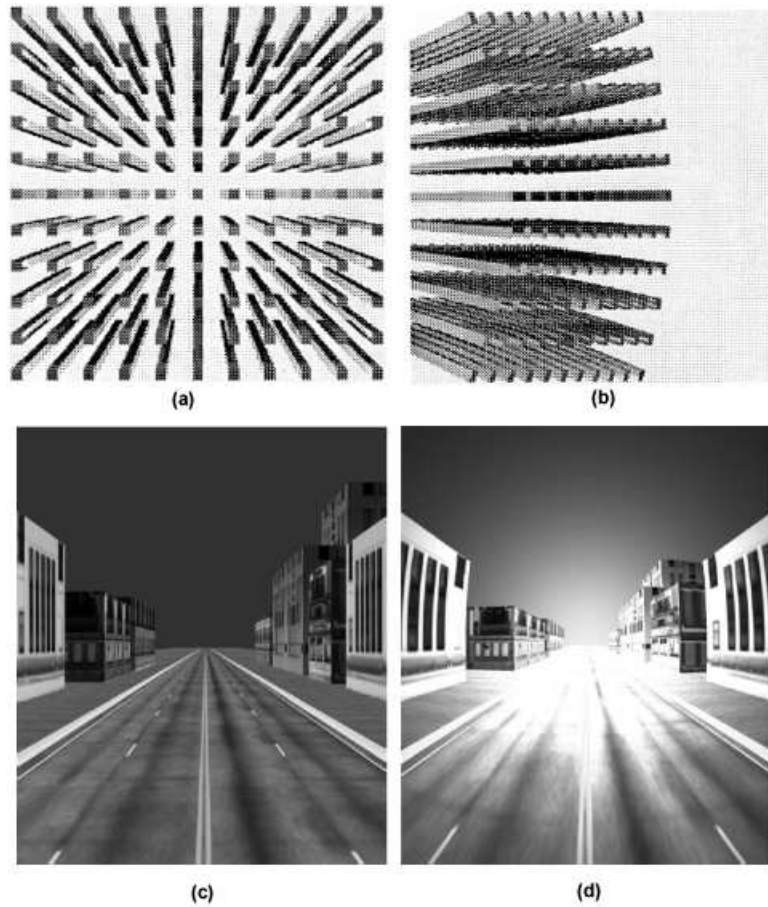


Abbildung 3.3: Beispiele für relativistisches Ray-Tracing aus [5] und [20]

3.2 Bild-basierte Verfahren

Bei den bildbasierten Verfahren werden Eingaben wie Fotos oder Videos (sogenannte Samples) verwendet, um zum Beispiel photorealistische Panoramaansichten zu erzeugen. Die bildbasierten Verfahren arbeiten grundlegend anders als das Ray-Tracing oder auch das Polygon-Rendering. Während beim Polygon-Rendering und beim Ray-Tracing 3-dimensionale Koordinaten für die Objekte vorliegen, enthalten die Samples nur 2-dimensionale Bildinformationen. Um dennoch ein Bild erzeugen zu können, dürfen die Samples nicht nur als 2-dimensionales Bild aufgefasst werden. Damit ein Pixel eines Samples eine bestimmte Farbe erhält, muss auf die entsprechende Bildposition ein Lichtstrahl mit genau dieser Farbe (im folgenden wird die Farbe auch durch die Wellenlänge λ charakterisiert) getroffen sein. Blickt man von der Aufnahmeposition des Samples in Richtung eines beliebigen Bildpunktes, so erhält man die Richtung des Lichtstrahls für diesen Bildpunkt. Die Richtung lässt sich mit zwei Winkeln Θ und Φ vollständig beschreiben (Abb. 3.4).

Somit enthält ein Sample bereits Informationen über Richtung und Wellenlänge zahlreicher (der Auflösung der Samples entsprechender) Lichtstrahlen. Nimmt man nun mehrere Samples zusammen, so kann man sogenannte Maps erzeugen. Diese Maps beschreiben meist die Lichtstrahlen um

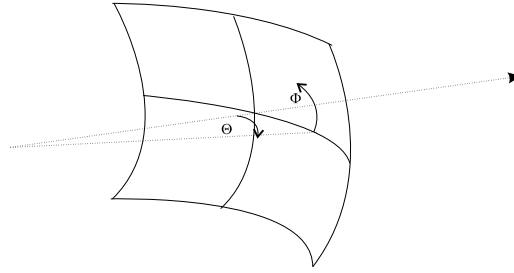


Abbildung 3.4: Blickrichtung in der Plenoptischen Funktion mit den Polarwinkeln Θ und Φ

einen bestimmten Beobachtungspunkt $((V_x, V_y, V_z)^T)^4$. Die Maps sind daher meist sphärisch oder zylindrisch um den Beobachter angeordnet (Abb. 3.5).

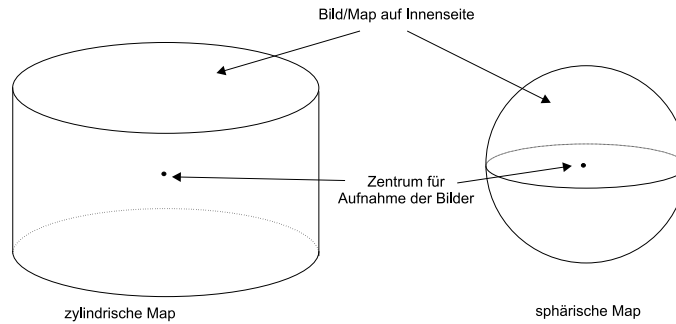


Abbildung 3.5: Beim generieren der Maps aus den Samples werden die Daten der Samples meist als Zylinder oder Kugel (Sphäre) angeordnet

Um nun für einen Beobachter an einer bestimmten Position und mit einer bestimmten Blickrichtung ein Bild zu erzeugen, wird das Bild mit Hilfe der Plenoptischen Funktion $P(\Theta, \Phi, \lambda, t, V_x, V_y, V_z)$ gerendert. P gibt für die Beobachterposition $(V_x, V_y, V_z)^T$ und die Blickrichtung (Θ, Φ) (Abb. 3.4) zum Zeitpunkt t die Intensität der Wellenlänge λ (Intensität der Farbe) für den Beobachter wieder. In [1] wird dies genauer erläutert (eine Implementierung mit zylindrischen Maps findet sich in [11]). Somit lassen sich mit relativ einfachen Mitteln photorealistische Panoramaansichten erzeugen. Doch diese Methode ist auch geeignet um zu zeigen wie ein Beobachter mit hoher Geschwindigkeit die Welt sehen würde.

3.2.1 Algorithmus von Weiskopf, Korbas und Ruder

In [21] wird eine Bild-basierte Technik für relativistische Szenen vorgestellt. Mit dieser Methode können klassische Bilder in relativistische Bilder umgewandelt werden. Ähnlich wie auch beim relativistischen Ray-Tracing in 3.1.2 wird dazu vor der klassischen Renderingstufe eines klassischen Bild-basierten Verfahrens eine relativistische Stufe eingefügt.

Der Algorithmus von WEISKOPF, KORBAS UND RUDER basiert auf einer Erweiterung der Plenoptischen Funktion P . Die Parameter Θ , Φ und λ werden nicht gleich mit Hilfe der klassischen

⁴ T steht für die transponierte Matrix.

