

**Konzept zur Unterstützung des Produktentwicklungsprozesses  
mittels Virtual Reality**

Von der Fakultät für Ingenieurwissenschaften, Abteilung Maschinenbau der  
Universität Duisburg-Essen  
zur Erlangung des akademischen Grades

DOKTOR-INGENIEUR

genehmigte Dissertation

von

Jörg Lubnau

aus

Hattingen

Referent: Prof. Dr.-Ing. Diethard Bergers

Korreferent: Prof. Dr.-Ing. habil. Bernd Künne

Korreferent: PD Dr.-Ing. Frank Lobeck

Tag der mündlichen Prüfung: 23.03.2009



## **Vorwort**

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter, an der Professur für Produktionstechnologie und Produktentwicklung - Fakultät für Ingenieurwissenschaften - der Universität Duisburg-Essen.

Mein besonders herzlicher Dank gilt Herrn Prof. Dr.-Ing. Diethard Bergers für die Schaffung der Rahmenbedingungen, die diese Dissertation ermöglicht haben. Durch seine fachliche Unterstützung und die hilfreichen Diskussionen hat er maßgeblich zum Gelingen dieser Arbeit beigetragen.

Für die Übernahme des Korreferates und die Unterstützung bei der Fertigstellung der Arbeit bedanke ich mich bei Herrn Prof. Dr.-Ing. habil. Bernd Künne und bei Herrn PD Dr.-Ing. Frank Lobeck. Ebenso gilt Herrn Prof. Dr.-Ing. Hans-Joachim Stracke für die wohlwollende Unterstützung und die freundliche Motivation mein besonderer Dank.

Im Rahmen der vorliegenden Arbeit wurden am Institut für Produkt Engineering eine Vielzahl an Stereoprojektionen durchgeführt. Für die uneingeschränkte Unterstützung bei der Bereitstellung und Instandhaltung der Projektionsanlage und den fachlichen Diskussionen gilt Herrn Klaus-Dieter Werner mein bester Dank.

Ferner bedanke ich mich bei Markus Hastrich, Timon Mallepree, Jens Standke, und Reinhard Giese für deren Unterstützung und stete Diskussionsbereitschaft.

Bei meiner Familie bedanke ich mich für die Unterstützung während des Studiums und der Promotion und das Verständnis auch in schwierigen Situationen.

<b>1</b>	<b>EINLEITUNG .....</b>	<b>1</b>
<b>2</b>	<b>GRUNDLAGEN DER PRODUKTENTWICKLUNG .....</b>	<b>3</b>
2.1	Wechselwirkungen von Markt und Produkt.....	3
2.2	Definition der Wertschöpfung .....	4
2.3	Produktentwicklungsprozess .....	5
2.4	Auswirkungen der Zielgröße Zeit .....	9
2.5	Die Konstruktion als Kernprozess der Produktentwicklung.....	11
2.6	Grenzen des Rapid Prototyping .....	14
2.7	Möglichkeiten des Virtual Prototyping .....	16
2.7.1	Flächenprojektionssysteme .....	18
2.7.2	Das Projektionsverfahren mittels Polarisierung.....	21
2.7.3	Die Visualisierungspipeline.....	25
2.8	Fazit .....	27
<b>3</b>	<b>ANFORDERUNGEN AN EINE PC BASIERTE VR-UMGEBUNG.....</b>	<b>30</b>
3.1	Die Integration in den bestehenden Produktentwicklungsprozess .....	30
3.2	3D-CAD als Kernkomponente zur Erzeugung von VP-Daten .....	31
3.3	Ein allgemeingültige VP-Datenformat.....	31
3.4	Darstellung der virtuellen Prototypen.....	32
<b>4</b>	<b>ANSATZPUNKTE FÜR EINE PC BASIERTE VR-UMGEBUNG .....</b>	<b>34</b>
4.1	Wahrnehmungsmechanismen.....	35
4.2	Der manuelle Visualisierungsprozess .....	44
4.3	Transformationsparameter .....	46
4.3.1	CAD-Parameter .....	47
4.3.2	Das systemneutrale Datenformat .....	51
4.3.3	Nutzbare Parameter .....	54
4.4	Steigerung des Immersionsgrades .....	56
4.4.1	Farbe.....	57
4.4.2	Beleuchtungsmodelle .....	66
4.4.3	Schattierungsverfahren.....	67
4.4.4	Farb- und Materialparameter .....	69
4.4.5	Ghosting-Reduktion .....	78
4.4.6	Sphärische Perspektive .....	83
4.4.7	Fazit .....	88
<b>5</b>	<b>SOFTWARETECHNISCHE REALISIERUNG .....</b>	<b>89</b>
5.1	CAD-Integration.....	94
5.2	VR-Modul .....	102
<b>6</b>	<b>BEISPIELHAFT VERIFIKATION .....</b>	<b>113</b>
6.1	Einsatz bei verteilten Standorten .....	113
6.2	Automatisierter Visualisierungsprozess .....	116

---

<b>7</b>	<b>ZUSAMMENFASSUNG UND AUSBLICK .....</b>	<b>120</b>
<b>8</b>	<b>LITERATUR.....</b>	<b>122</b>
<b>9</b>	<b>ANHANG .....</b>	<b>126</b>
<b>9.1</b>	<b>Abbildungsverzeichnis.....</b>	<b>126</b>
<b>9.2</b>	<b>Abkürzungen - Indizes.....</b>	<b>130</b>
<b>9.3</b>	<b>Tabelle und Abbildungen .....</b>	<b>132</b>

## 1 EINLEITUNG

Die weltweite Globalisierung und die damit verbundene Öffnung der Märkte führt insbesondere in den westlichen Industrieländern zu einem verstärkten Wettbewerbsdruck. Dies bedeutet, dass die Unternehmen gezwungen sind, ständig neue hoch entwickelte Produkte auf dem Weltmarkt anzubieten, um im globalen Wettbewerb auch langfristig zu bestehen [Gau00]. Durch den Wunsch nach immer schnelleren Produktinnovationen werden die Produktzyklen signifikant verkürzt. Dies erfordert eine ständige Optimierung der Produktentwicklungsprozesse. Hier kann das Konzept des Simultaneous Engineering (SE) zum Einsatz kommen. Simultaneous Engineering modularisiert und parallelisiert den in der Vergangenheit sequentiell abgelaufenen Entwicklungsprozess. Der durch SE hervorgerufene hohe Parallelisierungsgrad im Produktentwicklungsprozess erfordert jedoch eine erhöhte Kommunikationsdichte aller am Entwicklungsprozess beteiligten Parteien.

Um den gesteigerten Herausforderungen des Kommunikationsbedarfes in der Produktentwicklung zu begegnen, kommen virtuelle Prototypen (VP) zum Einsatz. Virtuelle Prototypen, deren Eigenschaften mit Hilfe von Computern simuliert werden [Hen97], erlauben eine fast völlige Parallelisierung der Entwicklungsarbeit. Arbeitsfortschritte bleiben damit in allen Entwicklungsstufen transparent. Virtual Prototyping kann in Kombination mit Virtual Reality (VR) als Hilfsmittel den Nutzen bieten, eine Abstimmung unterschiedlicher Abteilungen wie Einkauf, Vertrieb oder Produktion durch ein kostengünstiges Werkzeug im Produktentwicklungsprozess zu ermöglichen. Virtual Reality hat sich damit als wichtiges Instrument in der Produktentwicklung etabliert. VR wird unter anderem als Kommunikationsmodell bei der Parallelisierung von Prozessen in den verschiedenen Entwicklungsphasen eingesetzt und trägt deshalb einen erheblichen Teil zur Verkürzung der Entwicklungszeit bei [Spu97] [Hoy06].

Der Einsatz der VR-Technologie erfordert jedoch für anspruchsvolle immersive<sup>1</sup> Umgebungen Hochleistungsrechner, die erhebliche Investitionen bedingen. Als ein weiteres Defizit ist zudem die aufwendige Prozesskette zur Erzeugung der virtuellen Modelle zu nennen. Dafür wird Fachpersonal mit Expertenwissen benötigt. Vor diesem Hintergrund kommt die VR-Technologie heute meist nur in Großunternehmen zur Anwendung, da die Investitionshöhe, Betriebskosten und Personalkosten insbesondere für KMU nicht wirtschaftlich erscheinen. Eine weite Verbreitung dieser Technologie wird daher erst möglich, wenn Lösungen auf PC-Basis existieren.

Die Durchdringung der VR-Technologie in KMU wird durch erste Softwarepakete auf dem Markt unterstützt, die VR auf PC-Basis grundsätzlich ermöglichen. Die aufwendige Prozesskette zur Erstellung der Modelle besteht jedoch weiterhin. Die Anforderung an Photorealismus mit Echtzeitinteraktion wird auf PC-Basis zudem noch nicht erfüllt. Diese

---

<sup>1</sup> Immersion beschreibt im Kontext der virtuellen Realität das Eintauchen in eine künstliche Welt.

enormen Defizite sind auf die Leistungsfähigkeit der PC- und Softwarearchitektur zurückzuführen.

Für die Darstellung der VR-Modelle sind stereoskopische Projektionssysteme notwendig. Diese Projektionssysteme können in aktive und passive Systeme untergliedert werden (vgl. Abschnitt 2.7). Insbesondere bei der Kommunikation in Produktentwicklungsprozessen besteht häufig die Anforderung, dass eine Visualisierung zeitgleich für mehrere Nutzer ausgelegt sein muss. Da für KMU preiswerte VR-Lösungen anzustreben sind, rücken passive Systeme in den Vordergrund, da sie im Vergleich zu aktiven Systemen bei einem Multiusereinsatz zu geringeren Kosten zu realisieren sind.

Ziel dieser Arbeit ist die Untersuchung der gesamten Prozesskette zur Erstellung von VR-Modellen auf Single-Desktop-Basis im Hinblick auf deren Verkürzung. Dazu werden zum einen die Prozessarchitektur selbst analysiert und Optimierungskonzepte erarbeitet, zum anderen aber auch die physiologische Fragestellung untersucht, ob und inwieweit eine Reduzierung des Darstellungsgrades ohne nachhaltige Störung des immersiven Eindrucks erreicht werden kann.

In Kapitel 2 wird hierzu zunächst auf die Grundlagen der Produktentwicklung und insbesondere auf den wichtigsten Faktor einer erfolgreichen Produktentwicklung, der ‚time-to-market‘ eingegangen. Dabei richtet sich der Focus vor allem auf die Möglichkeiten, eine Reduktion der Entwicklungszeit mittels Virtual Reality zu erreichen. Anschließend werden in Kapitel 3 die Anforderungen an eine PC basierte VR-Umgebung formuliert. Aufbauend darauf werden in Kapitel 4 auf der Basis der definierten Anforderungen Ansatzpunkte für eine Realisierung aufgestellt. Die Realisierung wird in Kapitel 5 in Form eines Gesamtkonzeptes vorgestellt. Nach der Beschreibung des Gesamtkonzeptes, wird in Kapitel 6 eine beispielhafte Verifizierung anhand konkreter Beispiele durchgeführt.

## 2 GRUNDLAGEN DER PRODUKTENTWICKLUNG

### 2.1 Wechselwirkungen von Markt und Produkt

Das Ziel produzierender Unternehmen ist die Herstellung und der Vertrieb von Produkten, um Marktbedürfnisse zu befriedigen. Dies bedeutet, ein Unternehmen muss die Bedürfnisse der Kunden mit Hilfe von Produkten bedienen, damit diese bereit sind, den von den Unternehmen gewünschten Preis für deren Erwerb zu bezahlen.

Die Kaufbereitschaft der Konsumenten für die angebotenen Produkte wiederum sichert den Umsatz und damit auch den wirtschaftlichen Erfolg eines Unternehmens. „Aus diesem Grund ist der Kunden- bzw. Marktbezug eine der wichtigsten Aufgaben von Unternehmen“ [Ehr03]. Dies bedeutet, die Unternehmen müssen ihre Geschäftsfelder mit den Stellgrößen *Produkt*, *Markt* und *Unternehmen* (vgl. **Abbildung 2-1**) so optimieren, damit ein größtmöglicher Erfolg erreicht wird.

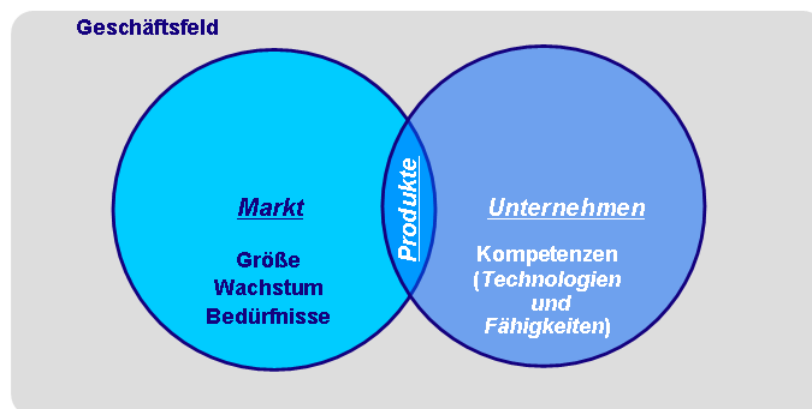


Abbildung 2-1: Das Geschäftsfeld [Ber07a]

Das *Produkt* wird als Bindeglied zwischen den Anforderungen des *Marktes* und den bestehenden Kompetenzen eines *Unternehmens* wie beispielsweise der genutzten Technologie und der vorhandenen Fähigkeiten eingesetzt. Das *Unternehmen* muss daher auf die Größe seines Marktanteils, der damit verbundenen Wachstumsrate und insbesondere auf die Kundenbedürfnisse mit entsprechenden Produkten bzw. Produktvarianten reagieren.

Eine Beeinflussung der Stellgröße *Produkt* wird durch neue Produkte, mit den Möglichkeiten der Produktdifferenzierung und mittels Produktdiversifikation erreicht. Die Produktdifferenzierung beinhaltet dabei alle Grade der Veränderung (Modifizierung) bestehender Produkte. Produktdiversifikation ergänzt dem gegenüber das bestehende Produktsortiment durch neue Produkte, die von den bisherigen genügend verschieden sind [Ber07a].



Auf die Stellgröße *Markt* kann mit einer Klassifizierung von Märkten in Teilmärkte (Marktsegmentierung) über gewichtete Kaufkriterien reagiert werden. Dieser marktorientierten Strategie - Market-Pull - steht die technologieorientierte Strategie - Technologie-Push - gegenüber. Im Rahmen der Strategie Technologie-Push besteht die Option neue Produktinnovationen auch mit Hilfe der Ausschöpfung des Differenzierungspotentials einer Technologie durch das Unternehmen zu initiieren. Zudem können Technologiesprünge als Impulse für neue Produkte genutzt werden. Damit ist es möglich durch die Verknüpfung vorhandener Fähigkeiten und der Bündelung von Kernkompetenzen mit neuen Technologien dem Markt neue Produkte anbieten zu können, um Bedürfnisse zu wecken und somit zum Unternehmenserfolg beizutragen. Ein Erfolg wird aber nur dann erreicht, wenn der durch das alte Produkt bereitgestellte Kundennutzen vom neuen Produkt mindestens erreicht, oder besser noch übertroffen wird. Um eine Ablösung alter Produkte zu erreichen, muss die geänderte Technologieintegration in den neuen Produkten also einen höheren Kundennutzen bieten.

Als Produkte werden hierbei bis in die Gegenwart meistens nur Sachgüter und teilweise sogar nur Rohstoffe als Erzeugnisse der Urproduktion verstanden [Gut83]. Den sachgutbezogenen Produkten stehen dann Dienstleistungen als immaterielle Güter gegenüber. Dienstleistungen können die Herstellung von Software, Finanzdienstleistungen oder individuelle Beratungsleistungen sein [Spu97]. Der Produktbegriff wird häufig unterschiedlich erklärt, aber die drei gebräuchlichsten Definitionen verschiedener Normen (VDI-Richtlinie 2221, DIN EN ISO 8402, DIN ISO 10303) zeigen, dass zur Herstellung eines Produktes immer auch ein Prozess notwendig ist. Die unterschiedlichen Begriffsdefinitionen erlauben auch immaterielle Güter als Produkte zu verstehen. Zur Herstellung von Produkten ist also stets ein Produktionsprozess notwendig. Die Produktion wird von Spur [Spu96] als Erzeugung von Sachgütern und nutzbarer Energie sowie die Erbringung von Dienstleistungen und damit von Produkten durch Produktionsfaktoren in der Wertschöpfungskette definiert. Produktionsfaktoren sind dabei alle zur Erzeugung verwendeten Güter und Dienste.