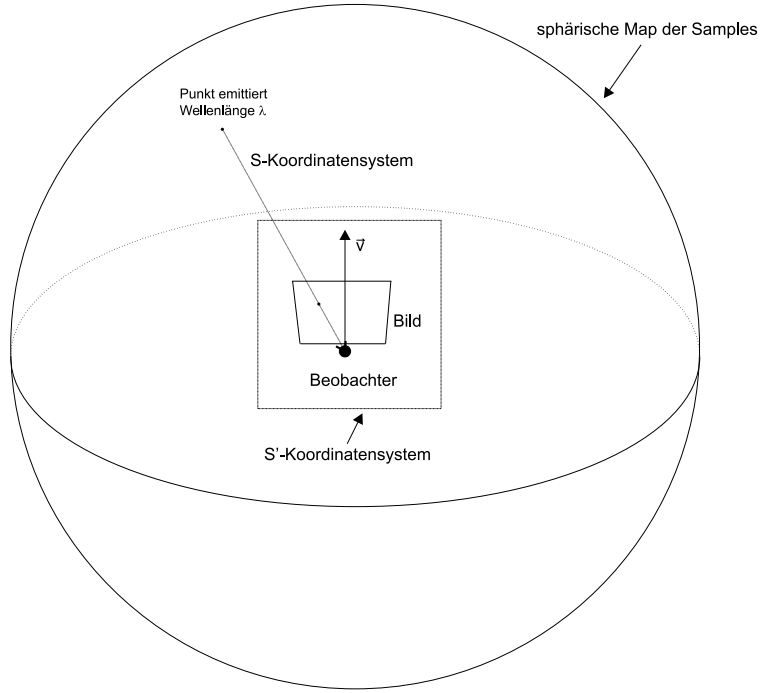


Abbildung 3.6: Prinzip des Bild-basierten Algorithmus von WEISKOPF, KORBAS UND RUDER

Funktion für P in eine Intensität für einen Bildpunkt umgesetzt, sondern zunächst in einer relativistischen Stufe umgewandelt. Die Plenoptische Funktion beschreibt nach [1] physikalisch gesehen kein Bild, sondern die Strahlung aus einer bestimmten Richtung an der Position des Beobachters. Somit entsprechen die Punkte der Map nicht einfach Bildpunkten, sondern charakterisieren die Lichtstrahlen aus der jeweiligen Richtung zum Mittelpunkt der Map (siehe auch Abb. 3.5). Für die relativistische Betrachtung ist daher wichtig, dass die Punkte des S -Koordinatensystems der Map keine Koordinatenpunkte sind, sondern Lichtstrahlen aus der entsprechenden Richtung. Somit kann man statt der Lorentz-Transformation auf die Aberrationsgleichung (vgl. Kapitel 2.5) zurückgreifen, die ausserdem bereits den Winkel Θ enthält und sich daher für diesen Fall gut einsetzen lässt.

Der Beobachter befindet sich im relativ bewegten System S' , das sich in z -Richtung mit der Geschwindigkeit \vec{v} auf das S -System zu bewegt (Abb. 3.6). Wie beim Ray-Tracing wird auch beim bildbasierten Rendern, Bildpunkt für Bildpunkt durchlaufen. Für jeden Bildpunkt werden zunächst die Winkel Θ' und Φ' des S' -Systems berechnet. Diese müssen dann ins S -System transformiert werden. Mathematisch lässt sich dies in Form zweier unterschiedlicher Plenoptischer Funktionen P und P' ausdrücken. Dabei entspricht P der klassischen Plenoptischen Funktion:

$$P'(\Theta', \Phi', \lambda', t', V'_x, V'_y, V'_z) = P(\Theta, \Phi, \lambda, t, V_x, V_y, V_z) \quad (3.8)$$

Mit Hilfe der Aberrationsgleichung (Gl. (2.4)) ergibt sich für die Winkel aus der Sicht des Beobachters (mit $\beta = \frac{v}{c}$) der Zusammenhang:

$$\Theta' = \arccos\left(\frac{\cos \Theta - \beta}{1 - \beta \cos \Theta}\right), \quad \Phi' = \Phi \quad (3.9)$$

Und umgekehrt:

$$\Theta = \arccos\left(\frac{\cos \Theta' + \beta}{1 + \beta \cos \Theta'}\right), \quad \Phi = \Phi' \quad (3.10)$$

Diese Gleichungen genügen bereits, um die geometrischen Effekte der speziellen Relativitätstheorie mit Hilfe von Gl. (3.8) darzustellen:

$$P'(\Theta', \Phi') = P\left(\arccos\left(\frac{\cos \Theta' + \beta}{1 + \beta \cos \Theta'}\right), \Phi\right) \quad (3.11)$$

Um auch den relativistischen Doppler-Effekt darzustellen, muss die Wellenlänge umgerechnet werden. Nach Gl. (2.3) in Kapitel 2.4 gilt für λ und λ' :

$$\lambda = \lambda' \frac{1 - \beta \cos \Theta}{\sqrt{1 - \beta^2}} = \lambda' \frac{\sqrt{1 - \beta^2}}{1 + \beta \cos \Theta'} \quad (3.12)$$

Der Doppler-Effekt kann also einfach durch Einfügen des neuen Parameters für λ in die Plenoptische Funktion visualisiert werden:

$$P'(\Theta', \Phi', \lambda') = P\left(\arccos\left(\frac{\cos \Theta' + \beta}{1 + \beta \cos \Theta'}\right), \Phi, \lambda' \frac{\sqrt{1 - \beta^2}}{1 + \beta \cos \Theta'}\right) \quad (3.13)$$

Für die Visualisierung des Suchscheinwerfer-Effektes muss eine weitere relativistische Renderingstufe nach der klassischen Stufe eingefügt werden. Der Suchscheinwerfer-Effekt lässt sich durch eine relativistische Betrachtung der Strahlungsgesetze des schwarzen Körpers herleiten. Nach [22] und [23] erhält man für den Zusammenhang der Strahldichte L und L' der zwei Systeme S und S' :

$$L_\lambda(\Theta, \Phi) = L'_{\lambda'}(\Theta', \Phi') \left(\frac{\sqrt{1 - \beta^2}}{1 - \beta \cos \Theta}\right)^5 = L'_{\lambda'}(\Theta', \Phi') \left(\frac{1 + \beta \cos \Theta'}{\sqrt{1 - \beta^2}}\right)^5 \quad (3.14)$$

Die Plenoptische Funktion beschreibt aber gerade die Strahldichte L . Somit gilt:

$$P(\Theta, \Phi, \lambda) = L_\lambda(\Theta, \Phi) = L'_{\lambda'}(\Theta', \Phi') \left(\frac{1 + \beta \cos \Theta'}{\sqrt{1 - \beta^2}}\right)^5 = P'(\Theta', \Phi', \lambda') \left(\frac{1 + \beta \cos \Theta'}{\sqrt{1 - \beta^2}}\right)^5 \quad (3.15)$$

Die geometrischen Effekte, der relativistische Doppler-Effekt und der Suchscheinwerfer-Effekt lassen sich mathematisch wie folgt zusammenfassen:

$$P'(\Theta', \Phi', \lambda') = \frac{P\left(\arccos\left(\frac{\cos \Theta' + \beta}{1 + \beta \cos \Theta'}\right), \Phi, \lambda' \frac{\sqrt{1 - \beta^2}}{1 + \beta \cos \Theta'}\right)}{\left(\frac{1 + \beta \cos \Theta'}{\sqrt{1 - \beta^2}}\right)^5} \quad (3.16)$$

Der Algorithmus von WEISKOPF, KORBAS UND RUDER lässt sich daher wie folgt beschreiben:

1. Für alle Bildpunkte:
 1. Berechne die Winkel Θ' und Φ' des Bildpunktes.
 2. Transformiere die Winkel Θ' und Φ' nach Gl. (3.10) zu den Winkeln Θ und Φ .
3. Für alle Wellenlängen:
 1. Berechne die Doppler-Verschiebung für λ nach Gl. (3.12).
 2. Berechne die klassische Plenoptische Funktion P mit den berechneten Parametern Θ , Φ und λ .
 3. Berechne den Bildpunkt mit Hilfe der berechneten Funktion P und dem Suchscheinwerfer-Effekt nach Gl. (3.16).