## 2.2 Definition der Wertschöpfung

Die Wertschöpfung ist das Ziel produktiver Tätigkeit. Diese wandelt vorhandene Güter in Güter mit höherem Nutzen und damit in Güter höheren Geldwertes. Dabei werden in einem produzierenden Unternehmen Materialien oder Dienstleistungen mit Hilfe der Produktionsfaktoren *Mensch*, *Maschine* und *Material* zu Materialien oder Dienstleistungen höheren Wertes transformiert (vgl. **Abbildung 2-2**).

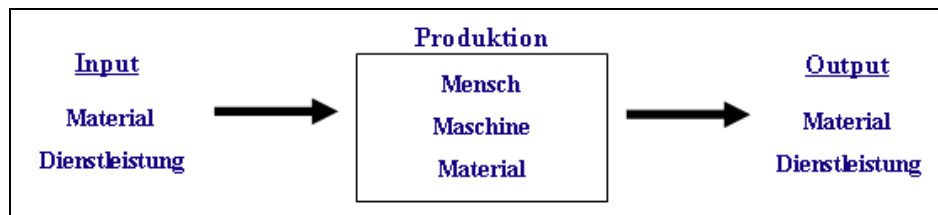


Abbildung 2-2: Wertschöpfung [Ber07b]

Die gesamte Wertschöpfungskette, die in einem produzierenden Unternehmen durchlaufen wird, startet mit der Lieferung von Gütern oder Dienstleistungen von einem Lieferanten, an die sich die Produktentwicklung und die Arbeitsvorbereitung anschließt und die über die Produktion und den Vertrieb bis hin zum Service für den Kunden weitergeführt wird (vgl. **Abbildung 2-3**).



Abbildung 2-3: Wertschöpfungskette [Ber07b]

Diesen technischen Wertschöpfungsprozess bezeichnet Seibert [Sei98] als Prozess der stufenweisen Steigerung des gesellschaftlichen Nutzens einer technischen oder technologischen Idee. Als Kenngröße kommt der Realisierungsdauer eine besondere Bedeutung im Wertschöpfungsprozess zu. Die Verkürzung der Entwicklungszeiten und damit eine Beschleunigung des Produktentwicklungsprozesses rückt in den Focus einer schnellen Markteinführung.

### 2.3 Produktentwicklungsprozess

Unter dem Begriff „Produktentwicklung“ ist eine zielgerichtete Planung von Handlungsanweisungen zu verstehen, die den planerischen, gestalterischen und organisatorischen Anteil einer Produktentstehung beinhaltet [Spu97]. Konkret wird die Produktentwicklung in die Einzelschritte Produktplanung, Produktkonzeption, Produktdefinition und Produkterprobung gegliedert (vgl. **Abbildung 2-4**).

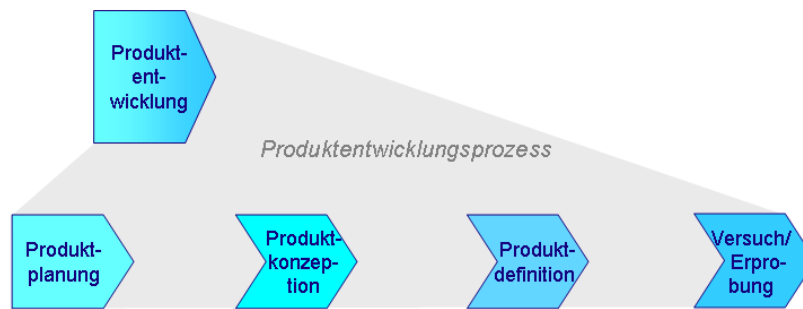


Abbildung 2-4: Produktentwicklungsprozess [Ber07a]

Der Produktentwicklungsprozess wird in der Produktplanung mit dem Finden eines für ein Unternehmen langfristig strategisch interessanten Betätigungsfeldes begonnen. Daran schließt die Produktkonzeption an, in der die Konzeption in Form der Grobgestaltung des Produktes, einer Prüfung der Machbarkeit, Auswahl von Alternativen und die Anforderungen an das Produkt bestimmt werden. Die Feingestaltung des Produktes bezüglich der Funktionalitäten, der Werkstoffauswahl, der Konstruktion und die Erstellung von technischen Dokumentationen wird in der Produktdefinition durchgeführt. Abgeschlossen wird der Produktentwicklungsprozess durch das Testen der Produkteigenschaften durch Überprüfung und Optimierung mittels Simulation oder mit Hilfe von Prototypen.

Globalisierung, offene Märkte und der dadurch entstehende internationale Wettbewerb führt zu immer kürzeren Produktlebenszyklen und fordert von den Unternehmen eine schnelle Entwicklungszeit für neue Produkte. Die Entwicklungszeit ist damit zu einem entscheidenden Erfolgsfaktor für viele Unternehmen geworden. Damit verbunden ist die Forderung nach einer datentechnischen Kopplung der Einzelaufgaben in der Prozesskette, da hiermit eine Optimierung des Produktentwicklungsprozesses erreicht werden kann. Die Optimierungsmaßnahmen müssen auf hohe Qualität, niedrige Entwicklungs- und Fertigungskosten und auf kurze Entwicklungszeiten ausgerichtet sein. Wie in **Abbildung 2-5** dargestellt, stehen unter anderem die beiden Stellhebel, *Managementmethoden* und *technische Methoden* zur Optimierung und Verkürzung des Produktentwicklungsprozesses zur Verfügung.

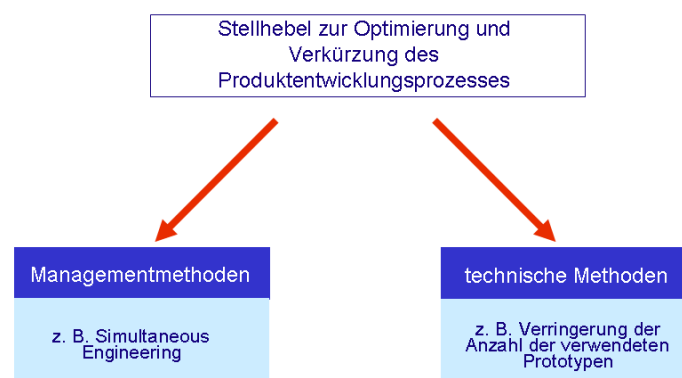


Abbildung 2-5: Stellhebel zur Verkürzung des Produktentwicklungsprozesses [Ber07c]

Technische Methoden zur Verkürzung des Produktentwicklungsprozesses sind neben der Wertanalyse zur Realisierung von Produktfunktionen zu den geringstmöglichen Kosten und der Fehlermöglichkeiten- und Einflussanalyse (FMEA) zur Vermeidung und Behebung von Fehlern in einer möglichst frühen Entwicklungsphase der Einsatz von Modellen. Zu den Varianten der Managementmethoden zählen unter anderem das Qualitätsmanagement, Design to X und das Simultaneous Engineering (SE). Mit der Methode des Simultaneous Engineering kann eine Modularisierung und Parallelisierung des häufig in viele Phasen unterteilten Produktentwicklungsprozess (vgl. **Abbildung 2-6**) erreicht werden.

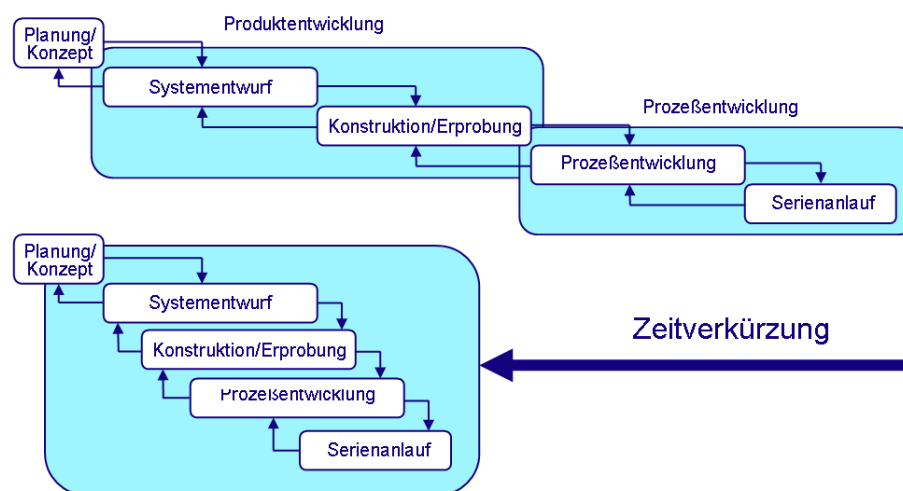


Abbildung 2-6: Zeitverkürzung mittels Simultaneous Engineering [Ber07c]

Da die Erarbeitung von Produktinnovationen häufig als Projekte definiert werden, erfordert die Parallelisierung von Aufgaben im Rahmen von Simultaneous Engineering ein leistungsfähiges Projekt- und Zeitmanagement. Ist es nicht möglich, die parallel ablaufenden Prozesse so zu planen, dass sie aufeinander abgestimmt sind, kann das gesamte Projekt gefährdet sein. Nach Henning [Hen97] werden 75% aller Produktmängel in der Entwicklungsphase verursacht, wobei 80% dieser Mängel bis zur Markteinführung nicht entdeckt werden. Bei den restlichen 20% ist eine spätere Änderung des Produktes aus Rentabilitätsgründen oft nicht mehr möglich. Diese Defizite können in einem mangelhaften Projekt-, Zeit- und Kommunikationsmanagement begründet sein.

Im Vergleich zu einer sequentiellen Projektdurchführung kann mit Simultaneous Engineering die jedoch Projektdauer beträchtlich reduziert werden. Die zentrale Forderung besteht also darin, dass Produkte zukünftig von unterschiedlichen Fachgebieten und größeren interdisziplinären Entwicklerteams, die räumlich voneinander entfernt sind, zu geringeren Kosten, mit höchster Qualität für kurze Produktlebenszyklen zu entwickeln. Die Umsetzung von kurzen Entwicklungszeiten mittels Simultaneous Engineering verlangt also ein interdisziplinäres Zusammenarbeiten und damit eine reibungslose Kommunikation. Als

Werkzeug zur Verbesserung der Kommunikation, kommen neben Modellen auch Prototypen zum Einsatz [Ber07c].

Grundsätzlich wird in Funktionsmodelle und Anschauungsmodelle unterschieden. Um die Funktion und Arbeitsweisen von Produkten zu untersuchen, kommen Funktionsmodelle zum Einsatz, um Ergebnisse für weitere Entscheidungen zu erhalten. Zu reinen Diskussions- und Kommunikationszwecken, beispielsweise für Kundengespräche oder Präsentationen, werden Anschauungsmodelle (Kommunikationsmodelle) genutzt. Mit Modellen ist es möglich, dass sich Fachleute unterschiedlicher Disziplinen über den Entwicklungsstand mit den fachspezifischen Anforderungen austauschen können. Durch Modelle ist damit die Kommunikation innerhalb von Entwicklungsteams und zwischen unterschiedlichen Fachabteilungen für eine frühzeitige Fehlererkennung, Fehlerbehebung und die Überprüfung der Konzepte positiv zu beeinflussen. Seibert [Sei98] fasst die Vorteile von Simultaneous Engineering gegenüber der phasenorientierten Produktentwicklung wie folgt zusammenfassen:

- Schnellere und flexiblere Projektdurchführung: Durch Simultaneous Engineering können die Entwicklungszeiten um 25 bis 50 % verkürzt werden. Verzögerungen können im weiteren Projektverlauf wieder eingeholt werden. Auf veränderte Markterfordernisse und Kundenwünsche kann schneller reagiert werden.
- Bessere bereichsübergreifende Zusammenarbeit: Die Barrieren zwischen den beteiligten Unternehmensbereichen werden durch die SE-Team Struktur überwunden. Know-how und Anforderungen anderer Bereiche fließen dadurch frühzeitiger in das Projekt ein. Es treten weniger Informationsverluste bei der Übergabe von Arbeitsergebnissen zwischen den beteiligten Bereichen auf. Die Mitarbeiter in den SE-Teams zeichnen sich durch eine höheres Verantwortungsbewusstsein und ein stärkeres gesamtunternehmensbezogenes Denken aus.
- Bessere Ergebnisqualität: Durch das frühzeitigere und intensivere Testen von Mustern und Prototypen werden Probleme früher erkannt und behoben. So werden ausgereifere, qualitativ bessere Produkte erzielt und der spätere Änderungsaufwand kann deutlich vermindert werden.

Das Erzeugen von Prototypen wird beim Simultaneous Engineering nicht von bestimmten Meilensteinen abhängig gemacht, sondern erfolgt nach geplanten Terminen. Dabei wird das „Prototyping“ als iterativer Versuchsprozess zur Optimierung der Anforderungen zum jeweiligen Zeitpunkt eingesetzt. „Unter Prototyping wird der Bau und die Erprobung von Prototypen verstanden, um die Funktionsfähigkeit technischer Lösungen nachzuweisen, deren technische Eigenschaften zu ermitteln und zu optimieren sowie vorhandene Mängel zu beseitigen“ [Seib98].

Der Einsatz von Prototypen erhöht zwar die Kosten, aber die daraus erwachsenen Vorteile übertreffen diese Kostenerhöhungen bei weitem. Die Vermeidung von Verzögerungen aufgrund mangelnden Produktverständnisses oder unzureichender Kommunikation zwischen beteiligten Personen ergibt eine erhebliche Reduzierung des Zeitbedarfes im Produktentwicklungsprozess. Mit Prototypen ist es ebenfalls möglich, Produktänderungen in einer frühen Phase mit wesentlich geringerem Aufwand zu beseitigen als in einer späteren Phase. Die Qualität des Produktkonzeptes kann mit Prototypen zudem auf Vollständigkeit und Richtigkeit der Spezifikationen überprüft und verbessert werden. Zur Herstellung und Erprobung von Prototypen für Serienprodukte werden jedoch mehr als 25% der Entwicklungszeit und bis zu 50% der Entwicklungskosten benötigt [Bul97].

Zwei Zielvoraussetzungen für den Einsatz von Prototypen in Produktentwicklungsprozessen sind die schnelle und kostengünstige Verfügbarkeit. Die Grundlage ist eine digitale Abbildung der Prototypen, um sie anschließend physisch oder virtuell zu erzeugen. Diese Ziele sind nur zu erreichen, wenn neue Werkzeuge zur Unterstützung des Produktentwicklungsprozesses konzipiert und eingesetzt werden. Bei der Konzeption neuer Werkzeuge kommt der Zielgröße Zeit demnach eine besondere Bedeutung zu.

## 2.4 Auswirkungen der Zielgröße Zeit

Eine eingehende Literaturrecherche bezüglich des Produktentstehungsprozesses hat gezeigt, dass sich alle unterschiedlichen Forderungen zu einer Zielgröße verdichten lassen: die „time-to-market“ [Geb96] [Mey95] [Sch89]. Darunter wird die Dauer von der Entscheidung zur Produktherstellung bis zu der Platzierung des Produktes auf dem Markt verstanden. McKinsey [Mey95] hat nachgewiesen, dass durch keine andere Maßnahme so viel Geld gewonnen werden kann, wie durch die Reduzierung der „time-to-market“ (vgl. **Abbildung 2-7**).

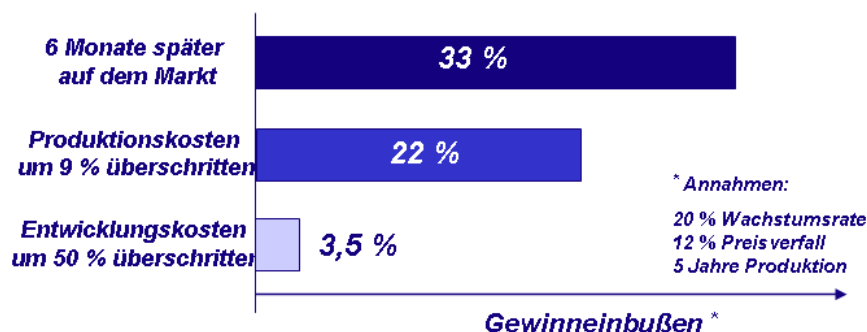


Abbildung 2-7: Einfluss unterschiedlicher Zielverfehlungen auf den erzielbaren Gewinn [Geb96]

McKinsey [Mey95] zeigt damit, dass eine Überschreitung der Entwicklungskosten um 50% den Gewinn nur um 3,5% verringert. Eine Überschreitung der Produktionskosten, reduziert den Gewinn bereits um 22%. Ein verspäteter Markteintritt um 6 Monate hat aber die folgenschwersten Auswirkungen auf den Gewinn. Die Gewinneinbußen belaufen sich in diesem Fall auf 33%.

Vergleichbare Aussagen trifft Schmelzer [Sch89]. Demnach reduziert sich der erzielbare Gewinn in wettbewerbsintensiven Märkten bei Produkten mit einer Vermarktungsdauer von 5 Jahren um folgende Beiträge:

- bei einer Erhöhung der Entwicklungskosten um 30 bis 50 %, wenn dadurch der Markteintrittszeitpunkt gehalten werden kann: 5 bis 10 % Gewinnreduktion;
- bei einer Überschreitung der geplanten Herstellkosten gegenüber der ursprünglichen Planung um 10 %: 10 bis 15 % Gewinnreduktion;
- bei einem qualitätsbedingten Preisnachlass um 10 %: 15 bis 20 % Gewinnreduktion;
- bei einem um sechs Monate verspäteten Markteintritt: 25 bis 30 % Gewinnreduktion.

Damit wird klar, warum die Entwicklungszeit und damit die „time-to-market“ als wichtigstes Kriterium für den wirtschaftlichen Erfolg einer Produktentwicklung genannt werden muss. Der Markt fordert von der Industrie in immer kürzerer Zeit immer bessere Produkte. So hat sich zum Beispiel der Produktlebenszyklus eines Autos in den letzten zehn Jahren um 12,5% verkürzt. Die Zeitspanne, in der sich die Entwicklung eines Automobils amortisiert, ist dagegen um 28 % länger geworden [Ste95].

Unternehmen mit einer Markteintrittsstrategie als „Schrittmacher“ können zu Beginn der Verwertungs- und Nutzungsphase häufig eine Monopolstellung aufbauen und - bis zum Eintritt von Unternehmen mit einer Strategie als „kreativer Nachahmer“ - Marktanteilssteigerungen und höhere Preise erzielen. Außerdem können sie aufgrund ihrer hohen technologischen Kompetenz Kostenvorteile realisieren, die sich ihre Wettbewerber erst aneignen müssen. Es gilt also wie oben beschrieben, den Produktentwicklungsprozess zu beschleunigen und damit den Schlüsselfaktor für den Erfolg eines Produktes und damit für den Erfolg eines Unternehmens, die „time-to-market“ zu verringern.

Als Notwendigkeit, um den Produktentwicklungsprozess zu unterstützen, kann ein Werkzeug genannt werden, mit dem es möglich ist, abteilungsübergreifend über Produktspezifika zu kommunizieren und damit schnell verbindliche Entscheidungen herbeizuführen. Dabei ist es wichtig, die Entscheidungen in einer sehr frühen Phase zu treffen, da die angefallenen bzw. verursachten Kosten in der frühen Phase des Produktentwicklungsprozesses noch sehr gering sind, aber dort bereits ein erheblicher Anteil der zukünftigen Kosten festgelegt werden.

Hier können Prototypen helfen, auch für nicht technische am Prozess beteiligte Disziplinen bzw. aus dem Bereich Marketing technische Details verständlich und beurteilbar zu machen. Um der genannten Notwendigkeit von Prototypen gerecht zu werden, haben sich in den letzten Jahren die Modellkonzepte des Rapid Prototyping und des Virtual Prototyping etabliert.

Unter dem Begriff Rapid Prototyping werden computergesteuerte, automatische Modellbauverfahren zur Herstellung von physischen Prototypen von Produkten und Werkzeugen verstanden [Ber07c]. Beim Rapid Prototyping stehen den am Produktentwicklungsprozess beteiligten Personen damit physische Modelle zur Beurteilung oder zur Prüfung in Form von Funktionsmodellen zur Verfügung. Somit ist es möglich, bei den stark parallelisierten Projektphasen des Simultaneous Engineering, ein Produktverständnis und damit eine Überprüfbarkeit sicherzustellen.

Mit Hilfe des Virtual Prototyping kann ein virtualisierter Prototyp erstellt werden, dessen Eigenschaften durch den Rechner simuliert werden [Hen97]. Das bedeutet, dass Aussagen über die Anordnung von Betriebsmitteln, Raumnutzung, Materialfluss und Beurteilungen über Funktion, Design und Geschmack in einer frühen Phase der Produktentwicklung getroffen werden können, in der es aufgrund fehlender Daten noch nicht möglich ist, physische Modelle zu erzeugen.

Grundlage beider Modellkonzepte ist die digitale Abbildung des Prototyps mit Hilfe bildgebender Verfahren, oder der Einsatz eines 3D CAD-Systems zur Erzeugung von Daten für das Rapid- oder Virtual Prototyping.

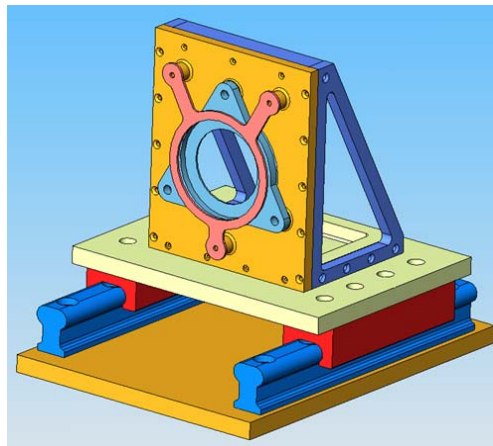
## **2.5 Die Konstruktion als Kernprozess der Produktentwicklung**

Wie in Abschnitt 2.3 bereits erläutert, beinhaltet der Produktentwicklungsprozess unter anderem den Prozessschritt der Produktdefinition. Aufgabe der Produktdefinition ist es, die Gestaltung des Produktes vorzunehmen. Die Gestaltung des Produktes beinhaltet neben der Bestimmung der Funktionalitäten, die Werkstoffauswahl sowie die Erstellung von technischen Dokumentationen wie beispielsweise Stücklisten oder Datenblätter.

In der Phase der Produktgestaltung richtet sich der Focus zunehmend auf den Konstruktionsprozess. Neben dem Erfinden, dem Entwerfen und dem Gestalten einzelner Bauteile wird unter dem Begriff der Konstruktion auch das Kombinieren von einzelnen Teilen zu Baugruppen verstanden. Konstruieren wird als zielorientierter, darstellender Prozess der Gestaltung von Teilfunktionen und deren Zusammensetzen zur Gesamtfunktion definiert [Rod91]. In heutigen Konstruktionsprozessen ist eine integrative Um- oder Neugestaltung von Produkten ohne Rechnerunterstützung nicht mehr denkbar.



In diesem Zusammenhang werden rechnerunterstützte Tätigkeiten unter dem Begriff CAD (**C**omputer **A**ided **D**esign) zusammengefasst. CAD ist zu einem integralen Bestandteil moderner Produktentwicklungsprozesse geworden. Die rechnerinterne Darstellung der modellierten Bauteile bildet den Schwerpunkt der Konstruktion. Das Unterscheidungsmerkmal von CAD-Systemen lässt sich in der rechnerinternen Darstellung der Geometrie finden. Hier wird unterschieden in zweidimensionale (2D) oder dreidimensionale (3D) Darstellung. Objekte in 2D-Modellen sind punkt, kontur und flächenorientiert und werden zum größten Teil für Zeichnungen genutzt. Daneben ermöglicht eine dreidimensionale Abbildung des Modells mit einem Volumenmodellierer eine spätere Ableitung von Zeichnungen, Stücklisten oder NC-Codes. Neben dem wirtschaftlichen Nutzen, das die in einem 3D-CAD-System generierten Produktdaten (vgl. **Abbildung 2-8**) eine Mehrfachnutzung ermöglichen, können Modelländerungen in einem Volumenmodellierer wesentlich einfacher durchgeführt werden.



*Abbildung 2-8: Produktdaten in einem 3D-CAD-System [Sol07b]*

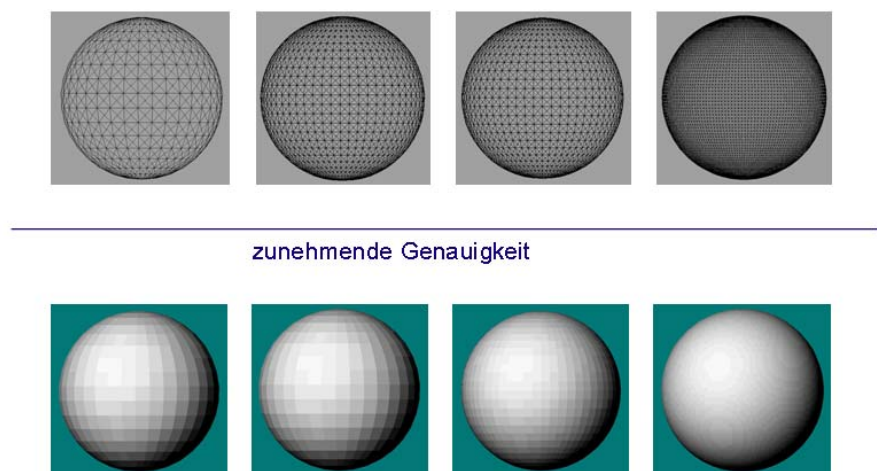
Die Konstruktionstätigkeit wird sich daher in Zukunft auf Volumenmodellierer konzentrieren. Neben der Featuretechnologie, die der Bereitstellung häufig verwendeter Funktionen dient, sind parametrische Funktionen als Unterstützung im Modellierungsprozess zu nennen. In parametrischen Konstruktionen wird die Geometrie mit variablen Parametern versehen, die einen automatisierten Neuaufbau des Modells durch Änderungen einzelner Parameter ermöglicht. Dies führt zu einer Erleichterung und Beschleunigung im Konstruktionsprozess [Spu97].

CAD-Systeme haben sich heute als Entwicklungswerkzeuge im Produktentwicklungs- und Konstruktionsprozess etabliert, da sie eine dreidimensionale Abbildung der Produkte erlauben und in der Lage sind, Informationen an andere am Produktentwicklungsprozess beteiligte Systeme weiterzugeben. Diese Weitergabeoption an andere Systeme ist insbesondere bei der zeitlichen und interdisziplinären Parallelisierung der Produktentwicklungsphasen im Sinne des Simultaneous Engineering von besonderer

Bedeutung. Für die Weitergabe an andere Systeme werden CAD-Schnittstellen genutzt, wobei jedoch häufig CAD-spezifische Parameter bei der Transformation in systemneutrale Datenformate verloren gehen, oder im Zielsystem nicht exakt wiedergegeben werden. Die Ein- und Ausgabe der Produktdaten an ein externes System erlauben allgemeingültig definierte Schnittstellenformate, wie *STEP*, *IGES*, *DXF*, *STL* oder *VRML*.

Für das Rapid- und das Virtual Prototyping kommen hauptsächlich die Datenformate *STL* (**S**tereolithographic **L**anguage) und *VRML* (**V**irtual **R**eality **M**odeling **L**anguage) zum Einsatz [Ber07c]. Während *STL* für die Bereitstellung von Objektdaten für generative Rapid Prototyping Verfahren eingesetzt wird, kommt *VRML* bei der Erstellung von Virtuellen Modellen zur Anwendung.

Die Gründe für die Wahl des *VRML*-Formates für die Konvertierung von CAD-Modellen in virtuelle Modelle, sind darin zu finden, dass Objekteigenschaften wie Farbe, Transparenz, Glanz oder Spiegelung als Objektparameter mit übertragen werden können. Diese Übertragung von Objekteigenschaften ist mit dem *STL*-Format nicht möglich. Die Erzeugung der Datensätze beider Datenformate erfolgt durch Tessellierung<sup>2</sup> der CAD-Objekte. Die Qualität und damit die Anzahl der Polygone werden durch den Sehnenfehler, der bei der Annäherung an gekrümmte Oberflächen auftritt, festgelegt. In **Abbildung 2-9** ist der Qualitätsanstieg und damit eine zunehmende Genauigkeit der Tessellierung einer Kugel durch die Reduzierung des Sehnenfehlers zu erkennen.



*Abbildung 2-9: Zunehmende Genauigkeit bei Reduzierung des Sehnenfehlers [Ber07c]*

Die so erzeugten Dateien werden anschließend als Grundlage für die Generierung von physischen Modellen mittels Rapid Prototyping oder zur Aufbereitung von virtuellen Modellen für den Einsatz in VR-Umgebungen genutzt.

## 2.6 Grenzen des Rapid Prototyping

Die vorhandenen Rapid Prototyping Verfahren werden unterschieden in generative und quasi-generative Verfahren. Assmann [Ass03] unterteilt die RP-Verfahren anhand einer Funktionsanalyse (vgl. **Abbildung 2-10**)

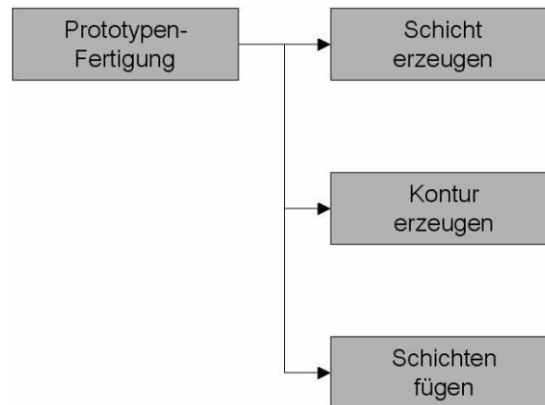


Abbildung 2-10: Funktionsanalyse der Prototypenherstellung [Ass03]

Wenn die Fertigungsschritte Schicht erzeugen, Kontur erzeugen und Schichten fügen gleichzeitig in einem Schritt erfolgen, kann von einem generativen Verfahren gesprochen werden. Werden einzelne Fertigungsschritte nacheinander durchgeführt, wird von einem quasi-generativem Verfahren gesprochen [Ass03].

Als Vorteil der generativen Verfahren kann die Möglichkeit genannt werden, sehr komplexe Strukturen zu erzeugen. Der wesentliche Nachteil der generativen Verfahren liegt in der begrenzten Werkstoffauswahl begründet. Im Gegensatz dazu umfasst beispielsweise die Werkstoffauswahl des quasi-generativen Verfahrens Fräsen alle spanbaren Werkstoffe. Hier muss aber auf die eingeschränkte Erzeugung von komplexen Geometrien mit Hinterschneidungen hingewiesen werden.

Ausgehend vom 3D-CAD Datensatz muss für den Einsatz eines generativen RP-Verfahrens die Geometrie häufig, wie in Abschnitt 2.5 erläutert, mittels Triangulation in das neutrale Datenaustauschformat *STL* konvertiert werden (vgl. **Abbildung 2-11**).

---

<sup>2</sup> Bei der Tesselierung werden Polygone in primitive Flächen, beispielsweise Dreiecke oder Vierecke zerlegt.

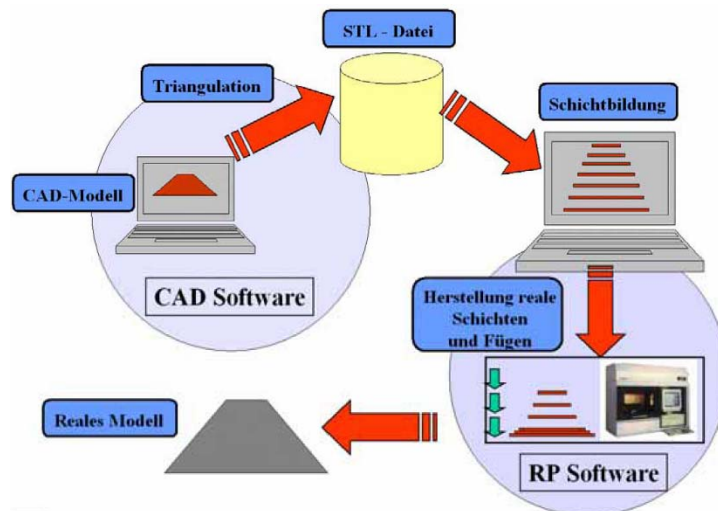


Abbildung 2-11: Prozess der Prototypenherstellung [Lob04]

Nach der Aufbereitung (z.B. Anordnung im Bauraum) des *STL*-Datensatzes und der Festlegung der Fertigungsparameter im RP-Softwaresystem wird der NC-Code für die vorgesehene RP-Maschine erzeugt. Dabei wird das Modell innerhalb der RP-Software in Schichten gleicher Dicke zerlegt und in dem anschließenden Fertigungsprozess sind diese Schichten zu erzeugen und zu fügen.

Insbesondere bei der Erstellung von Funktionsprototypen aus seriennahen Werkstoffen für die Überprüfung von Funktionen kommt dem Fräsen als quasi-generativem Verfahren eine besondere Bedeutung zu. Da heutige CAD-Systeme eine direkte CAM Bearbeitung durch integrierte CAM-Module ermöglichen, wird direkt auf Grundlage des 3D-CAD Modells ein entsprechender NC-Code für eine CNC-Fräsmaschine erstellt. Für die Erzeugung des CNC-Codes ist Fachwissen über besondere Bearbeitungsmethoden des Werkstoffes, Frässtrategien, Werkzeugwahl, und Spezifika der genutzten Fräsmaschine notwendig. Für die Bedienung der Maschine, wie das Bestücken mit Werkzeugen und das Fixieren des Halbzeugs auf dem Maschinentisch ist daher zusätzliches Fachpersonal erforderlich.