Vor- und Nachteile des Algorithmus von Weiskopf, Korbas und Ruder

Vorteile

- Die klassischen Plenoptischen Verfahren lassen sich leicht mit diesem Algorithmus erweitern.
- Da hier reale Bilder verwendet werden, kann sich der Betrachter die Auswirkungen der Relativitätstheorie sehr gut vorstellen.
- Der Rechenaufwand ist höchstens genauso gross wie beim Ray-Tracing.
- In [21] ist auch eine Erweiterung auf Filmsequenzen angegeben.

Nachteile

- Bei der vorgestellten Methode kann man nur eine Relativbewegung in z-Richtung simulieren.⁵
- Die aufgezeichnete Szene auf der Map wird als statisch vorausgesetzt.
- Die Samples lassen sich nicht immer mit gewöhnlichen Kameras erstellen. Normalerweise sind hierzu komplizierte Spezialkonstruktionen nötig.
- Der Algorithmus ist zunächst nur auf eine konstante Geschwindigkeit ausgelegt. Geschwindigkeitsänderungen oder Beschleunigungen können nicht simuliert werden.

Beispiele für relativistisches Bild-basiertes Rendering

Das Bild-basierte Rendering verspricht photorealistische Bilder von schnell bewegten Szenen. Um einen kleinen Eindruck vom Realismus dieser Methode zu bekommen, sind in Abbildung 3.7 einige mit diesem Verfahren erzeugten Bilder zu sehen.

In Bild (a) ist die Szene in Ruhe dargestellt. Bild (b) zeigt die Szene bei einer Geschwindigkeit von $0.99c$ ohne Doppler- und Suchscheinwerfer-Effekt. Schließlich ist in Bild (c) die Szene bei einer Geschwindigkeit von $0.3c$ mit Doppler- und Suchscheinwerfer-Effekt zu sehen.

Mögliche Probleme dieses Verfahrens

Aus Kapitel 2.3 ist bereits bekannt, dass schnelle Objekte teilweise gedreht erscheinen. Da die Samples jedoch nicht die kompletten 3-dimensionalen Informationen der Szene enthalten, liegt die Vermutung nahe, dass bei einigen Szenen keine korrekte Darstellung wiedergegeben werden kann, da keine Daten über die Rückseiten der aufgenommenen Objekte zur Verfügung stehen. Ein Beweis bzw. eine Widerlegung für diese Behauptung steht jedoch noch aus.

Eine mögliche Auflösung dieses Problems könnte in der Tatsache begründet werden, dass in den relativistischen Ansichten die Szene, wegen der Aberration des Lichtes, nach vorne verschoben wird. Somit besteht die Möglichkeit, dass jene Objekte, dessen Rückseite zu sehen ist, in der klassischen Ansicht hinter dem Beobachter sind. Eine sphärische Map würde somit gerade die Rückseite des Objektes enthalten. Auf diese Weise könnte die Szene dennoch korrekt dargestellt werden.

⁵In [21] wird allerdings angegeben, dass sich dieses Problem mittels Rotationen lösen lässt.



Abbildung 3.7: Beispiel für das relativistische Bild-basierte Rendering[20]

3.3 Polygon-Rendering Algorithmen

Das Polygon-Rendering hat seinen Namen aufgrund der Art, wie die 3-dimensionalen Bildinformationen verarbeitet und gespeichert werden. Jedes Objekt, das dargestellt werden soll, wird mit Hilfe eines 3-dimensionalen Polygongitters angenähert. Die Punkte der Polygone werden dann mit Hilfe einer Projektionsmatrix auf eine Bildebene projiziert. Für eine genauere Beschreibung dieser Technik sei an dieser Stelle zum Beispiel auf Kapitel 8 in [2] verwiesen.

Da beim Polygon-Rendering nicht alle Bildpunkte einzeln projiziert werden müssen, sondern nur die Objektpunkte, hat man gegenüber den oben genannten Verfahren einen Geschwindigkeitsgewinn. Dieser Geschwindigkeitsvorsprung wird noch ausgebaut durch die Hardwareunterstützung der heutigen PCs. Daher ist dieses Verfahren besonders gut für interaktive Simulationen der Relativitätstheorie geeignet.

3.3.1 T-Buffer Algorithmus von Hsiung, Thibadeau und Wu

Bereits 1990 wurde von HSIUNG, THIBADEAU UND WU der sogenannte Time-Buffer (oder kurz: T-Buffer) Algorithmus vorgestellt[7]. In Analogie zum Z-Buffer, wird hier der T-Buffer eingesetzt. Dieser speichert jedoch, statt der Tiefe (Z-Buffer), die Zeit eines Raum-Zeit-Ereignisses. Das eigentliche relativistische Rendering wird jedoch nicht mit dem T-Buffer selbst durchgeführt, sondern mit Hilfe der Lorentz-Transformation. Der T-Buffer entscheidet dann, ob ein Punkt sichtbar ist oder nicht. Es handelt sich dabei also um einen relativistischen Culling-Algorithmus, der in Analogie zum Z-Buffer arbeitet. Das sich der klassische Z-Buffer Algorithmus so leicht in den relativistischen T-Buffer Algorithmus umwandeln lässt ist bemerkenswert und keinesfalls auf einen typischen Zusammenhang zwischen relativistischer Visualisierung und klassischer Visualisierung zurückzuführen. Im Gegensatz zum klassischen Verdeckungsproblem, das von der Geometrie der Objekte abhängt, kommt beim relativistischen Verdeckungsproblem die Bewegung der Objekte hinzu und somit eine zusätzliche zeitliche Abhängigkeit.

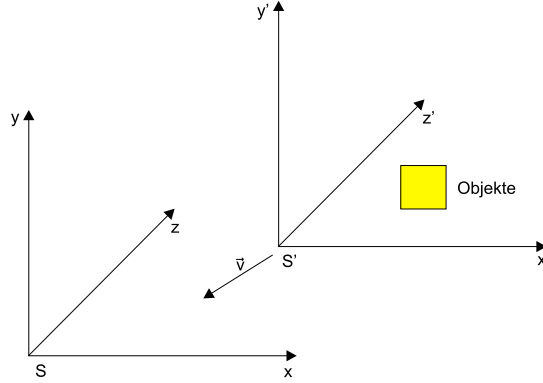


Abbildung 3.8: Prinzip des relativistischen Polygon-Rendering

Beim T-Buffer Algorithmus gehen HSIUNG, THIBADAEU UND WU zunächst von zwei Koordinatensystemen aus. Im System S ruht die Kamera, während in S' alle Objekte ruhen (Abb. 3.8). Die Systeme bewegen sich mit einer Geschwindigkeit von \vec{v} zueinander und sollen sich zum Zeitpunkt $t = t' = 0$ in ihren Ursprüngen treffen. Um ein raum-zeitliches Ereignis in S , in S' -Koordinaten umrechnen zu können, kann man die allgemeine Lorentz-Transformation aus Gl. (1.5) benutzen:

$$L = \begin{pmatrix} (\gamma - 1)n_x^2 + 1 & (\gamma - 1)n_x n_y & (\gamma - 1)n_x n_z & -\beta\gamma n_x \\ (\gamma - 1)n_x n_y & (\gamma - 1)n_y^2 + 1 & (\gamma - 1)n_y n_z & -\beta\gamma n_y \\ (\gamma - 1)n_x n_z & (\gamma - 1)n_y n_z & (\gamma - 1)n_z^2 + 1 & -\beta\gamma n_z \\ -\beta\gamma n_x & -\beta\gamma n_y & -\beta\gamma n_z & \gamma \end{pmatrix} \quad (3.17)$$

Der Vektor \vec{n} bezeichnet dabei die Richtung der Geschwindigkeit \vec{v} und ist gleich dem normierten Vektor von \vec{v} . β erhält man als Verhältnis aus dem Betrag von \vec{v} und der Lichtgeschwindigkeit c ($\beta = \frac{|\vec{v}|}{c}$).