Um auch komplexe Geometrien mit Hinterschneidungen frästechnisch herstellen zu können, kann ein spezielles Verfahrensprinzip zur Anwendung kommen. Der herzustellende Prototyp wird zunächst schichtweise gefräst und anschließend wieder zusammengefügt. Dabei muss jedoch neben der Auswahl einer geeigneten Fügetechnik auch ein zusätzlicher Prozessschritt durchlaufen werden [Sta08]. Zusammenfassend kann gesagt werden, dass ein physischer Prototyp nicht auf „Knopfdruck“ herzustellen ist. Vielmehr muss eine Prozesskette durchlaufen werden, die unter Umständen einen erheblichen Ressourceneinsatz notwendig macht. Neben der Anschaffung von RP-Maschinen und der Bereithaltung von entsprechendem Bedienpersonal fallen je nach RP-Verfahren häufig erhebliche Kosten für die speziellen Modellbauwerkstoffe für die RP-Maschinen an [Ber07c].

Neben der Notwendigkeit einer physischen Erzeugung des Modells beim Rapid Prototyping, kann als weiterer Nachteil, das eingeschränkte Einsatzgebiet genannt werden. Das bedeutet, dass häufig nicht alle am Produktentwicklungsprozess beteiligten Personen und der in physischer Form vorhandene Prototyp zum geplanten Termin am gleichen Besprechungsort zusammenfinden können. Eine sinnvolle Kommunikation und Einbeziehung in den Entwicklungsprozess gestaltet sich dadurch als schwierig. Daraus können unterschiedliche Interpretationen von Anforderungen oder Design entstehen, die Entwicklungsfehler zur Folge haben.

## 2.7 Möglichkeiten des Virtual Prototyping

Die Möglichkeit 3D-CAD Daten als virtuelle Modelle darzustellen ist zum integralen Bestandteil des Produktentwicklungsprozesses geworden. Mit dem Virtual Prototyping ist es allen am Produktentwicklungsprozess Beteiligten möglich, das nicht mehr auf Zeichnungen beruhende Konzept zu verstehen. Die in der Vergangenheit bestehende Problematik, den Geometrieentwurf effektiv allen Projektteams mit unterschiedlichem fachlichem Hintergrund zu kommunizieren, entfällt damit. Die Voraussetzungen, Zeichnungen nach Form, Funktion und Entwurfsabsicht interpretieren zu können, sind nicht mehr nötig, da mit Hilfe von Virtual Reality die virtuellen Prototypen als dreidimensionale Objekte visualisiert werden können. VR definiert Henning als Mensch-Maschine-Schnittstelle, die es erlaubt, eine computergenerierte Umwelt in Ansprache mehrerer Sinne als Realität wahrzunehmen [Hen97]. Unter virtueller Realität wird in der Regel eine dreidimensionale Darstellung in Stereoprojektion verstanden. Dem rechten und dem linken Auge des Betrachters wird dabei ein jeweils um den Augenabstand versetztes Bild des Modells in der passenden Perspektive zugeführt (vgl. Abbildung 2-12).

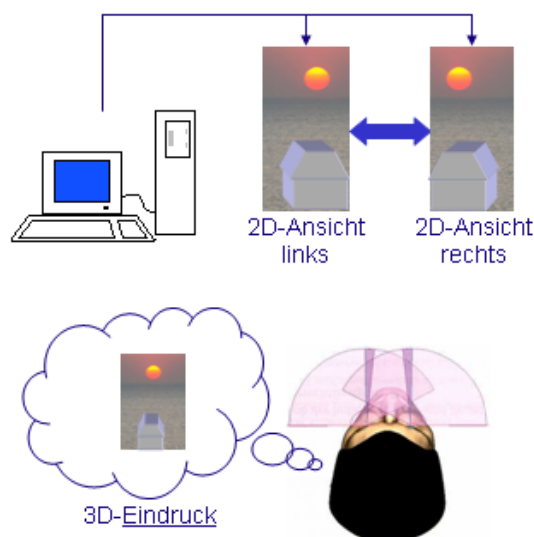


Abbildung 2-12: Stereoskopisches Sehen in Projektionssystemen [Ber07c]

Dadurch wird dem Betrachter ein räumlicher Eindruck der visualisierten Daten vermittelt. Der Immersionsgrad<sup>3</sup> einer VR-Umgebung ist durch die beiden Hauptmerkmale Interaktionsgrad und Darstellungsgrad gekennzeichnet. Interaktion beschreibt die Gestaltung einer Benutzerschnittstelle zu Programmen und Hardware um, Orientierung, Navigation, System-Reaktionen oder Meldungen durch Benutzeraktionen zu beeinflussen. Der Darstellungsgrad beschreibt die Qualität der Visualisierungen beispielsweise in Form von Bildauflösung, Bildrate oder Raumwahrnehmung. Bei totaler Immersion werden alle Bezüge zur realen Außenwelt abgebrochen und vollständig durch virtuelle Ersatzstimuli ersetzt. Der Benutzer ist bei totaler Immersion davon überzeugt, dass alles real ist. Der Immersionsgrad ist aber nicht als physikalische Größe messbar, da er ein subjektiver Eindruck des Betrachters ist.

Um eine virtuelle Welt möglichst real erscheinen zu lassen, ist ein sehr hoher Immersionsgrad erforderlich. Durch diese Anforderungen kommt dem eingesetzten Computer eine zentrale Bedeutung im VR-System zu. Er hat die Aufgabe die virtuelle Welt zu generieren und deren Zusammenspiel mit Ein- und Ausgabegeräten zu koordinieren. Neben den Rechnerkomponenten wie z.B. CPU oder Hauptspeicher ist die Grafikkarte von besonderer Bedeutung, da sie für die Ausgabe und damit für die Visualisierung der Daten genutzt wird. Neben einer hohen Auflösung für eine photorealistische Darstellung von Modellen, muss die Grafikkarte für die Visualisierung von großen, bewegten und komplexen 3D-Objekten eine möglichst hohe Anzahl von Polygonen pro Sekunde abbilden können.

Für die Ausgabe der Stereobilder besteht zum einen die Möglichkeit, eine Grafikkarte zu nutzen, welche über zwei Grafikkartenausgänge und einer Stereo-Buffering<sup>4</sup> Funktion für die zeitgleiche Ausgabe der unterschiedlichen Stereobilder verfügt. Zum anderen kann die Ausgabe der Stereobilder über zwei vernetzte Rechner oder über Rechnercluster erfolgen. Die von der Grafikkarte ausgegebenen Bilddaten müssen anschließend mit einem geeigneten stereoskopischen Sichtsystem für den Benutzer dargestellt werden. Hier kann eine Vielzahl unterschiedlicher Systeme zur Anwendung kommen. Es wird dabei in aktive und passive Systeme unterschieden.

Aktive Systeme zeichnen sich dadurch aus, dass sie die beiden unterschiedlichen Stereobilder abwechselnd auf einer Projektionswand oder einen Computerbildschirm abbilden. Mit einer Shutterbrille wird dem jeweiligen Auge dann nur das zugehörige Bild zugeführt. Bei aktiven Systemen ist eine Kommunikation über Kabel oder Infrarotschnittstellen zwischen Brille und Rechner erforderlich,

um die auf der Projektionsfläche dargestellten Bilder bei der jeweilig genutzten Bildfrequenz mit der Brille zu synchronisieren.

---

<sup>3</sup> Mit dem Immersionsgrad wird der Grad des Realitätseindrucks bewertet.

<sup>4</sup> Stereo-Buffering beschreibt ein spezielles Verfahren der Computergrafik, bei dem der Bildspeicher bei Grafikkarten für eine Stereoausgabe unterteilt wird, um eine gleichmäßige Bildfrequenz ohne Flackern zu erreichen.

Passive Systeme visualisieren die Bilder für das linke und das rechte Auge simultan auf der entsprechenden Projektionsfläche. Mit Hilfe spezieller Brillen können die Objekte dreidimensional wahrgenommen werden. Diese Brillen führen dabei im Vergleich zur Shutterbrille keine aktive Funktion aus. Als passives System zeichnet sich das Polarisationsverfahren durch seine uneingeschränkte Farbfähigkeit und seinen relativ günstigen Preis für große Anwendergruppen aus [Hen97]. Die Projektionsfläche kann je nach vorgesehener Anwendung gewählt werden. Verfügbare Lösungen reichen von tischgroßen Projektionsflächen (Workbenches) über Powerwalls für Auf- oder Rückprojektionsflächen, auf denen komplette Fahrzeugkarosserien dargestellt werden können, bis zu würfelförmigen Projektionsräumen, den CAVEs (Cave Automatic Virtual Environment).

### 2.7.1 Flächenprojektionssysteme

Unter einer Workbench wird im Allgemeinen eine Tischflächenprojektion verstanden. Dabei werden auf einer Art Schreibtisch über stereoskopische Rückprojektion dreidimensionale Modelle dargestellt (vgl. **Abbildung 2-13**).



*Abbildung 2-13: Workbench [BARCO]*

Sie wird als semi-immersives Projektionssystem für das Arbeiten in virtuellen Umgebungen eingesetzt. Dieses Darstellungssystem wurde 1992 am Fraunhofer Institut Medienkommunikation (IMK) entwickelt und derzeit von dem Unternehmen BARCO unter der Bezeichnung TAN Responsive Workbench als Darstellungssystem in den Abmaßen 1,80m x 1,10 m vertrieben. Am IMK wird die Workbench zur Entwicklung neuer Visualisierungs- und Interaktionstechniken und zur Erforschung verteilter virtueller Umgebungen eingesetzt.

Als Weiterentwicklung der Workbench kann die zweikanalige Workbench, die Holobench genannt werden. Die Holobench besteht aus zwei Projektionsebenen, die als horizontale und

vertikale Displays nahtlos gefügt und in einem definierten Winkel zueinander angeordnet sind (vgl. **Abbildung 2-14**).



*Abbildung 2-14: Holobench [RWTH Aachen]*

Die stereoskopische Darstellung wird von der Rückseite auf die Tisch- bzw. Arbeitsfläche projiziert und die 3D-Objekte können als auf der Arbeitsfläche schwebend vom Benutzer wahrgenommen werden. An der RWTH Aachen wird die Holobench zur Simulation von Werkzeugmaschinen genutzt. Die zu einem Virtual Prototyping Modell aufbereiteten Daten der Werkzeugmaschinen können durch Interaktion mittels Force Feedback vom Benutzer analysiert werden. Force Feedback ermöglicht eine Rückgabe der im Modell errechneten Kräfte beispielsweise als Bewegungswiderstände.

Als Großflächenprojektionssystem werden Powerwalls eingesetzt. Durch die Synchronisierung mehrerer Hochleistungsprojektoren werden mittels Rückprojektion die Objekte auf die Powerwall projiziert. Die Powerwall zeichnet sich insbesondere durch ihre Größe und die hohe Darstellungsqualität aus. Heutige Powerwallsysteme erreichen eine Detailgenauigkeit mit einer Auflösung von 9 Megapixel<sup>5</sup>. Die Forschungsarbeit umfasst den Bereich Datenexploration und Visualisierung. Die Automobilbranche ist damit in der Lage, Prototypen von neuen Fahrzeugmodellen bereits in Originalgröße zu betrachten, ohne einen physischen Prototypen fertigen zu müssen (vgl. **Abbildung 2-15**).

---

<sup>5</sup> Als Pixel wird die kleinste Einheit einer Rastergrafik bezeichnet.





Abbildung 2-15: Powerwall [BARCO]

Mehrseitige Visualisierungssysteme mit zwei, drei oder mehrseitigen raumhohen Projektionswänden, wie sie im Virtual Reality Center Aachen (VRCA) zum Einsatz kommen, die einen realen dreidimensionalen Eindruck vermitteln können, werden als CAVE bezeichnet (vgl. **Abbildung 2-16**).

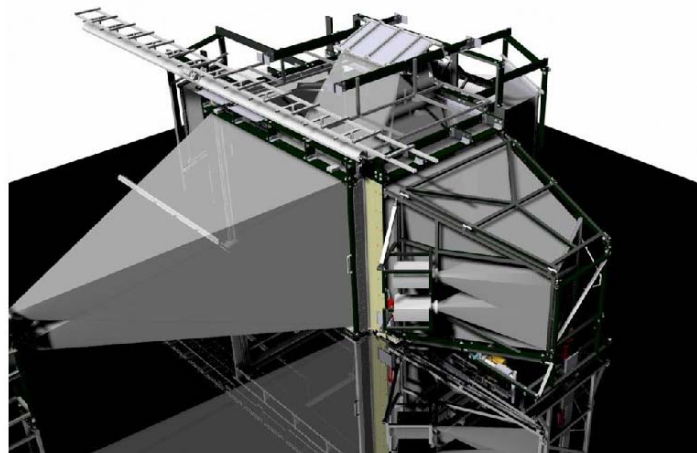


Abbildung 2-16: CAVE [VRCA]

Für eine stereoskopische Wahrnehmung der dargestellten Objekte sind für jede Projektionsfläche jeweils zwei Projektoren notwendig. Eine besondere Herausforderung bei dem Einsatz einer CAVE ist die Synchronisierung aller Projektionsflächen, um einem möglichst hohen Grad der Immersion zu erreichen. Bei der Nutzung einer CAVE kommen häufig Tracking-Systeme zur Anwendung. Tracking-Systeme ermöglichen eine Blickwinkelanpassung der visualisierten Szene an den jeweiligen Standort und die Blickrichtung des Betrachters. Bei einem vollimmersiven Interaktionsraum handelt es sich um

eine CAVE, deren sechs Seiten aus Projektionsflächen bestehen. Mit diesem System kann der höchste Grad an Immersion erzielt werden, da der Betrachter den visuellen Bezug zur realen Umgebung verliert. Am Fraunhofer Institut Arbeitswirtschaft und Organisation wird seit 2001 die 6-Seiten-CAVE HyPi-6 betrieben (vgl. **Abbildung 2-17**).



*Abbildung 2-17: 6-Seiten-CAVE HyPi-6 [Fraunhofer IAO]*

Mit diesem Werkzeug zur Erzeugung eines besonders hohen Immersionsgrades, können wissenschaftliche Untersuchungen an virtuellen Modellen mit einem komplett kubischen Tiefeneffekt durchgeführt werden. Neben Montagesimulationen werden am IAO mit Hilfe der HyPi-6 Untersuchungen zu Strömungssimulationen an Ventilen durchgeführt. Architekten und Städteplanern ermöglicht die HyPi-6 völlig neue Gestaltungsperspektiven. Ein weiterer Einsatz des 6-Seiten-CAVE wird neben Trainings- und Ausbildungsszenarien für Fahr- und Flugsimulationen in der Therapie von Höhenangst und Orientierungsschwierigkeiten gesehen.

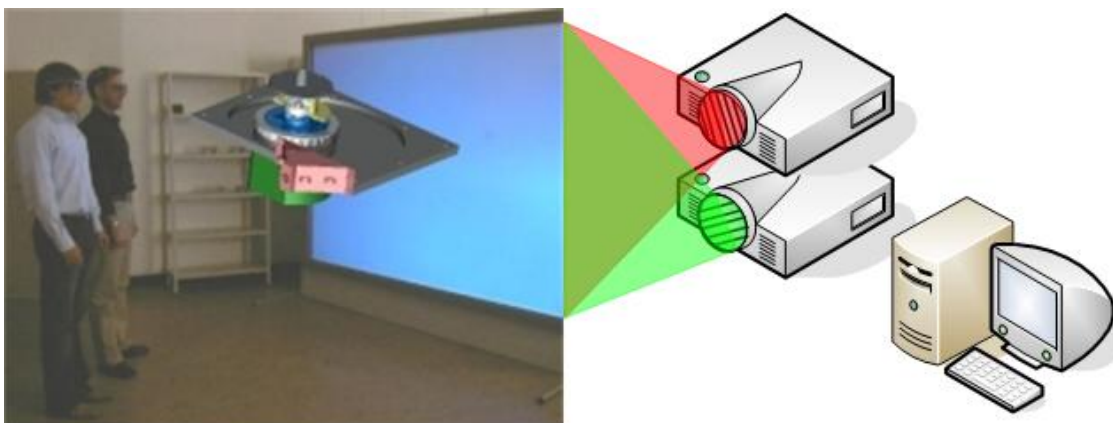
Da Flächenprojektionssysteme häufig mit dem Polarisationsverfahren betrieben werden, wird auf dieses Verfahren und die dafür notwendigen Komponenten nachfolgend näher eingegangen.

### **2.7.2 Das Projektionsverfahren mittels Polarisation**

Neben einem Computersystem mit der Fähigkeit zwei unterschiedliche Stereobilder zeitgleich auszugeben, ist mindestens eine Projektionsfläche, Polarisationsbrillen für die

Benutzer und zwei Projektoren mit entsprechend vor geschalteten Polarisationsfiltern notwendig. Projektoren mit der LCD<sup>6</sup>-Technologie werden den Projektoren mit DLP<sup>7</sup>- oder CRT<sup>8</sup>-Technologie häufig vorgezogen, da sie leichter verfügbar und kostengünstiger sind [Ste01].

Die Trennung der durch die Projektoren abgegebenen Bilder wird beim stereoskopischen Polarisationsverfahren durch polarisiertes Licht erreicht. Das elektromagnetische Wellenspektrum des Lichts wird durch Polarisationsfilter gerichtet. Bei den Polfiltern wird unterschieden in lineare und zirkulare Polfilter. Der Vorteil von zirkularen Polarisationsfiltern gegenüber linearen Polfiltern ist die stabile Trennung der Bilder, wobei die Kopfhaltung des Betrachters unwichtig ist. Dem steht aber der Nachteil gegenüber, dass diese ideale Trennung der Bilder nur bei einer bestimmten Wellenlänge erreicht wird. Am Institute of Computing für Physics and Technologie, Protvino, Russia und am Fraunhofer Institut Medienkommunikation, Sankt Augustin wurde bei Untersuchungen festgestellt, dass bei anderen Wellenlängen keine komplette Bildtrennung erreicht wird, so dass farbige „Schatten“ entstehen [Kli03]. In der **Abbildung 2-18** ist eine komplette Projektionsanlage zu erkennen.



*Abbildung 2-18: Das Projektionsverfahren mittels Polarisation*

Neben den zwei Projektoren mit vor geschalteten Linear-Polarisationsfiltern, sind das Rechnersystem, die Projektionsfläche, das visualisierte Objekt und zwei Benutzer mit Polarisationsbrillen abgebildet. Die Bildkanäle werden durch die zueinander um 90° versetzt montierten Filter so polarisiert, dass die Polarisations Ebenen rechtwinklig zueinander angeordnet sind. Auf der transparenten Rückprojektionswand werden die Bildkanäle anschließend überlagert angezeigt. Die Polarisationsfilter in den Brillen der Betrachter müssen so ausgerichtet sein, dass nur das Teilbild wahrgenommen wird, welches für das

---

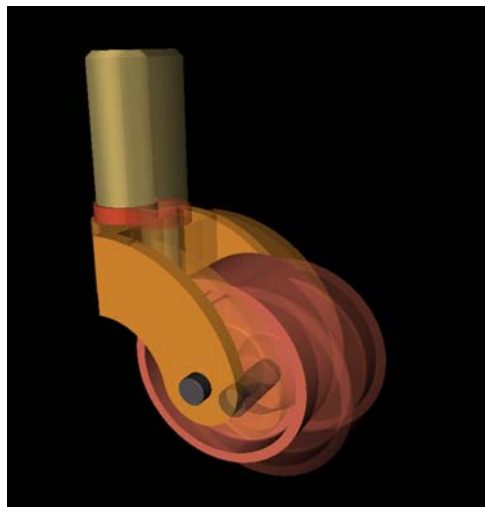
<sup>6</sup> LCD – Liquid Cristal Display

<sup>7</sup> DLP - Digital Light Processing

<sup>8</sup> CRT – Cathode Ray Tube

entsprechende Auge bestimmt ist. Im Gehirn werden die beiden unterschiedlichen Teilbilder zu einer räumlichen Szene zusammengefügt.

Dem Vorteil, dass die lineare Polarisation eine sehr kosteneffiziente Technik ist, um die Trennung der Teilbilder zu erreichen [Bar03], steht diesem Verfahren der Nachteil gegenüber, dass mit den zur Zeit zur Verfügung stehenden Polarisationsfiltern keine totale Polarisation zu erreichen ist. Aus diesem Grund wird ein geringer Teil des Bildkanals, welches beispielsweise für das rechte Auge bestimmt ist, auch mit dem linken Auge wahrgenommen (vgl. **Abbildung 2-19**).



*Abbildung 2-19: Crosstalk-Erscheinung*

Die nicht vollständige Trennung des rechten und linken Bildkanals wird als Crosstalk oder Ghosting bzw. Ghostview bezeichnet. Dabei wird unter Crosstalk das physikalische Vorhandensein einer nicht vollständigen Kanaltrennung und unter Ghosting die physiologische Wahrnehmung von Crosstalk verstanden [Ste97]. In Untersuchungen der Bauhaus Universität Weimar und dem Center for Mathematics and Computer Science, Amsterdam wird die Reduzierung von Crosstalk-Erscheinungen im Zusammenhang mit der Nachleuchtzeit auf Monitorbildschirmen und mit den nicht komplett opaken Kristallen in einer Shutterbrille beschrieben [Fro07] [Smi07]. Klimenko [Kli03] hingegen beschreibt die Reduktion von Artefakten auch für passive Stereoprojektionssysteme mittels linearen Polarisationsfiltern und verweist dabei auf die Grundidee von Limpscomb [Lim99]. In **Abbildung 2-20** wird das erzeugte Bild für das linke Auge mit **L** und für das rechte Auge mit **R** gekennzeichnet.

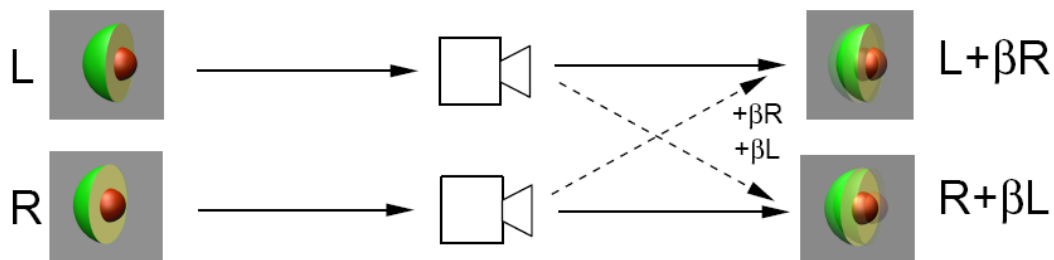


Abbildung 2-20: Entstehung des Geisterbildes [Kli03]

Diese vom Rechner erzeugten Bilder werden anschließend an die Projektoren gesendet. Wenn der Betrachter die Bilder nach dem Durchgang durch beide Polarisationsfilter (Projektor und Brille) sieht, nimmt das rechte Auge aufgrund mangelnder Filterung das rechte Bild **R** und Teile des linken Bildes **βL** wahr. Das gleiche gilt für das linke Auge. Der Betrachter hat somit Crosstalk Erscheinungen.

Durch den Einsatz von Softwaremethoden eliminiert Klimenko diese Artefakte, bevor die Bilder auf die Projektionsfläche projiziert werden. Vor der Bildausgabe durch die Grafikkarte wird in **Abbildung 2-21** verdeutlicht, wie nach Gleichung 2-1 vom linken Bild **L** ein Teil des rechten Bildes **αR** subtrahiert wird.

$$L' + \beta R' = (1 - \alpha\beta)L + (\beta - \alpha)R \quad (\text{Gl.: 2-1})$$

$$R' + \beta L' = (1 - \alpha\beta)R + (\beta - \alpha)L \quad (\text{Gl.: 2-2})$$

Das Bild **L'** wird erst jetzt an den Projektor ausgegeben. Gleiches wird mit dem rechten Bild nach Gleichung 2-2 durchgeführt.

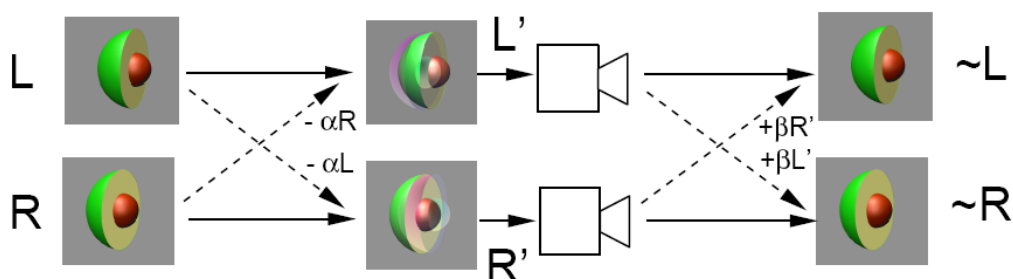


Abbildung 2-21: Eliminierung des Geisterbildes [Kli03]

Beim nachfolgenden Durchgang durch die Polarisationsfilter werden zu den Bildern **L'** und **R'** die entsprechenden Teilbilder, wie oben beschrieben aufgrund mangelnder Filterung wieder addiert und damit eine teilweise Geisterbildeliminierung erreicht.

Die notwendigen Rechenoperationen werden je nach Echtzeitrenderingleistung durch die Grafikkarte ermöglicht. Die beschriebene Methode verbessert durch die Reduktion von Geisterbildern wesentlich die Stereo-Wahrnehmung. Der Softwarealgorithmus der Substraktiven-Geisterbild-Reduktion (subtractive crosstalk reduction) wurde in das VE System AVANGO implementiert [Kli03]. Für die stereoskopische Darstellung von AVANGO-Szenen werden SGI-Rechner wie Onyx oder Onyx II eingesetzt, da diese Rechner für eine schnelle Verarbeitung der Daten mehrere Prozessoren und mehrere Visualisierungspipes einsetzen. Die softwaretechnische Umsetzung erfolgt mit der Grafiksoftware IRIS Performer von SGI. Diese eignet sich insbesondere für die Entwicklung von dreidimensionalen Echtzeit- und interaktiven Grafikanwendungen. IRIS Performer wurde für IRIX Workstations entwickelt und setzt auf OpenGL auf [Ste07]. OpenGL ist eine **API (Application Programming Interface)** Bibliothek, die in der Lage ist, 3D-Szenen zu verwalten und darzustellen.

Für die stereoskopische Projektion mittels linearen Polarisationsfiltern von virtuellen Prototypen auf der Grundlage von CAD-Daten ist allerdings das Durchlaufen einer Prozesskette notwendig, der Visualisierungspipeline.

### 2.7.3 Die Visualisierungspipeline

Im Allgemeinen können CAD-Daten auf einem Monitor nur 2-dimesnional dargestellt werden. Das erhebliche Potential bei der stereoskopischen Projektion von CAD-Daten liegt in der Eigenschaft begründet, dass dem Anwender echte 3D Informationen zur Verfügung gestellt werden. Damit wird das Verständnis von komplexen 3D-Strukturen um ein vielfaches erleichtert. Für die stereoskopische Visualisierung von CAD-Daten bedarf es häufig einer manuellen Nach- und Aufbereitung. In einem gemeinsamen Projekt der BMW Group, des Fraunhofer Instituts für Produktionstechnologie IPT, Aachen und des Laboratorium für Werkzeugmaschinen und Betriebslehre WZL, Aachen wurde im Rahmen von Untersuchungen zu Fabrikplanungsprozessen die gesamte Visualisierungspipeline zur stereoskopischen Visualisierung von CAD-Daten durchlaufen [Sch02]. Zur Verdeutlichung der einzelnen Prozessschritte der Visualisierungspipeline sind in **Abbildung 2-22** alle durchzuführenden Aktivitäten dargestellt.

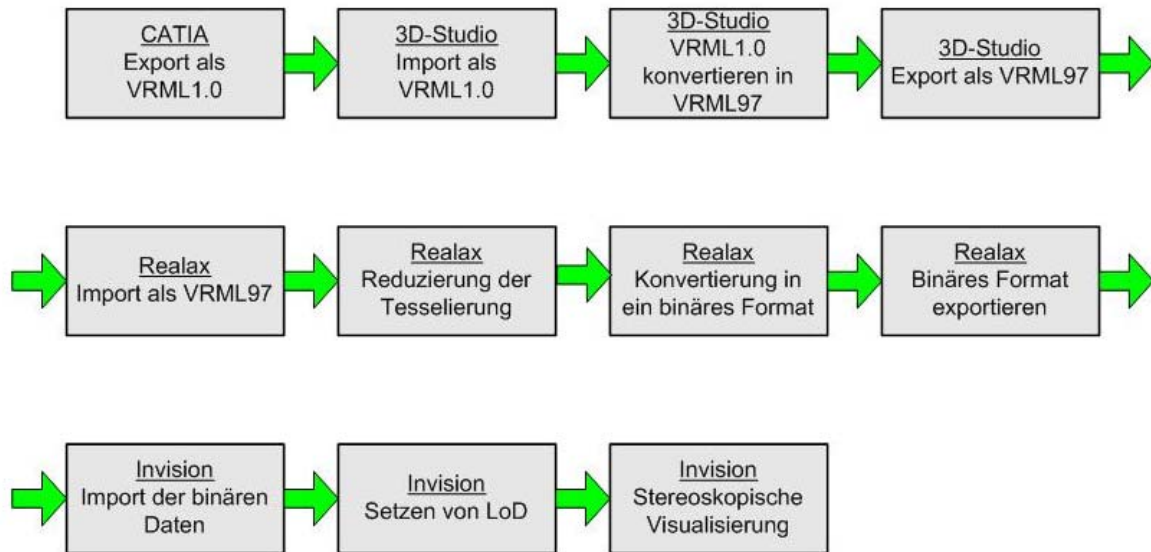


Abbildung 2-22: Die Visualisierungspipeline

Folgende Systeme finden dabei Anwendung:

- CATIA
- 3D-Studio
- Realax
- Invision

Als Grundlage der Projektionsdaten dienen CAD-Daten aus dem CAD-System **CATIA**. Die aus **CATIA** im VRML1.0-Format exportierten Daten werden in das Softwaresystem **3D-Studio** importiert und dort in das Format VRML97, für den Import in **Realax**, umgewandelt. Die vom CAD-System durchgeführte Tesselierung der 3D-Objekte wird nach dem Import der Daten in **Realax** reduziert, nachfolgend in ein binäres Datenformat konvertiert und exportiert. Nach dem anschließenden Import der Binärdaten in **Invision** wird in diesem System die LoD<sup>9</sup>-Technik genutzt, um das 3D-Objekt in definierten Abständen als vereinfachte Geometrie darzustellen. Für die stereoskopische Darstellung von virtuellen Modellen auf der Grundlage von CAD-Daten werden damit vier Softwaresysteme eingesetzt und zehn manuelle Prozessschritte durchlaufen.

<sup>9</sup> LoD - Level of Detail



## 2.8 Fazit

Globaler Wettbewerbsdruck und das Verlangen nach ständig neuen Produktinnovationen bei einer signifikanten Reduzierung der Produktzyklen fordert neue Unternehmensstrategien. Für den Unternehmenserfolg hat sich dabei als wichtigster Kennparameter, die „time-to-market“ herausgestellt. Diese Forderung macht eine Beschleunigung des Produktentwicklungsprozesses notwendig. Zur Optimierung und Verkürzung des Produktentwicklungsprozesses wird immer häufiger die Managementmethode Simultaneous Engineering genutzt. Um dem erhöhten Kommunikationsbedarf bei der Parallelisierung der unterschiedlichen Entwicklungsphasen Rechnung zu tragen, kommen Modelle zum Einsatz. Sie tragen erheblich dazu bei, die Kommunikation zwischen allen am Produktentwicklungsprozess beteiligten Personen zu verbessern. Unterschiedliche Interpretationen von Anforderungen oder Design können damit ausgeschlossen werden und Entwicklungsfehler so vermieden werden.

Als Modelle stehen physische und virtuelle Produktmodelle zur Auswahl. Virtuelle Modelle sind gegenüber physischen Modellen insbesondere dadurch gekennzeichnet, dass sie in digitaler Form vorliegen und jederzeit auch global zur Verfügung stehen können. Zur Herstellung sind keine Werkstoffe nötig und Größenverhältnisse spielen keine Rolle.

Virtuelle Prototypen zeichnen sich dadurch im Besonderen aus, um den standortübergreifenden Produktentwicklungsprozess zu beschleunigen. Virtuelle Modelle können mit Hilfe von Virtual Reality als dreidimensionale Objekte dargestellt werden, um das Verständnis von komplexen Geometrien zu erleichtern. Grundlage zur Erzeugung von virtuellen Modellen ist die digitale Abbildung als Volumenmodell. Hier werden häufig CAD-Systeme als Datenquelle genutzt. Grundsätzlich ermöglichen professionelle CAD-Systeme zwar eine stereoskopische Darstellung der Konstruktionselemente, eine komplette VR-Umgebung wird damit aber noch nicht erzeugt.

In CAD-Systemen wird dies beispielsweise dadurch verhindert, dass nur der Arbeitsbereich, in dem das Modell generiert wird und nicht die umgebenden Menüleisten stereoskopisch dargestellt werden. Um eine Vollbilddarstellung mit einem hohen Immersionsgrad zu erhalten, ist das Ausblenden der Menüleisten aber zwingend notwendig. Auch fehlt oftmals die Unterstützung von immersiven Projektionssystemen. Die beiden Funktionalitäten Multiviewport- und Tracking-Unterstützung sind für hochimmersive Anwendungen jedoch zwingend notwendig. Die bestehenden CAD-Lösungen bieten daher häufig keine Möglichkeit, hochimmersive VR-Umgebungen zu erzeugen.