Um vom S' -System Koordinaten ins S -System umzurechnen, kann die gleiche Transformation verwendet werden. Es muss nur die Richtung der Geschwindigkeit \vec{v} der Systeme zueinander umgekehrt werden. Dies lässt sich zum Beispiel durch negieren von β bewerkstelligen:

$$L^{-1} = \begin{pmatrix} (\gamma - 1)n_x^2 + 1 & (\gamma - 1)n_x n_y & (\gamma - 1)n_x n_z & \beta\gamma n_x \\ (\gamma - 1)n_x n_y & (\gamma - 1)n_y^2 + 1 & (\gamma - 1)n_y n_z & \beta\gamma n_y \\ (\gamma - 1)n_x n_z & (\gamma - 1)n_y n_z & (\gamma - 1)n_z^2 + 1 & \beta\gamma n_z \\ \beta\gamma n_x & \beta\gamma n_y & \beta\gamma n_z & \gamma \end{pmatrix} \quad (3.18)$$

Um das relativistische Polygon-Rendering zu verstehen, muss man sich zunächst den Zusammenhang zwischen Objekt, Licht, Kamera und Bild klarmachen. Damit eine Kamera ein Bild von einem Objekt erhält, muss Licht von dem Objekt empfangen werden. Vom aussenden des Lichtes bis zum Eintreffen bei der Kamera vergeht eine bestimmte Zeit Δt . In dieser Zeit legt das Licht die Strecke $c\Delta t$ zurück. Diese Strecke muss genau der Entfernung zwischen Objekt und Kamera zum Zeitpunkt des Aussendens des Lichtes entsprechen. In einem Minkowski-Diagramm (siehe auch Kapitel 1.7) entspricht dies der Berechnung eines raum-zeitlichen Ereignisses als Schnittpunkt der Weltlinie des Objektes mit der Kamera (Abb. 3.9). Diese Bedingung soll nun im folgenden mathematisch ausgedrückt werden.

Wird zum Zeitpunkt t_{Kamera} im S -System an der Position $P_{\text{Kamera}} = (x_{\text{Kamera}} \ y_{\text{Kamera}} \ z_{\text{Kamera}})^T$

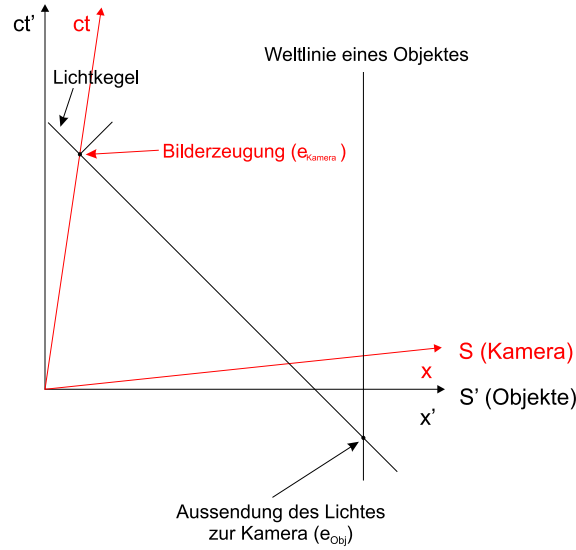


Abbildung 3.9: Minkowski-Diagramm mit Ruhesystem der Kamera und dem dazu relativ bewegten Ruhesystem der Objekte

ein Bild aufgenommen, so entspricht das einem Ereignis e_{Kamera} im S -System:

$$e_{\text{Kamera}} = \begin{pmatrix} x_{\text{Kamera}} \\ y_{\text{Kamera}} \\ z_{\text{Kamera}} \\ t_{\text{Kamera}} \end{pmatrix} \quad (3.19)$$

e_{Kamera} lässt sich mit Hilfe der Lorentz-Transformation aus Gl. (3.17) ins S' -System umrechnen:

$$e'_{\text{Kamera}} = \begin{pmatrix} x'_{\text{Kamera}} \\ y'_{\text{Kamera}} \\ z'_{\text{Kamera}} \\ t'_{\text{Kamera}} \end{pmatrix} = L e_{\text{Kamera}} \quad (3.20)$$

Um nun feststellen zu können von welchen Objekten Licht bei der Kamera eintrifft, muss zunächst ermittelt werden, wann welches Objekt und von wo das Objekt Licht aussendet. In S' sind alle Objekte in Ruhe. Somit ist ein beliebiger Objektpunkt $P'_{\text{Obj}} = (x'_{\text{Obj}} \ y'_{\text{Obj}} \ z'_{\text{Obj}})^T$ zu jedem Zeitpunkt t'_{Obj} in Ruhe. Dies entspricht einem Ereignis $e'_{\text{Obj}}(t'_{\text{Obj}}) = (x'_{\text{Obj}} \ y'_{\text{Obj}} \ z'_{\text{Obj}} \ t'_{\text{Obj}})^T$. Gesucht ist das Ereignis e'_{Obj} , dessen Licht zum Zeitpunkt t'_{Kamera} bei P_{Kamera} eintrifft:

$$\underbrace{\sqrt{(x'_{\text{Obj}} - x'_{\text{Kamera}})^2 + (y'_{\text{Obj}} - y'_{\text{Kamera}})^2 + (z'_{\text{Obj}} - z'_{\text{Kamera}})^2}}_{\text{Entfernung zwischen Punkt und Kamera}} = \underbrace{|c(t'_{\text{Obj}} - t'_{\text{Kamera}})|}_{\text{Zeit des Lichtes}} \quad (3.21)$$

Durch Umformungen erhält man t'_{Obj} :

$$t'_{\text{Obj}} = t'_{\text{Kamera}} - \frac{1}{c} \sqrt{(x'_{\text{Obj}} - x'_{\text{Kamera}})^2 + (y'_{\text{Obj}} - y'_{\text{Kamera}})^2 + (z'_{\text{Obj}} - z'_{\text{Kamera}})^2} \quad (3.22)$$

Durch Gl. (3.22) ist das Ereignis e'_{Obj} nun bekannt. Um den Punkt rendern zu können, muss das Ereignis jedoch zunächst in das Kamerasystem S transformiert werden. Dies lässt sich mit der inversen Lorentz-Transformation aus Gl. (3.18) bewerkstelligen:

$$e_{\text{Obj}} = \begin{pmatrix} x_{\text{Obj}} \\ y_{\text{Obj}} \\ z_{\text{Obj}} \\ t_{\text{Obj}} \end{pmatrix} = L^{-1}e'_{\text{Obj}} \quad (3.23)$$

Der Ortspunkt P_{Obj} des Ereignisses e_{Obj} kann nun im Grunde mit einem klassischen Polygon-Rendering Algorithmus gerendert werden. Es kann nun aber vorkommen, dass ein anderes Ereignis $e_{\text{Obj}2}$ mit dem Punkt $P_{\text{Obj}2}$ auf den gleichen Bildpunkt projiziert wird. Liegt das Ereignis $e_{\text{Obj}2}$ weiter in der Vergangenheit als das Ereignis e_{Obj} , so würde der Punkt $P_{\text{Obj}2}$ in der Realität nicht zu sehen sein.