Für das Erreichen eines hohen Immersionsgrades bei der stereoskopischen Visualisierung, ist die Anschaffung von hochpreisigen Soft- und Hardwarekomponenten erforderlich. Aus diesem Grund wird VR heute in der Regel nur von Großunternehmen, insbesondere der Automotive-Branche, eingesetzt.



Um eine Durchdringung von VR-Technologien in KMU zu erreichen, befinden sich erste Softwarepakete auf dem Markt, die VR auf PC-Basis grundsätzlich ermöglichen. Diese sind häufig nur für spezielle Betriebssysteme einsetzbar. Die Analyse der Softwarearchitekturen hat zudem ergeben, dass auch damit zahlreiche manuelle Prozessschritte zur Erstellung und Optimierung des Datenmodells für die 3D-Visualisierung aus CAD-Daten durchzuführen sind. Damit ist eine sofortige 3D-Visualisierung von CAD-Daten mit einem externen Programm weitestgehend ausgeschlossen. Die Vorbereitung einer VR-Präsentation ist auch aufgrund der Datenvielfalt meist sehr aufwendig und eine schnelle Generierung einer 3D-Präsentation ist in den seltensten Fällen möglich [Hoy06]. Die Prozesskette zur Datengenerierung ist also zu komplex. Neben der Anschaffung der Softwarepakete wird für dessen Einsatz außerdem zusätzlich entsprechend qualifiziertes Bedienpersonal mit Expertenwissen benötigt.

Soll ein Datenaustausch mit Partnern unter der Beibehaltung spezifischer Objekteigenschaften erfolgen, ist dies nur möglich, wenn dort das gleiche CAD-System eingesetzt wird. Simultaneous Engineering verlangt aber oft einen standortübergreifenden Einsatz von Modellen. Somit wird eine Systemkompatibilität zur Grundvoraussetzung für einen fehlerfreien Datenaustausch. Es kann aber nicht davon ausgegangen werden, dass alle Projektbeteiligten und externe Partner stets über die gleichen Softwarekomponenten verfügen. Zwar besteht die Möglichkeit die Daten über systemneutrale Datenformate auszutauschen, hierbei gehen jedoch häufig wichtige Objektparameter für die Nutzung als virtueller Prototyp verloren.

Für die Visualisierung der VR-Modelle sind stereoskopische Sichtsysteme erforderlich. Da für KMU möglichst kostengünstige Lösungen anzustreben sind, kann das passive Flächenprojektionssystem mittels Polarisation genannt werden. Dieses Verfahren ist insbesondere bei einem Multiusereinsatz zu geringeren Kosten zu realisieren als aktive Verfahren. Das Projektionsverfahren mittels Polarisation auf PC-Basis besitzt aber noch erhebliches Optimierungspotential zur Steigerung des Immersionsgrades durch monokulare und binokulare Einflussparameter. Unter monokularen Einflussparametern wird beispielsweise die Farboptimierung einer Baugruppe, um die Beurteilung von Design- oder Anschauungsprototypen zu vereinfachen, oder eine Erhöhung der Tiefenwirkung (3D-Eindruck) um die Auffassung von komplexen Modellgeometrien zu beschleunigen, verstanden. Der Immersionsgrad kann durch binokulare Parameter gesteigert werden, indem Geisterbilder vermieden werden. Hier kann die Methode der Substraktiven-Geisterbild-Reduktion [Kli03] zur Anwendung kommen. Neben der zur Zeit bestehenden Notwendigkeit zum Einsatz von hochpreisigen SGI-Rechnern [Kli03] und die aus diesem Grund bestehende Barriere zum Einsatz bei KMU liegt der Nachteil dieser Methode insbesondere darin begründet, dass eine Reduktion der Crosstalk-Erscheinungen in Regionen mit unterschiedlichen Farben, wie zum Beispiel ein grünes Objekt auf einem roten Hintergrund nicht durchzuführen ist [Fro07].

Die oben aufgezeigten Methoden zur Steigerung des Immersionsgrades mittels mono- und binokularer Einflussparameter finden in den vorhandenen VR-Systemen auf PC-Basis noch keine Berücksichtigung. Außerdem würden diese Methoden einen erheblichen Zeitaufwand erfordern und damit einer Beschleunigung des Produktentwicklungsprozesses entgegenstehen.

Eine Vermeidung von Crosstalk-Erscheinungen könnte mit Polarisationsfiltern erreicht werden, die eine totale Polarisation im gesamten für den Menschen sichtbaren Wellenlängenbereich des Lichtes ermöglichen. Diese Polarisationsfilter stehen zu Zeit nicht zur Verfügung. Geisterbilder erheblich zu reduzieren oder ganz zu eliminieren ist mit den vorhandenen Polarisationsfiltern daher nur bedingt möglich. Geisterbilder verringern aber deutlich den Immersionsgrad.

Aktuelle Forschungsarbeiten wie am Virtual Reality Center der RWTH Aachen mit dem Softwaresystem VISTA [VRC06], am Fraunhofer-Institut für Werkzeugmaschinen und Umformtechnik mit dem Softwaresystem VRax [VRE06] und an der Technischen Universität Chemnitz an der Professur für Werkzeugmaschinenkonstruktion und Umformtechnik mit dem System moVE [Wei07] ermöglichen zwar grundsätzlich eine Kopplung zwischen CAD und VR, aber eine Unabhängigkeit zwischen den Komponenten CAD-System, VR-System, Betriebssystem und einer Hardware auf Single-PC-Basis in Kombination mit einer Integration in den bestehenden Produktentwicklungsprozess unter Beachtung eines standortübergreifenden Einsatzes und der Unterstützung von Multiviewport und Tracking ist bei keiner der vorhandenen Arbeiten gegeben. Konzepte zur Steigerung des Immersionsgrades fehlen ebenfalls.

Aus diesen Defiziten heraus ergeben sich Anforderungen an eine neu zu konzipierende Prozessarchitektur zur Visualisierung von VR-Modellen auf Single-Desktop-Basis, um eine Reduzierung der „time-to-market“ zu erreichen.

### 3 ANFORDERUNGEN AN EINE PC BASIERTE VR-UMGEBUNG

Aufbauend auf den beschriebenen Defiziten, können nun die Anforderungen zur Unterstützung des Produktentwicklungsprozesses formuliert werden. Die Untersuchung der heutigen Produktentwicklungsprozesse hat gezeigt, dass bezüglich der Problemstellung Optimierung der Produktentwicklung unter dem Aspekt einer Reduktion der „time-to-market“ eine erhebliche Beschleunigung des gesamten Entwicklungsprozesses die oberste Priorität darstellt. Wie bereits angedeutet, kann dies mit Hilfe des Simultaneous Engineering durch die Parallelisierung der Entwicklungsphasen erreicht werden.

Simultaneous Engineering erfordert allerdings aufgrund der interdisziplinären Zusammenarbeit einen erhöhten Kommunikationsbedarf. Die Einbindung eines neuen Konzeptes in den Produktentwicklungsprozess, muss die Einbindung aller beteiligten Funktionsgruppen unterstützen. Das zu entwickelnde Konzept muss zudem sicherstellen, dass ein schnelles Verständnis, im Besonderen von dreidimensionalen Objekten, ermöglicht wird. Nur die fehlerfreie Kommunikation und das zügige Verstehen von Sachverhalten und Zusammenhängen versetzen die beteiligten Personen in die Lage, Entscheidungen möglichst schnell zu treffen. Dadurch wird eine erhebliche Beschleunigung des Produktentwicklungsprozesses gewährleistet. Dabei ist zu berücksichtigen, dass ein globaler Einsatz des Konzeptes und die Integration in den bestehenden Produktentwicklungsprozess möglich sein müssen. Um eine möglichst breite Akzeptanz der VR-Technologie zu erreichen, sind einfache kostengünstige Systeme und eine einfache Prozessarchitektur notwendig. Die Anforderungen an eine PC basierte VR-Umgebung lassen sich zusammenfassen:

- Integration in den bestehenden Produktentwicklungsprozess
- 3D-CAD System als Ausgangskomponente
- Weitestgehend Systemunabhängig
- Automatisierter Visualisierungsprozess auf PC-Basis

#### 3.1 Die Integration in den bestehenden Produktentwicklungsprozess

Virtuelle Prototypen können mittels VR helfen, diese Anforderungen zu befriedigen. Der Einsatz der VR-Technologie ist besonders sinnvoll, wenn dadurch neben der Beschleunigung bestehender Prozesse auch Kosten reduziert werden können. Ein Konzept zur Unterstützung des Produktentwicklungsprozesses muss deshalb auch auf die schnelle und kostengünstige Bereitstellung von Virtuellen Prototypen in Stereoprojektion ausgerichtet sein.

Als Forderung wird weiterhin das Erreichen eines hohen Immersionsgrades auf Single-PC-Basis formuliert. Bei der Berücksichtigung dieses Kriteriums richtet sich der Focus auf die KMU, um diese in die Lage zu versetzen, ihren Produktentwicklungsprozess derart zu optimieren, um auch zukünftig wettbewerbsfähig zu bleiben. Bei der Unterstützung des integrativen Charakters des Konzeptes durch die Forderung zur Einbindung in den bestehenden Produktentwicklungsprozess gewinnt die Verwaltung der unterschiedlichen Entwicklungsstände eines Produktes an Bedeutung. Es ist zu gewährleisten, dass eine vollständige Unterstützung des Lebenszyklus des Produktmodells in der Entwicklungsphase erfolgt. Da während der Entwicklung eine ständige Veränderung des Produktmodells statt findet, muss sichergestellt sein, dass auf der einen Seite stets das aktuelle VR-Modell zur Visualisierung ausgewählt wird und auf der anderen Seite keine redundante Datenhaltung entsteht.

Um diesen Anforderungen gerecht zu werden, ist ein System zu konzipieren, das über Schnittstellen an ein bereits bestehendes System, welches in den Entwicklungsprozess integriert ist, angekoppelt werden kann. Aus diesem Grund muss die Konzeption weitestgehend systemunabhängig gestaltet sein. Wenn die Nutzung bereits vorhandener Systeme sichergestellt werden kann, ist eine hohe Mitarbeiterakzeptanz zu erwarten und damit der Erfolg eines neuen Konzeptes eher gegeben.

Als Kernziele für ein neues Konzept können daher neben geringen Investitions- und Unterhaltskosten, eine weitestgehende Systemunabhängigkeit und eine hohe Mitarbeiterakzeptanz genannt werden.

### **3.2 3D-CAD als Kernkomponente zur Erzeugung von VP-Daten**

Die zentrale Anforderung, die sich für die Umsetzung des beschriebenen Ansatzes stellt, ist der Einsatz eines CAD-Systems. Dies bedeutet, dass die Erstellung der virtuellen Prototypen grundsätzlich in einem CAD-System erfolgen muss. Darüber hinaus muss das Konzept allgemeingültig für eine Vielzahl von unterschiedlichen CAD-Systemen anwendbar sein. Die Anforderungen an die Benutzeroberfläche des Anwenders werden als komplette Integration in das CAD-System formuliert. Ein Expertenwissen für die Nutzung des Konzeptes soll damit nicht notwendig sein. Prinzipiell wird gefordert, CAD interne Funktionalitäten für die Gestaltung der virtuellen Prototypen zu verwenden.

### **3.3 Ein allgemeingültige VP-Datenformat**

Eine in den Produktentwicklungsprozess integrierte Lösung muss eine globale Einsetzbarkeit des Konzeptes sicherstellen. Nur wenn es möglich ist, Daten virtueller Prototypen schnell

weltweit unter gleichen Bedingungen darzustellen, ist eine Integration in Simultaneous Engineering zu erwarten. Konkret bedeutet dies, dass ein systemneutrales Datenformat benötigt wird, welches die im CAD-System erzeugten VP-Daten abbildet. Dabei darf weder ein Datenverlust, noch eine Daten- bzw. Qualitätsminderung auftreten. Durch die Anwendung der beschriebenen Forderung soll insbesondere vermieden werden, dass eine Nachbearbeitung von Darstellungsfehlern und andere Änderungen im Rahmen eines Konvertierungsprozesses erfolgen müssen.

Das auszuwählende Datenformat darf weiterhin nicht nur darauf festgelegt sein, von einem CAD-System als Exportformat verwendet zu werden, sondern insbesondere das Merkmal aufweisen, dass es auch von anderen darstellenden Softwaresystemen ohne Datenverlust importiert werden kann. Damit wird ein problemloser Datenaustausch zwischen verschiedenen Systemen sichergestellt.

### **3.4 Darstellung der virtuellen Prototypen**

Zur stereoskopischen Visualisierung der VR-Modelle ist bei dem Konzept eine Architektur zu wählen, die eine prinzipielle Implementierung in jedes CAD-System, aber auch einen Einsatz ohne CAD-System durch den direkten Import von VP-Daten ermöglicht. Damit muss eine Unabhängigkeit von CAD- und Visualisierungssystemen sichergestellt sein. Dabei ist zu beachten, dass die Darstellungen der VP-Modelle mit allgemein zur Verfügung stehenden Hardwarekomponenten erreicht werden kann. Diese Forderung kann in zwei Teilbereiche gegliedert werden. Zum einen in die benötigte Rechnerkapazität und zum anderen in die notwendigen Visualisierungskomponenten.

Die Rechnerkapazität auf Single-PC-Basis ist ein Kriterium dafür, wie eine bestimmte Datengröße des VP-Modells in Echtzeit dargestellt werden kann. Dies bedeutet, dass das entsprechende Modell an die vorhandene Rechnerkapazität angepasst sein muss. Weiterhin ist zu fordern, dass grundsätzlich eine Stereoausgabe der Daten durch die Rechnerarchitektur unterstützt wird.

Die Visualisierungskomponenten müssen so zusammengestellt sein, dass eine große Anwenderzahl zu möglich geringen Kosten eine maximale Darstellungsgüte erhalten. Dabei soll von einer unangemessenen Darstellungsgüte in Form von Photorealismus abgesehen werden. Vielmehr wird der Schwerpunkt auf eine dreidimensionale Visualisierung der VP-Modelle gelegt, um das Verständnis von komplexen Strukturen zu erleichtern. Hier ist eine photorealistische Darstellung häufig nicht notwendig. Insbesondere Ingenieure benötigen oftmals keine Schönheit, sondern Funktionalität.

Zu den wichtigsten Forderungen gehört auch, dass die VP-Modelle in Echtzeit dargestellt und eine Interaktionsfähigkeit in Form von Blickwinkeländerungen durchgeführt werden kann. Um die Flexibilität und Erweiterbarkeit des Konzeptes sicherzustellen, muss

grundsätzlich die Möglichkeit bestehen, eine Multiviewport- und Tracking-Unterstützung für hochimmersive Anwendung einzubinden.

Das Konzept soll weiterhin darauf ausgelegt sein, einen möglichst hohen Immersionsgrad auf der Basis von PC-Komponenten zu erreichen. Diese Forderung wird als Optimierung der mono- und binokularen Einflussparameter formuliert.

Neben den speziellen Forderungen gilt als allgemeine Anforderung, dass die Umsetzung des beschriebenen Ansatzes gewährleistet, dass das Konzept den maximal möglichen Grad an Automatisierung für den Gesamtprozess liefert, ohne dabei die notwendige Flexibilität einzuschränken.

Die Summe der bisher formulierten Forderungen macht die Komplexität der Problemstellung deutlich. Aufbauend auf den Anforderungen werden nun Ansatzpunkte für eine PC basierte VR-Umgebung zur Unterstützung des Produktentwicklungsprozesses vorgestellt.

## 4 ANSATZPUNKTE FÜR EINE PC BASIERTE VR-UMGEBUNG

Nachdem die Anforderungen festgelegt sind, werden nun Ansatzpunkte für ein Gesamtkonzept zur Unterstützung des Produktentwicklungsprozesses aufgestellt, welche die gestellten Anforderungen erfüllen. Das Konzept sieht vor, die „time-to-market“ durch den Einsatz von virtuellen Prototypen in Stereoprojektion (VR) zu reduzieren. Die Ansatzpunkte des Konzeptes gliedern sich grundsätzlich in die Erhöhung des Immersionsgrades von virtuellen Prototypen und in die stereoskopische Visualisierung mit einem weitestgehend systemunabhängigen Visualisierungsmodul. Der Grundbaustein zur Erzeugung der Modelle ist ein CAD-System. Eine konsequente Datenhaltung kann bei der Nutzung eines CAD-Systems vorausgesetzt werden, da diese Systeme bereits in dem übergeordneten Produktentwicklungsprozess integriert sind. Die Verwaltung der unterschiedlichen Entwicklungszustände der CAD-Daten wird heute durch den Einsatz von PDM-Systemen sichergestellt [Lob04]. Bevor die Ausarbeitung des Konzeptes erfolgt, wird die prinzipielle Anwendung des Konzeptes vorgestellt.