An dieser Stelle ist also ein relativistisches Culling nötig, mit dem entschieden werden kann, ob ein Punkt sichtbar ist oder nicht. Mit dem T-Buffer lässt sich dieses Problem lösen. Im T-Buffer werden für jeden Pixel die Ereigniszeiten (t_{Obj}) gespeichert. Die Ersetzungsregel für den T-Buffer ist genau umgekehrt zum Z-Buffer: Kleinere Werte für die Zeit werden durch grössere ersetzt. Somit hat immer das Ereignis vorang, das weiter in der Zukunft liegt (grösseres t).

Zusammenfassend lässt sich der T-Buffer Algorithmus wie folgt beschreiben:

1. Initialisierung:

- Initialisiere klassisches Rendering (Framebuffer auf Hintergrundfarbe, Z-Buffer initialisieren, usw.)
- T-Buffer initialisieren: Setze Buffer auf kleinst möglichen Wert (-MAXWERT).
- Berechne $e'_{\text{Kamera}} = Le_{\text{Kamera}}$.

2. T-Buffer:

- Durchlaufe die Ecken aller Dreiecke (*Vertices*):
 1. Berechne t'_{Obj} nach Gl. (3.22)
 2. Transformiere Ereignis e'_{Obj} nach e_{Obj} mit Gl. (3.23).
 3. Projiziere Ortspunkt von e_{Obj} mit Hilfe einer klassischen Projektion auf einen Bildpunkt B_{2D} .
 4. Wenn t_{Obj} grösser ist als der vorhandene Wert im T-Buffer für B_{2D} , dann setze B_{2D} auf Farbe von Objekt und setze Wert im T-Buffer für B_{2D} auf t_{Obj} .

Laufzeit des relativistischen Polygon-Rendering

Die Laufzeit des Polygon-Rendering hängt besonders von der Anzahl der transformierten Punkte ab, und somit von der Anzahl n der Ecken aller transformierten Dreiecke. Es ergibt sich daher für das klassische wie auch für das relativistische Polygon-Rendering folgende Laufzeit:

$$\text{Poly}_{\text{Laufzeit}} = O(n) \quad (3.24)$$

Streng genommen wird für den T-Buffer eine Laufzeit in Abhängigkeit von der Auflösung benötigt. Da der T-Buffer jedoch in Analogie zum Z-Buffer arbeitet, lässt er sich mit heutiger Hardware

derart beschleunigen, dass dieser Punkt für die Laufzeitbetrachtung unerheblich ist.

Im Gegensatz zum Ray-Tracing, in dem die Auflösung eine erhebliche Rolle für die Laufzeit spielt, ist das Polygon-Rendering unabhängig von der Laufzeit. Dies ist vor allem der Grund dafür, warum dieses Verfahren so wichtig für die Echtzeitdarstellung relativistischer Szenen ist.

Vor- und Nachteile des T-Buffer Algorithmus von Hsiung, Thibadeau und Wu

Vorteile

- Beim Polygon-Rendering müssen nur einzelne Punkte der Objekte projiziert werden. Daher ist relativistisches Polygon-Rendering sehr schnell.
- Da das klassische Polygon-Rendering heute von den meisten PCs durch Grafikkarten hardwareunterstützt wird, ist sogar eine Echtzeitnavigation durch relativistische Szenen möglich.[17]
- Nach [7] lässt sich dieser Algorithmus, durch Abwandlung eines Z-Buffer Schaltkreises, leicht in Hardware realisieren.
- Die klassische Rendering-Pipeline kann beibehalten werden und muss nur erweitert werden.
- Es wurde bereits die allgemeine Lorentz-Transformation verwendet. Somit sind alle Relativbewegungsrichtungen möglich.

Nachteile

- Da beim Polygon-Rendering nur einzelne Punkte der Objekte betrachtet werden und nicht das gesamte Objekt, kann man das wahre Aussehen schnell bewegter Objekte nur annähern.
- Alle Objekte müssen die gleiche Relativgeschwindigkeit zum Beobachter haben.⁶
- Beleuchtungseffekte lassen sich nicht relativistisch simulieren, da die Schattierungsalgorithmen auf das klassische Rendering zugeschnitten sind.
- Doppler- und Suchscheinwerfer-Effekt lassen sich nicht einfach in diesen Algorithmus integrieren. Sie müssen mit Hilfe von Texturen auf die Objekte gebracht werden.
- Der Algorithmus ist zunächst nur auf eine konstante Geschwindigkeit ausgelegt. Geschwindigkeitsänderungen oder Beschleunigungen können noch nicht simuliert werden.

Beispiel für relativistisches Polygon-Rendering

In Abbildung 3.10 ist ein Beispiel für relativistisches Polygon-Rendering aufgeführt. Die Bilder wurden mit der von DANIEL WEISKOPF entwickelten Software *Virtual Relativity* erstellt [17]. In Bild (a) ist die Szene fast in Ruhe ($\beta = 0.099$). In Bild (b) ist die Szene mit einer Geschwindigkeit von $\beta = 0.95$ gerendert worden.

⁶Für Objekte mit verschiedenen Geschwindigkeiten ist in Abschnitt 4.3 von [7] eine Erweiterung angegeben.

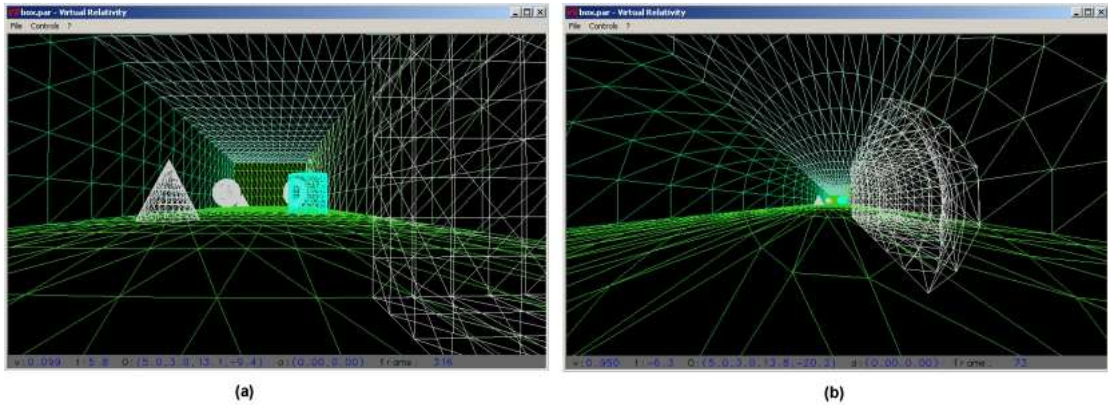


Abbildung 3.10: Beispiele für relativistisches Polygon-Rending mit Hilfe des Programmes Virtual Relativity von DANIEL WEISKOPF[17]

3.3.2 Erweiterung für eine gekrümmte Beobachter-Weltlinie von Rau, Weiskopf und Ruder

Der T-Buffer Algorithmus in der obigen Form, ist nur in der Lage gerade Beobachter-Weltlinien zu visualisieren. Bei interaktiven Simulationen der Relativitätstheorie ist es jedoch wünschenswert, wenn man auch Szenen mit gekrümmter Beobachter-Weltlinien darstellen kann. Daher erweitern RAU, WEISKOPF UND RUDER[13][20] den T-Buffer Algorithmus. Aus der Sicht des Ruhesystems S' der Objekte stellt sich die Situation wie in Abbildung 3.11 dar. Die Weltlinie des Objektes ist gerade, während die Weltlinie des Beobachters auch gekrümmte Abschnitte enthalten kann.

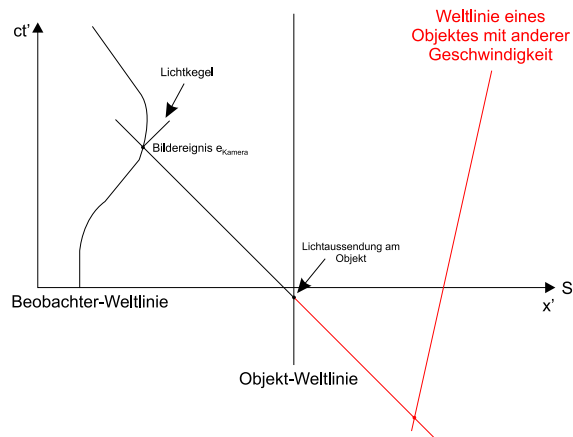


Abbildung 3.11: Minkowski-Diagramm für einen Beobachter mit gekrümmter Weltlinie

Das Prinzip des T-Buffer Algorithmus basiert nun darauf, dass der Lichtstrahl von einem Bilderzeugungereignis e_{Kamera} auf der Beobachter-Weltlinie zum Lichtaussendungsereignis e_{Obj} auf der Objekt-Weltlinie zurückverfolgt wird. Ist also das Bilderzeugungereignis e_{Kamera} und das damit zusammenhängende Bezugssystem S bekannt, so lässt sich mit Hilfe des T-Buffer Algorithmus auch eine gekrümmte Weltlinie visualisieren.

3.4 Sonstige Visualisierungsalgorithmen

3.4.1 Textur-basierter Algorithmus von Daniel Weiskopf

Beim Textur-Basierten Algorithmus von DANIEL WEISKOPF[18] handelt es sich um eine Mischung aus Polygon-Rendering und Bild-basiertem Rendering. Zunächst wird die Szene mit Hilfe des klassischen Polygon-Rendering auf eine sphärische Map gerendert. Ähnlich wie beim Bild-basierten Verfahren aus Kapitel 3.2.1 kann man nun mit der Aberrationsgleichung ein relativistisches Bild erzeugen. Dazu werden die Texturkoordinaten der sphärischen Map ähnlich wie beim Bild-basierten Verfahren manipuliert. Die so manipulierte Map wird als Textur über eine Kugel gelegt. Mit dem Beobachter im Zentrum der Kugel wird nun ein Bild mit einem klassischen Polygon-Rendering Algorithmus erzeugt.

Wegen der hohen Geschwindigkeit des Polygon-Rendering und der Hardwareunterstützung für Texturen, verspricht dieses Verfahren ebenfalls eine hohe Geschwindigkeit und somit die Visualisierung in Echtzeit. Da beim Polygon-Rendering nur einzelne Punkte der Objekte relativistisch behandelt werden und nicht alle Punkte der Objekte, kann es zu Artefakten kommen. Dieses Problem kann beim Texture-Basierten Algorithmus nicht auftreten, da das relativistische Rendering durch Texturmanipulation durchgeführt wird und das Problem so umgangen wird. Dies ist auch die ursprüngliche Motivation für dieses Verfahren gewesen.

3.4.2 Radiosity

Mit Hilfe von Radiosity Verfahren lassen sich besonders gut diffuse Beleuchtungseffekte generieren. In [20] ist angedeutet, wie sich die klassischen Radiosity Verfahren auf ein relativistisches Verfahren erweitern lassen. Es basiert darauf, dass die Bestrahlstärke B abhängig ist von der Strahlungsleistung ϕ . Ähnlich wie beim Suchscheinwerfer-Effekt kann man nun die Strahlungsgrößen relativistisch betrachten. In dieses Verfahren lassen sich daher auch gut der Doppler- und der Suchscheinwerfer-Effekt integrieren.

Kapitel 4

Ausblicke

Die im vorangegangenen Kapitel dargestellten Visualisierungsalgorithmen stellen sehr grundlegende Möglichkeiten zur Visualisierung relativistischer Szenen zur Verfügung. Um jedoch einen vollständigen Eindruck von der speziellen Relativitätstheorie zu erhalten, reichen diese Algorithmen bei weitem nicht aus. Dieses Kapitel soll einen kleinen Überblick über noch offene Fragen im Bereich der Visualisierung und auch Simulation der speziellen Relativitätstheorie geben. Obwohl hier bereits erste Lösungsideen angedeutet werden, sind noch sehr viele Fragen für die theoretische wie auch für die praktische Informatik offen.

4.1 Gekrümmte Beobachter- und Objekt-Weltlinien

Im erweiterten T-Buffer Algorithmus von RAU, WEISKOPF UND RUDER aus Kapitel 3.3.2 ist es bereits möglich einen Beobachter mit gekrümmter Weltlinie zu visualisieren. Um aber einige Aspekte der Relativitätstheorie besser zu verstehen ist mehr als ein Beobachter nötig. Dies lässt sich in der Praxis beispielsweise durch vernetzte Rechner realisieren. Will man jedoch zulassen, dass alle Beobachter sich individuell bewegen können, so erhält man verschiedene Beobachter mit verschiedenen, gekrümmten Weltlinien. Die anderen Beobachter erscheinen wie Objekte mit einer gekrümmten Weltlinie in der Simulation eines bestimmten Beobachters.

Betrachtet man die zu simulierende Situation zunächst für einen Beobachter und ein Objekt aus der Sicht eines insgesamt ruhenden Betrachters, so erhält man das Minkowski-Diagramm in Abbildung 4.1. Die Visualisierung einer gekrümmten Weltlinie erscheint nach diesem Minkowski-Diagramm zunächst als einfaches Problem: Nachdem man das Ereignis e_{Kamera} wie im Algorithmus von RAU, WEISKOPF UND RUDER bestimmt hat, kann man einen Schnittpunkt mit der gekrümmten Weltlinie des Objektes bestimmen und dann mit Hilfe des ursprünglichen T-Buffer Algorithmus von HSIUNG, THIBADEAU UND WU das Objekt visualisieren. Die Bestimmung des Schnittpunktes mit der gekrümmten Weltlinie ist jedoch nicht so einfach.

Um dieses Problem zu lösen, muss man zunächst die gekrümmte Weltlinie in geeigneter Form darstellen. Um sowohl eine konstante Geschwindigkeit (gerade Weltlinie), als auch eine Beschleunigung (gekrümmte Weltlinie) darzustellen, eignen sich am besten Strecken und Abschnitte, die durch quadratische Splines beschrieben werden. Dabei ist wichtig, dass hier nicht beliebige Strecken und Splines zugelassen sind, sondern nur solche, die Geschwindigkeiten kleiner gleich der Lichtgeschwindigkeit beschreiben.

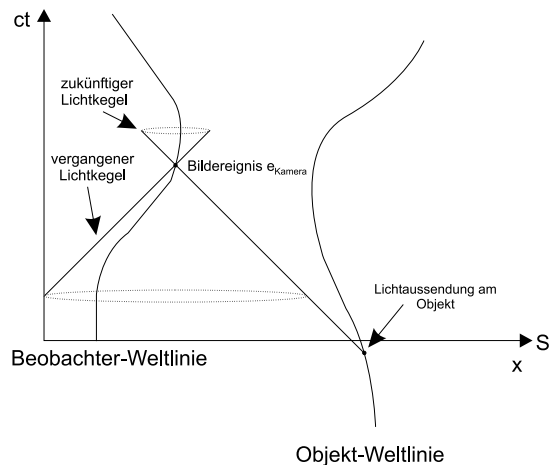


Abbildung 4.1: Minkowski-Diagramm für einen Beobachter und ein Objekt mit gekrümmter Weltlinie

Um den richtigen Schnittpunkt zu finden muss zunächst der Abschnitt gefunden werden, auf dem sich dieser Schnittpunkt befindet. Die einfachste Möglichkeit hierfür ist eine naive Suche über alle Strecken und quadratischen Splines. Nimmt man an das es k -Abschnitte (Strecken und Splines) gibt, so kann man von einer Suchzeit der Größenordnung $O(k)$ ausgehen.