Die Architektur des Gesamtkonzeptes sieht vor, dass alle Parametereinstellungen zur Erhöhung des Immersionsgrades im CAD-System vorzunehmen sind. Das für die stereoskopische Visualisierung der CAD-Modelle notwendige Visualisierungsmodul soll in der CAD-Umgebung eingebettet sein. Durch die Aktivierung des Moduls wird der automatisierte Visualisierungsprozess angestoßen. Das Gesamtkonzept stellt damit eine einfache Bedienung und eine hohe Mitarbeiterakzeptanz sicher.

Die Vorgehensweise zur Gestaltung des Gesamtkonzeptes ist wie folgt gegliedert:

Zu Beginn werden Untersuchungen durchgeführt, um festzustellen, durch welche Einflussgrößen ein hoher Immersionsgrad in VR-Anwendungen charakterisiert ist.

Für eine große Systemunabhängigkeit sind die in einem CAD-System vorbereiteten Elemente als virtuelle Prototypen in ein systemneutrales Datenformat zu transformieren. Dazu sind Parameter zu definieren, die eine Beibehaltung der im CAD-System vorgenommenen Parametereinstellungen sicherstellen.

Die Ergebnisse zu den Einflussgrößen auf den Immersionsgrad werden den nutzbaren Transformationsparametern gegenübergestellt. Daraus ergeben sich anschließend Einflussparameter, die zur Steigerung des Immersionsgrades bei der stereoskopischen Darstellung von CAD-Daten weiter zu untersuchen sind. Insbesondere wird untersucht, wie eine Steigerung des Immersionsgrades durch eine Erhöhung der Tiefenwirkung (3D-Eindruck), durch die Zuweisung von Materialeigenschaften und durch eine Reduzierung von Geisterbildern bei der stereoskopischen Visualisierung von VR-Modellen aus CAD-Systemen erreicht und auf Photorealismus verzichtet werden kann. Die Ergebnisse führen zu

Empfehlungen, die es ermöglichen, im CAD-System entsprechende Parametereinstellungen vorzunehmen. Um die Nutzung der Untersuchungsergebnisse in einem Großteil der CAD-Systeme zu gewährleisten, werden diese so allgemeingültig formuliert, dass sie prinzipiell für jedes CAD-System einsetzbar sind.

Für die automatisierte stereoskopische Visualisierung der virtuellen Prototypen wird anschließend ein Softwarealgorithmus für ein Visualisierungsmodul konzipiert, welches zum einen direkt aus dem CAD-System angesprochen wird, zum anderen aber auch eigenständig ohne den Einsatz eines CAD-Systems zu nutzen ist.

Zunächst werden die Einflussgrößen auf den Immersionsgrad näher betrachtet.

#### 4.1 Wahrnehmungsmechanismen

Die Darstellung von real wirkenden Computergrafiken erfordert eine Abbildung der Realität als digitales Modell. Die Vielzahl von komplexen physikalischen Prozessen, die in der realen Welt an der Entstehung von Bildeindrücken beteiligt sind, sind im Rechner nur anhand vereinfachter, mathematischer Modelle nachzubilden. Die Genauigkeit mit der diese analogen Prozesse in der digitalen Umgebung des Computers simuliert werden können, ist für den Grad der Immersion verantwortlich. Programme, welche die Physik von Farbgebung, Beleuchtung und Materialverhalten exakt simulieren können, würden Bildeindrücke im Computer erzeugen, die von der realen Welt nicht mehr zu unterscheiden sind. Dies ist mit heutiger Technik jedoch noch nicht realisierbar.

Wie in den Anforderungen beschrieben, wird der Immersionsgrad in der vorliegenden Arbeit nicht zwingend mit der Erhöhung des Detaillierungsgrades in Form von Photorealismus gesteigert. Das Ziel ist neben einer guten Darstellungsgüte eine optimierte Raumwahrnehmung der visualisierten Objekte zu erzielen, um das Verstehen von komplexen Strukturen zu vereinfachen. Dabei ist der Immersionsgrad zu steigern, ohne den Bedarf an Rechnerkapazität zu erhöhen. Für die Untersuchungen zur Steigerung des Immersionsgrades ist es notwendig, die unterschiedlichen Einflussfaktoren zu kennen. Nachfolgend werden die Einflussparameter auf die Qualität von Computergrafiken erläutert, um anschließend durch eine optimale Abstimmung dieser Parameter den Immersionsgrad von virtuellen Prototypen zu steigern.

Bei totaler Immersion sind alle Bezüge zur realen Welt ausgeschlossen und durch Ersatzstimuli ausgetauscht. Diese Ersatzstimuli können durch computergenerierte Umgebungen erzeugt und sind mit einem oder mehreren Sinn /-en als Realität wahrzunehmen [Hen97]. Der Mensch verfügt grundsätzlich über die folgenden sechs Sinne:

- Vestibulär (Gleichgewichtssinn)
- Gustatorisch (schmecken)



- Olfaktorisch (riechen)
- Haptisch (fühlen)
- Auditiv (hören)
- Visuell (sehen)

In dieser Arbeit liegt der Focus jedoch auf der visuellen Wahrnehmung, da keine Zusatzkomponenten für das Ansprechen weiterer Sinne bei diesem Konzept vorgesehen sind. Es gilt also den Immersionsgrad durch die visuelle Wahrnehmung des Nutzers zu erhöhen. Folgende visueller Parameter sind dabei zu berücksichtigen:

- Bildauflösung
- Bildrate
- Interaktionsfähigkeit
- Natürliches Sichtfeld
- Darstellung (Projektionswand / CAVE)
- Bildkomplexität
- Raumwahrnehmung

Die **Bildauflösung** hat einen bedingten Einfluss auf die Darstellungsgüte. Bei einer geringen Bildauflösung ist zwar kein Photorealismus zu erzielen, eine immersive Darstellung ist jedoch durchaus möglich. Da heutige Grafikkarten in der Lage sind, Szenen mit sehr hohen Auflösungen darzustellen, ist die Darstellungsgüte in hochpreisigen VR-Systemen häufig nur durch die Bildauflösung der Projektoren begrenzt.

Neben der Bildauflösung kommt der **Bildrate** eine besondere Bedeutung zu. Bei einer Bildwiederholfrequenz ab 50 Hz empfindet der Mensch eine Bewegung als ruckel- und flimmerfrei und damit als real. Für eine in Echtzeit bewegte VR-Präsentation muss aufgrund der Visualisierungskomponenten sichergestellt sein, dass die Bildrate von 50 Hz nicht unterschritten wird.

Die **Interaktionsfähigkeit** ist mit der Bildrate eng verknüpft. Da sämtliche Bildberechnungen zeitkritisch sind, spielt die Gesamtverzögerungszeit (Latenzzeit) eine wichtige Rolle [Hen97]. Um den Realitätseindruck nicht zu zerstören, müssen Verzögerungen unter 100 ms liegen um sie nicht mehr wahrzunehmen [Hen97].

Für einen sehr hohen Immersionsgrad muss eine Abdeckung des **natürlichen Sichtfeldes** des Menschen von ca. 170° horizontal und ca. 110° vertikal durch die Visualisierung erfolgen. Dies ist durch die **Darstellung** auf großen Projektionsflächen zu erreichen.

Eine **hohe Bildkomplexität** kann ebenfalls einen Beitrag zu der Steigerung des Immersionsgrades leisten. Neben der Komplexität der dargestellten Szene ist die Detailtreue der einzelnen Elemente zudem von Bedeutung.

Als wichtigstes Werkzeug zur visuellen Steigerung des Immersionsgrades ist das stereoskopische Sehen und die damit verbundene **Raumwahrnehmung** zu nennen. Erst das Aufnehmen und Verstehen von räumlichen Parametern durch das visuelle System des Menschen kann einen Ersatz der realen Welt ermöglichen.

Von allen genannten Parametern ist die Raumwahrnehmung von besonderem Interesse, da sie das Erfassen einer virtuellen Umgebung erst ermöglicht. Insbesondere ist eine Analyse, welche Mechanismen zu der Wahrnehmung räumlicher Tiefe führen, wichtig, da die räumliche Tiefe den Umgang mit der Umgebung in besonderer Weise erleichtert. Die Literatur liefert eine Vielzahl von Parametern, die dem Menschen für die Tiefenwahrnehmung zur Verfügung stehen [Cut95].

Diese Parameter setzen sich aus der Wahrnehmung verschiedener Eigenschaften der Objekte zusammen: der Position (Entfernung und Richtung), ihrer Größe, Form und Orientierung, sowie ihren Oberflächeneigenschaften (Farbe, Oberflächentextur) und weiteren Materialeigenschaften wie Transluzenz (partielle Lichtdurchlässigkeit eines Objektes), Eigenstrahlung etc. Zu einer räumlichen Wahrnehmung können alle genannten Eigenschaften beitragen. Es wird dabei aber grundsätzlich in zwei Raumwahrnehmungsmechanismen unterschieden [Hof02]:

- Egozentrische Tiefe (EgT)
- Exozentrische Tiefe (ExT)

Unter der **egozentrischen Tiefe** wird die wahrgenommene absolute Entfernung der Teilobjekte des Raumes vom Betrachter und unter der **exozentrischen Tiefe** die wahrgenommene Entfernung verschiedener Teilobjekte des Raumes voneinander in Blickrichtung verstanden. Um das Verständnis über die unterschiedlichen Parameter und Mechanismen zu schaffen, die zur Wahrnehmung der räumlichen Tiefe führen, wird nachfolgend ein systematischer Überblick und eine Erklärung zu den wichtigsten Parametern gegeben (vgl. **Abbildung 4-1**). Wie bereits im Abschnitt 2.8 erläutert, wird in diesem Zusammenhang unterschieden zwischen monokularen und binokularen Einflussparametern.

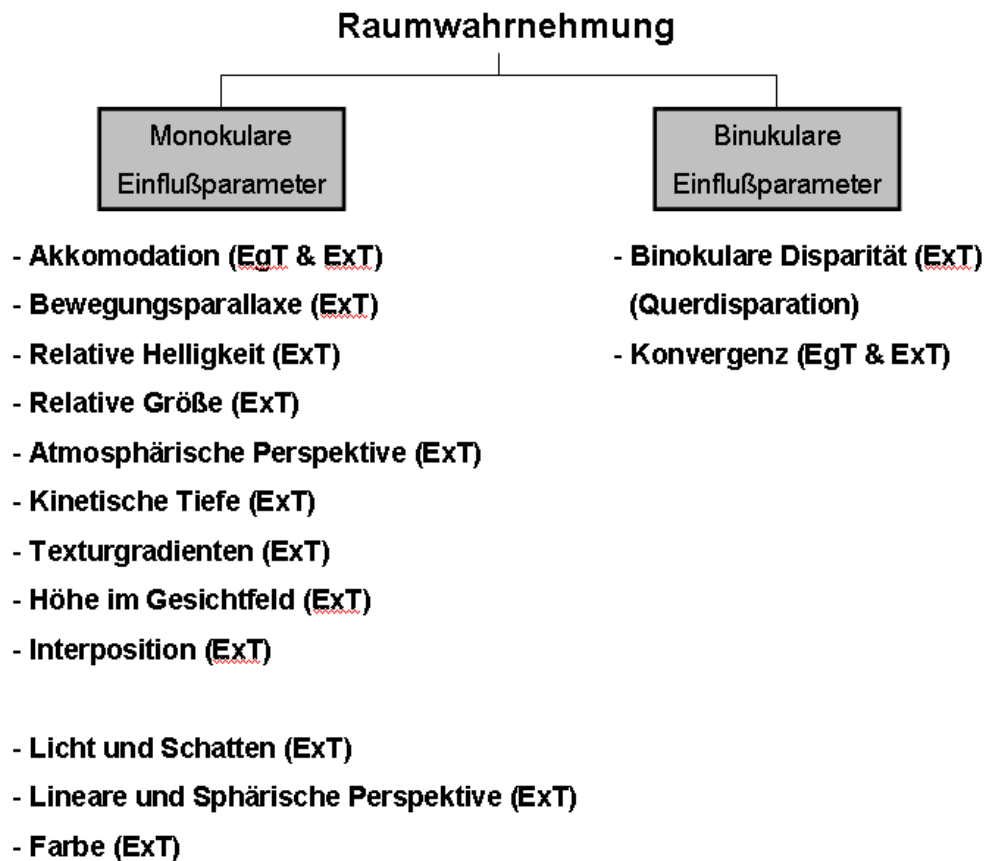


Abbildung 4-1: Parameter der Raumwahrnehmung

Die **binokulare Disparität** wird auch Querdisparation oder Deviation genannt. Bei der binokularen Disparität werden zwei von einander entfernte Objekte auf unterschiedlichen Bereichen der Netzhäute der beiden Augen projiziert. Der Differenzwinkel der Lote von den Objekten auf die Netzhaut wird Querdisparation genannt. Damit stellt sie Informationen über exozentrische Tiefe zur Verfügung, nicht aber über absolute Entfernung vom Betrachter. Mit Hilfe der Querdisparation ist ein Rückschluss auf die exozentrische Tiefe zwischen zwei Objekten möglich.

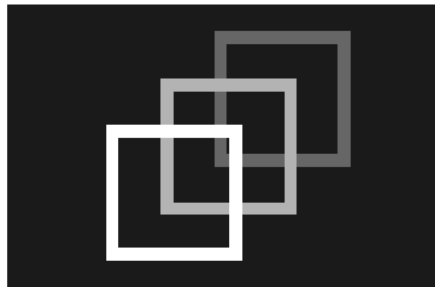
Mit der **Konvergenz**, d.h. mit dem Einwärts- oder Auswärtsdrehen der beiden Augen, wird zwischen Objekten verschiedener Entfernungen gewechselt. Die Konvergenz liefert eine absolute Entfernungsinformation.

Durch die **Akkommodation** wird die Brechkraft der Augenlinsen verändert. Die Unschärfe von entfernten Gegenständen wird durch die Einstellung der Brechkraft mit Hilfe von Kontraktion oder Expansion der Linse auf der Netzhaut scharf abgebildet. Mit der Akkommodation kann eine Bestimmung der absoluten Entfernung erfolgen.

Ändert der Beobachter seine Position, wird auf das Verhältnis der absoluten Entfernungen zweier Objekte rückgeschlossen. Dieser Mechanismus wird als **Bewegungsparallaxe** bezeichnet. Ein in der Nähe stehender Baum ändert scheinbar schneller seine Position, wie

der dahinter liegender Bergrücken. Die absolute Entfernung der Objekte ist mit der Bewegungsparallaxe aber nicht zu bestimmen.

Entfernt sich ein Objekt vom Betrachter, so verringert sich seine durch Reflexion erzeugte Leuchtdichte (Intensität) mit dem Abstand (vgl. **Abbildung 4-2**). Aufgrund der **relativen Helligkeit** (engl. Depth Cueing) kann ein Rückschluss auf die Veränderung der Entfernung erfolgen.



*Abbildung 4-2: Relative Helligkeit [Ste97]*

Näher angeordnete Objekte erscheinen größer als entfernt liegende Objekte. Das Größenverhältnis von gleich großen Objekten, die aber eine unterschiedliche Distanz zum Betrachter aufweisen, wird als **relative Größe** bezeichnet (vgl. **Abbildung 4-3**). Sind die wahren Abmessungen eines Gegenstandes bekannt, wird mit Hilfe der Größe dieses Gegenstandes auf die Entfernung eine Rückfolgerung möglich.



*Abbildung 4-3: Relative Größe*

Durch die in der Luft schwebenden Teilchen, wird die Atmosphäre mit zunehmender Entfernung trüber und damit kontrastärmer (vgl. **Abbildung 4-4**). Die Sichtbarkeit von Objekten nimmt aufgrund dieser **atmosphärischen Perspektive** mit der Entfernung vom Betrachter ab.

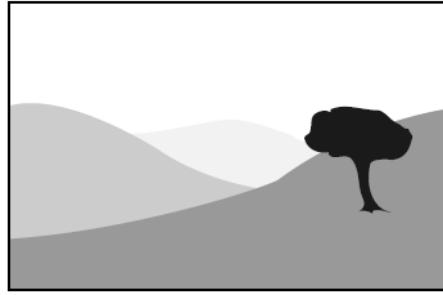


Abbildung 4-4: Atmosphärische Perspektive [Ste97]

Bewegte Objekte können aufgrund der kontinuierlichen Veränderung der Position einem statischen Beobachter Informationen über die Tiefe liefern. Diese Tiefeninformation wird **kinetische Tiefe** genannt. Dieser Tiefeneffekt tritt beispielsweise auf, wenn sich ein Objekt horizontal bewegt. Näher liegende Gegenstände wandern schneller an dem Objekt vorbei als weiter entfernt liegende Gegenstände.