Wegen der oben gemachten Einschränkung der Strecken und Splines durch die Lichtgeschwindigkeit, gibt es nur einen Schnittpunkt mit dem Lichtkegel. Daher gibt es die Vermutung, dass man statt der naiven Suche eine binäre Suche durchführen kann. Dadurch würde sich die Laufzeit für die Suche auf $O(\log(k))$ verbessern. Die genaue Untersuchung dieses Sachverhaltes steht noch aus.

4.2 Simulation nicht visueller Eindrücke

Der Schwerpunkt dieser Arbeit liegt auf der Visualisierung. Doch für die Simulation der Relativitätstheorie kann man einige wichtige Aspekte nicht ausser acht lassen. Im folgenden sollen daher einige dieser Aspekte und die damit verbundenen Probleme für die Informatik näher erläutert werden.

4.2.1 „Gefühlte“ Zeit/Relativität der Gleichzeitigkeit

Für die Simulation der speziellen Relativitätstheorie mit mehr als einem Beobachter ergibt sich sehr schnell ein Zeitdarstellungsproblem. Während bei klassischen Mehrbenutzersimulationen für alle Beobachter die Zeit gleich schnell verläuft, ist dies bei relativistischen Simulationen nur der Ausnahmefall. Wie in Kapitel 1.7.1 bereits beschrieben ist, müssen zwei Ereignisse, die für einen Beobachter gleichzeitig stattfindet, für einen zweiten Beobachter nicht unbedingt gleichzeitig stattfinden. Somit muss man für verschiedene Beobachter verschiedene Zeiten simulieren. Es stellt sich an dieser Stelle die Frage, in welcher Form man die Zeit für den menschlichen Beobachter mit dem Computer darstellen sollte, so dass der Beobachter ein möglichst gutes Gefühl für den relativistischen Zeitablauf erhält.

Simulation durch beschleunigte Uhren

Die einfachste Möglichkeit ist die grafische Darstellung einer Uhr in der Simulationsumgebung. Entsprechend dem relativistischen Zeitablauf wird die Geschwindigkeit dieser Uhr angepasst.

Praktisch lässt sich ein solches System recht einfach realisieren: Man bezieht die Uhren aller Beobachter auf ein insgesamt ruhendes Laborsystem in dem sich die Beobachter bewegen. Dem Laborsystem könnte unsere normal Zeit zu Grunde liegen. Die Uhren, der relativ zum Laborsystem bewegten Beobachter, könnten dann entsprechend der Zeitdilatationsgleichung (vgl. Kapitel 1.5) synchronisiert werden.

Die grösste Schwäche der Simulation durch beschleunigte Uhren ist, dass der Beobachter nur einen visuellen Eindruck vom relativistischen Begriff der Zeit erhält. Um einen besseren Eindruck zu erhalten wäre es wünschenswert, wenn der Beobachter die Zeitabläufe wirklich spüren und miterleben könnte.

Simulation durch verzögerte Darstellung

Damit ein menschlicher Beobachter die relativistischen Zeitabläufe wirklich erfahren kann, kann man die Vorgänge mit Computern in der Realzeit des Benutzers ablaufen lassen. Dazu muss man, wie auch bei der Simulation durch beschleunigte Uhren, zunächst ein insgesamt ruhendes Laborbezugssystem definieren. Die Geschwindigkeit der Simulation der einzelnen Beobachter wird dann mit Hilfe der Gleichung für die Zeitdilatation aus Kapitel 1.5 angepasst. Somit werden vom Computer zunächst alle Bewegungsabläufe der Beobachter und Objekte gespeichert und dann verzögert dargestellt.

Bei dieser Methode tritt jedoch schnell ein Problem auf, sobald zwei Beobachter mit unterschiedlichen Geschwindigkeit in einem interaktiven System unterwegs sind. Betrachten wir dazu zwei Beobachter A und B mit den Geschwindigkeiten $0.2c$ und $0.5c$. Nach der Zeitdilatation (Gl. (1.7)) muss die Zeit für den schnelleren Beobachter B langsamer ablaufen als für den langsameren Beobachter A. Somit muss die Simulation für B schneller laufen als die für Beobachter A. Nehmen wir nun an, Beobachter A möchte nach 2 Sekunden seiner Simulation seine Bewegungsrichtung oder Geschwindigkeit ändern, dann muss dies auch für die Simulation von Beobachter B bekannt gegeben werden. Da B aber nach 2 Sekunden, aufgrund der schneller ablaufenden Simulation, bereits viel weiter in der Simulation ist und daher bereits die Stelle an der die Änderungen der Geschwindigkeit oder Bewegungsrichtung für Beobachter B hätte vorgenommen werden sollen, simuliert worden ist, können die Änderungen von Beobachter A für B nicht mehr korrekt visualisiert werden.

Um dieses Problem zu umgehen, könnte man die Auswirkungen der Änderungen verspätet für Beobachter A simulieren, so dass sie für B noch zum richtigen Zeitpunkt visualisiert werden können. Für Beobachter A wäre die Benutzung eines solchen Systems jedoch sehr verwirrend, da die Reaktionen auf seine Kommandos erst sehr spät simuliert werden.

4.2.2 Konzepte zur „interaktive“ Steuerung

Um eine realitätsnahe Simulation zu erhalten ist es besonders wichtig, eine möglichst intuitive und realistische Steuerung einzuführen, die speziell auf die Anforderungen relativistischer Szenen zugeschnitten ist. Es stellt sich an dieser Stelle zunächst die Frage, warum man nicht einfach eine klassische Steuerung eines Autos oder eines Flugzeuges als Vorbild verwenden kann. Dies hat vor allem damit zu tun, dass die Steuerungen von Auto und Flugzeug auf weit niedrigere Geschwindigkeiten zugeschnitten sind. Um beispielsweise eine Richtungsänderung durchzuführen, genügt es bei

hohen Geschwindigkeiten nicht einfach den Richtungsvektor des Beobachters durch die Steuerung zu beeinflussen, es muss statt dessen eine Beschleunigung in die entsprechende Richtung durchgeführt werden. Somit ist das Ansprechverhalten einer relativistischen Steuerung weitaus träger, als die Steuerung für ein Auto. Am nächsten kommt daher ein Steuerungsprinzip, dass man bei einer Rakete verwendet.

Ähnlich wie es den meisten Computerbenutzern aus zahlreichen Spielen bekannt ist, wird die Vorwärtsbewegung jeweils in eine Navigationsrichtung durchgeführt. Dabei wird keine plötzliche Geschwindigkeitsänderung durchgeführt, sondern eine Beschleunigung in die entsprechende Navigationsrichtung. Zusätzlich sollte man eine Rückwärtsbeschleunigung erlauben, die genau in die Gegenrichtung beschleunigt und somit den Beobachter bremsen kann.

Für Richtungsänderungen müssen besondere Anforderungen an die Steuerung gestellt werden. Während es bei klassischen Systemen (z.B. Autos, Flugzeuge, ...) kein Problem ist eine plötzliche Richtungsänderung durchzuführen, ist dies bei relativistischen Simulationen nicht ganz so trivial. Will man realitätsnah bleiben, so muss man die Richtungsänderung durch eine Beschleunigung in die jeweilige Richtung beschreiben. Dies führt jedoch dazu, dass die Steuerung sehr träge und langsam reagiert, was ein Nachteil für die Benutzung der Steuerung bedeutet.

Ein weiteres Steuerungselement könnte, neben der Navigationsrichtung, eine Blickrichtungssteuerung sein. Dies hätte vor allem bei der Visualisierung mit Head-Mounted-Displays Vorteile für den Benutzer. Bei konstanten Geschwindigkeiten lässt sich diese Steuerung auch sehr leicht in das System integrieren, da hier praktisch die gleichen Prinzipien gelten wie bei der Steuerung der Blickrichtung in klassischen 3-dimensionalen Simulationen. Dies gilt jedoch nur im Falle eines unbeschleunigten Beobachters. Sobald der Beobachter beschleunigt wird treten Kräfte auf, die die Änderung der Blickrichtung erschweren. Dies ließe sich beispielsweise mit einer aus Computerspielen bekannten Variante des sogenannten Force-Feedback realisieren.

Um dem Benutzer der Steuerung eine Rückmeldung zu geben, ist es besonders wichtig eine Anzeige für die Geschwindigkeit einzufügen. Während bei klassischen Simulationen meist ein fester Punkt (z.B. die Erde) als Bezugspunkt zur Messung der Geschwindigkeit vorhanden ist, so liegt es gerade in der Natur der Relativitätstheorie, dass es einen solchen Punkt nicht gibt. Um dennoch eine Geschwindigkeit messen zu können, kann man sich einen solchen Bezugspunkt schaffen, indem man die Simulation in einem insgesamt ruhenden Laborsystem stattfinden lässt. Die Geschwindigkeiten aller Objekte lassen sich nun auf dieses Laborsystem beziehen. Einige weitere Möglichkeiten wären: Bezugnahme auf andere Objekte oder Beobachter, oder sogar die Geschwindigkeit auf ein Objekt (z.B. Beobachter selbst) in der Vergangenheit zu beziehen (z.B. Geschwindigkeit vor einer Beschleunigung und danach).

Obwohl die Steuerung bereits in einigen Arbeiten von DANIEL WEISKOPF([20][13]) Beachtung gefunden hat, bleibt es dennoch eine Herausforderung, eine gute intuitive und realistische Steuerung zu entwickeln.

4.2.3 Relativistische Masse/Relativistische Massenträgheit

Ein besonderes Problem, im Hinblick auf die Erfahrbarkeit, erhält man bei der in Kapitel 1.6 erwähnten relativistischen Masse. Eine Masse lässt sich nicht direkt visualisieren. Um sie dennoch deutlich zu machen, könnte man mit Hilfe von Beschleunigungskräften, die auf den Benutzer wirken, den Eindruck einer Gewichtszunahme des Benutzers simulieren.

Literaturverzeichnis

- [1] ADELSON, E.H. und BERGEN, J.R.: "Computational Models of Visual Processing". *MIT Press*, (1991), Seiten 3–20
- [2] DOMIK, PROF. DR. GITTA: *Skriptum zur Vorlesung Computergrafik: Computergraphik*. (1994–1999). Universität Paderborn
- [3] EHLERS, JÜRGEN: "Aktuelle Probleme der Gravitationsphysik". *Physik Journal 1*, (2002), Nr. 7/8: Seiten 43–47
- [4] HÖFLING, OSKAR: *Physik*, Seiten 685–720. Dümmler, (1980)
- [5] HSIUNG, P.-K. und DUNN, R.H.P.: "Visualizing relativistic effects in spacetime". *In Proceedings of Supercomputing '89 Conference*, (1989), Seiten 597–606
- [6] HSIUNG, P.-K. und DUNN, R.H.P. und THIBADEAU, R.H. und COX, C.B. und WU, M. und OLBRICH, P.A.: "Wide-band relativistic Doppler-Effect visualisation". *In Proceedings of IEEE Visualization '90 Conference*, (1990), Seiten 83–92
- [7] HSIUNG, P.-K. und THIBADEAU, R.H. und WU, M.: "T-Buffer: Fast Visualisation of Relativistic Effects in Spacetime". *Computer Graphics 1990*, (1990), 24(2): 83–88
- [8] KOSTER, ROLAND: "Minkowski Spacetime", (2002).
<http://home.wanadoo.nl/ronald.koster/minkowski.pdf>
- [9] KRAUS, UTE und RUDER, HANNS und WEISKOPF, DANIEL und ZAHN, CORVIN: "Was Einstein noch nicht sehen konnte". *Physik Journal 1*, (2002), Nr. 7/8: Seiten 77–82
- [10] LORENZEN, DIRK H.: "Wenn sich die Bilder biegen - was Gravitationslinsen über das All verraten", (2002).
<http://www.weltderphysik.de/themen/universum/astro/kosmologie/gravitationslinsen/>
- [11] MCMILLAN, LEONARD und BISHOP, GARY: "Plenoptic Modeling: An Image-Based Rendering System". *In SIGGRAPH 95 Conference Proceedings*, (1995), Seiten 39–46
- [12] MÜLLER, KATHARINA: "Gravitationslinsen", (2002).
<http://www.physik.unizh.ch/~kmueller/text/vorlesung/dark/node10.html>
- [13] RAU, RENÉ T. und RUDER, HANNS und WEISKOPF, DANIEL: "Special Relativity in Virtual Reality". *Mathematical Visualization*, (1998), Seiten 269–279
- [14] RUDER, HANNS und RUDER, MARGRET: *Die spezielle Relativitätstheorie*. Vieweg, (1993)

- [15] SCHRÖTER, JOACHIM: "Über die Bilder bewegter Objekte und die Unsichtbarkeit der Lorentz-Kontraktion". *Zeitschrift für Naturforschung*, (1966), Seiten 669–679
- [16] VITZETHUM, MARKUS: "Relativitätstheorie subjektiv", (2001).
<http://www.tp1.physik.uni-erlangen.de/~vitzethum/science/facharbeit/facharbeit.html>
- [17] WEISKOPF, DANIEL: "Relativistischer Renderer: Virtual Relativity".
<http://www.vis.uni-stuttgart.de/~weiskopf/relativity/software.html>
- [18] WEISKOPF, DANIEL: "A texture mapping approach for the visualization of special relativity".
In IEEE Visualization '99 Late Breaking Hot Topics Proceedings, (1999), Seiten 41–44
- [19] WEISKOPF, DANIEL: "An immersive virtual environment for special relativity". *In Proceedings WSCG 2000 Conference*, (2000), Seiten 337–344
- [20] WEISKOPF, DANIEL: *Visualisation of Four-Dimensional spacetime*. Doktorarbeit, Eberhard-Karls-Universität zu Tübingen, (2001)
- [21] WEISKOPF, DANIEL und KOBRAS, DANIEL und RUDER, HANNS: "An Image-Based Approach to Special Relativistic Rendering". (2000), Report Nr. 145. Eberhard-Karls-Universität zu Tübingen
- [22] WEISKOPF, DANIEL und KRAUS, UTE und RUDER, HANNS: "Searchlight and Doppler effects in the visualization of special relativity: a corrected derivation of the transformation of radiance". *ACM Transactions on Graphics*, (1999), 18(3): 278–292
- [23] WEISKOPF, DANIEL und KRAUS, UTE und RUDER, HANNS: "Illumination and acceleration in the visualization of special relativity: a comment on fast rendering of relativistic objects". *The Journal of Visualisation and Computer Animation*, (2000), Seiten 185–195

Versicherung

Ich versichere, dass ich diese Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Paderborn, 6. Februar 2004

Oliver Sudmann

Danksagung

An dieser Stelle möchte ich mich zunächst bei allen bedanken, die mich bei dieser Studienarbeit unterstützt haben. Mein besonderer Dank gilt Dr. Martin Ziegler für das Thema und die Betreuung; ausserdem Privdoz. Dr. Udo Schelb für die wissenschaftliche ‚Rückendeckung‘ in Sachen Relativitätstheorie. Desweiteren möchte ich mich bei meiner Mutter bedanken, die mich, soweit es ihr möglich ist, im Studium unterstützt. Zuletzt möchte ich mich noch besonders bei allen Freunden bedanken, die die Arbeit Korrektur gelesen haben.