Texturen können genutzt werden, um über fehlende Detailtreue hinwegzutäuschen. Durch den Einsatz von Texturen ist es möglich, aus der einfachen Oberflächengeometrie einer Kugel den Planeten Erde zu generieren (vgl. **Abbildung 4-5**). Mit Texturen ist also ein einfaches Objekt durch Aufbringen eines Bildes auf die Objektoberfläche wirkungsvoll in Szene zu setzen.

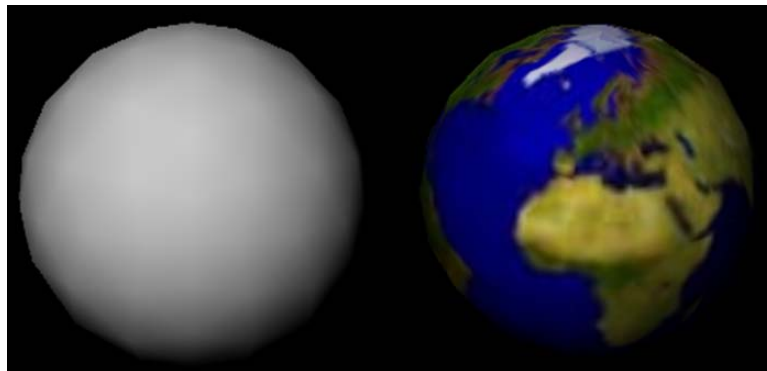
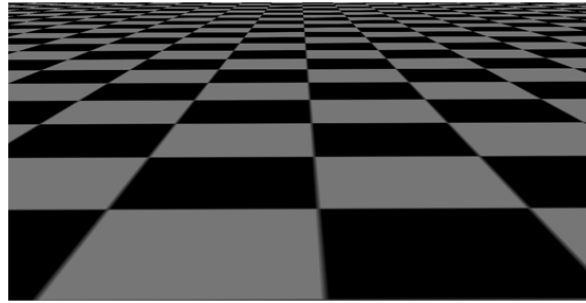


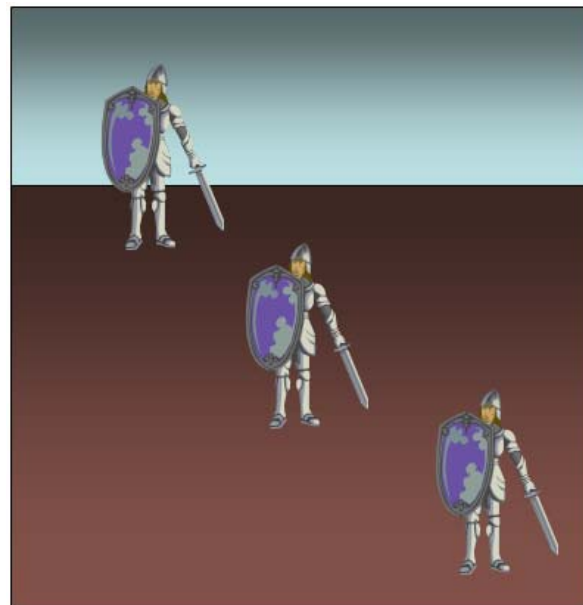
Abbildung 4-5: Textur

Die gleichmäßige Veränderung von Texturen stellt eine wichtige Information der Tiefe dar. Diese gleichmäßigen Texturveränderungen werden als **Texturgradienten** bezeichnet. Durch das Ansteigen des Dichtegradienten der Textur Elemente wird eine Tiefeninformation extrahiert (vgl. **Abbildung 4-6**).



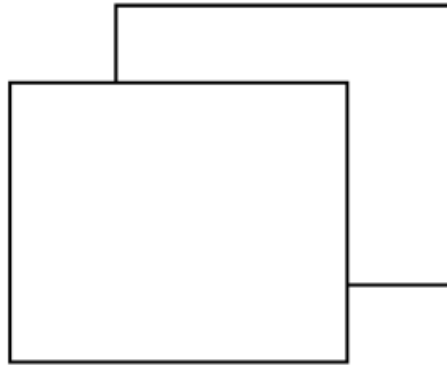
*Abbildung 4-6: Texturgradient*

Eine weitere Tiefeninformation wird über die **Höhe im Gesichtsfeld** geliefert. Objekte, die sich näher am Horizont befinden (vgl. **Abbildung 4-7**) werden als größer und weiter entfernt wahrgenommen, als Objekte, die nicht so nah am Horizont positioniert sind und die gleiche Größe aufweisen. Die Figur am Horizont wirkt größer als die Figur in der rechten unteren Bildecke der Abbildung 4-7.



*Abbildung 4-7: Höhe im Gesichtsfeld*

Eine eindeutige Information über die Entfernung zweier Objekte ist aus der **Interposition** (Verdeckung) zu gewinnen, d.h., wenn ein Objekt von einem anderen verdeckt wird (vgl. **Abbildung 4-8**). Dieser Mechanismus liefert jedoch nur die Information, welches Objekt vom Betrachter weiter entfernt ist. Auf Grund der Verdeckung wird keine Aussage über die Größe des Abstandes beider Objekte getroffen.



*Abbildung 4-8: Interposition*

Bei der Erstellung von virtuellen Prototypen haben das **Licht** und der **Schatten** eine besondere Bedeutung. Die Art der Lichtquellen und ihre Positionierung besitzen erheblichen Einfluss, sowohl auf die Wirkung der Farben, als auch auf die Oberflächenstruktur. Nur bei optimaler Beleuchtung und naturgetreuer Schattierung und Schattenwurf (vgl. **Abbildung 4-9**) wirken die gewählten Farben und die erstellte Oberflächenstruktur real.



*Abbildung 4-9: Licht und Schatten [Fol94]*

Die **lineare Perspektive** ist eines der wichtigsten Effekte zur Erzeugung einer Tiefeninformation. Sie wird durch den Verlauf von parallelen Linien zu einem Fluchtpunkt erzielt (vgl. **Abbildung 4-10**). Darstellungen mit einer deutlichen Perspektive erzeugen auch einen starken Tiefeneffekt.

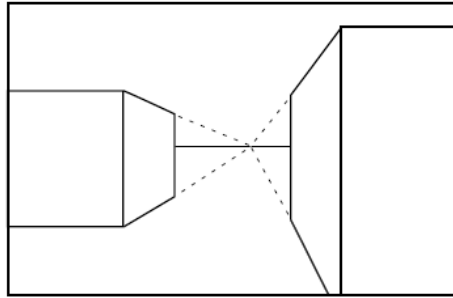


Abbildung 4-10: Lineare Perspektive [Ste97]

Neben der linearen Perspektive ist auch die **sphärische Perspektive** (vgl. **Abbildung 4-11**) zu nennen. Während die lineare Perspektive Tiefeninformation in einer Ebene liefert, wird mit der sphärischen Perspektive eine räumliche Tiefeninformation gewonnen.

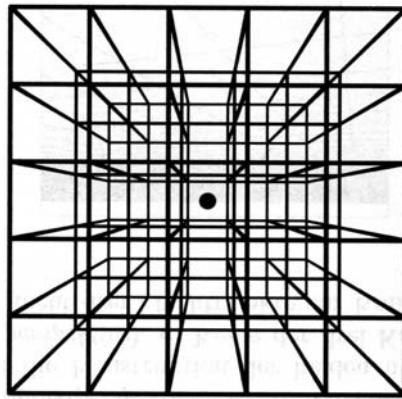


Abbildung 4-11: Sphärische Perspektive [Mal00]

Dargestellt auf einem Hintergrund anderer Helligkeit scheinen farbige Flächen zu schweben [Gra86]. Wie in **Abbildung 4-12** zu sehen ist, scheint der gelbe Würfel gegenüber dem schwarzen Hintergrund zu schweben, diesen Effekt weist der graue Würfel in einem wesentlich geringeren Maße auf.



Abbildung 4-12: Farbiger Tiefeneffekt



Die **Farbe** ist also ein wichtiger Parameter, um die egozentrische Tiefenwirkung zu beeinflussen. Schumann und Müller [Sch00] nennen diesen Wahrnehmungsmechanismus kinetischen Effekt. Da diese Benennung leicht mit der kinetischen Tiefe verwechselt werden kann, wird in dieser Arbeit die Bezeichnung farbiger Tiefeneffekt benutzt, um eine deutliche Trennung zwischen den beiden Effekten zu erreichen.

Nachdem ein grundlegendes Verständnis über die unterschiedlichen Wahrnehmungsmechanismen geschaffen wurde, soll nun untersucht werden, welche Darstellungsparameter bei einer Konvertierung von CAD-Daten in ein systemneutrales Datenformat zu nutzen sind. Da für die nachfolgenden Untersuchungen ein manueller Datenaufbereitungs- und Visualisierungsprozess durchlaufen werden muss, wird dieser im nächsten Abschnitt zuvor erläutert.

## 4.2 Der manuelle Visualisierungsprozess

Für die nachfolgenden Untersuchungen sind CAD-Daten in einem allgemeingültigen Datenformat zu visualisieren. Grundlage des Visualisierungsprozesses ist ein CAD-System. Da die Erstellung einer CAD-Geometrie eines Prototypen nicht Gegenstand des Visualisierungsprozesses ist, wird von einem vorhandenen CAD-Modell ausgegangen. Bei dem Datenaufbereitungs- und Visualisierungsprozess kommen beispielhaft die Softwaresysteme SolidWorks, Cosmo Worlds und VRED (vgl. Abbildung 4-13) zum Einsatz, die nun kurz erläutert werden.

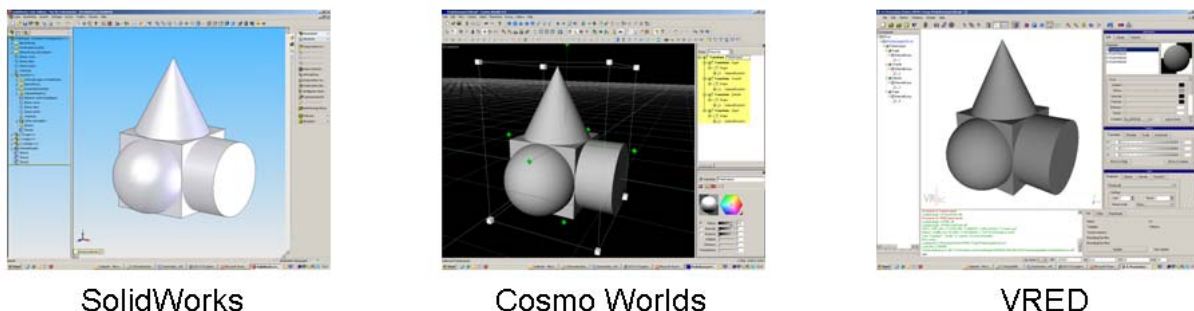


Abbildung 4-13: Verwendete Softwaresysteme

Wie bereits im Abschnitt 2.5 dargelegt, wird das Modell im CAD-System gestaltet. Das CAD-System stellt über Schnittstellen die Funktionalität der Datenkonvertierung in andere Datenformate zur Verfügung. Die aus dem CAD-System SolidWorks exportierten Daten werden in Cosmo Worlds importiert und aufbereitet. Cosmo Worlds ist ein VRML-Autorensystem, welches die Positionierung des Modells, die Definition sämtlicher Materialeigenschaften und das Verknüpfen mit anderen VRML-Daten (vgl. Abschnitt 4.3.2)

ermöglicht. Durch die Verwendung dieses *VRML*-Editors wird die effektive Erstellung von virtuellen Umgebungen erleichtert, da das Softwaresystem über eine übersichtliche, grafische Benutzeroberfläche verfügt. Damit ist es dem Anwender möglich, die komplette Modellierung durchzuführen ohne Programmbefehle manuell eingeben zu müssen. Der *VRML*-Programmcode wird dabei automatisch im Hintergrund erstellt. Nach der Aufbereitung der Daten werden diese wiederum als *VRML*-Datei gespeichert und in *VRED* importiert.

Neben der Darstellung von Szenen als Einzelbilder wird mit dem VR-System *VRED* die stereoskopische Visualisierung der Szenen durchgeführt. *VRED* wird ebenfalls als Autorensystem verwendet. Es hat jedoch gegenüber *Cosmo Worlds* einen sehr eingeschränkten Funktionsumfang, da der Schwerpunkt des Systems auf der stereoskopischen Darstellung von 3D-Szenen liegt. Die manuellen Prozessschritte für den softwareseitigen Visualisierungsprozess lassen sich wie folgt untergliedern:

- Transformation des CAD-Modells in das *VRML*-Format
- Import des *VRML*-Modells in *Cosmo Worlds*
- Optimierung der Darstellungsparameter in *Cosmo Worlds*
- Export des optimierten *VRML*-Modells in eine *VRML*-Datei
- Import des *VRML*-Modells in *VRED*

Die Bedienung der Softwareprogramme soll hier nicht weiter konkretisiert werden. Falls dies notwendig sein sollte, wird an den entsprechenden Stellen darauf jedoch kurz eingegangen. Nach dem Import des *VRML*-Modells in *VRED* erfolgt die stereoskopische Visualisierung (vgl. Abschnitt 2.7 ) der 3D-Szene hardwareseitig mit den nachfolgend aufgelisteten Komponenten:

- 1 Projektionsscheibe mit Fresnel-Profil 2460mm x 1840mm
- 2 Projektoren Sharp XG-NV7XE mit Polarisationsfiltern
- 1 Rechner Intel P4 3,2 GHz, 2 GB Ram, Nvidia Quadro FX1400
- Polarisationsbrillen für die jeweilige Anzahl der User

Vor den Parameteruntersuchungen zur Erhöhung des Immersionsgrades muss zwingend festgestellt werden, welche Parameter bei einem Transformationsvorgang von CAD-Daten in ein neutrales Datenformat beizubehalten sind. Aus diesem Grund wird nachfolgend auf die möglichen Parametereinstellungen in einem CAD-System und anschließend auf die

nutzbaren Parameter des systemneutralen Datenformates näher eingegangen. Danach werden die Transformations- und Exportparameter, ausgehend von den CAD-Daten zu VRML-Daten hinsichtlich der verwendbaren Parameter untersucht. Erst im Anschluss daran, können die Untersuchungen zur Steigerung des Immersionsgrades folgen.

### 4.3 Transformationsparameter

In heutigen CAD-Systemen ist es möglich, eine Vielzahl von Darstellungsparametern zu variieren. Neben dem Erstellen von Beleuchtungsszenen mit unterschiedlichen Beleuchtungseinrichtungen können mit Texturen und individuellen Materialeinstellungen photorealistische Szenen abgebildet werden. Materialeigenschaften (vgl. **Abbildung 4-14**) unterscheiden sich beispielsweise in der Art der Lichtreflexion. Materialien, welche Licht in alle Raumrichtungen gleichmäßig reflektieren, wirken stumpf (z.B. Papierhandtuch).

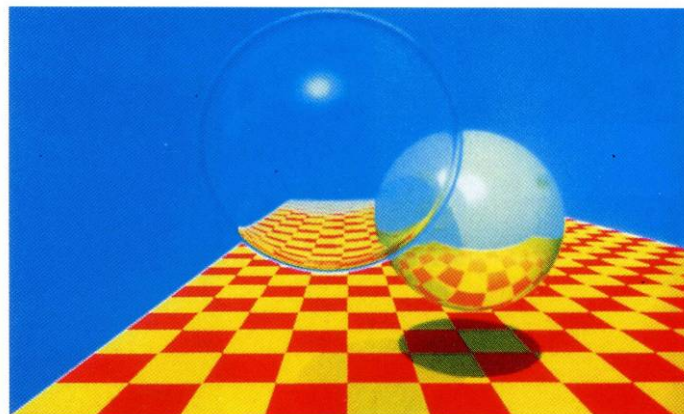


Abbildung 4-14: Materialeigenschaften [Fol94]

Andere hingegen glänzen und reflektieren Licht einer Lichtquelle nur in bestimmte Richtungen (polierte Metallkugel). Zu den Materialeigenschaften eines Objektes gehört auch die Transparenz. Der Transparenzgrad kann dabei variieren zwischen volltransparent (z.B. Glaskugel) teiltransparent oder nicht transparent (z.B. Metallkugel).

Soll nun ein im CAD-System erstelltes Objekt mit zugewiesenen Lichtquellen, Texturen und Materialeigenschaften in ein anderes Dateiformat transformiert werden, ist zu klären, welche Parameter durch die Transformation und den anschließenden Export in das neue Datenformat übernommen werden. Für die Transformation und den Export in ein anderes Datenformat stehen in CAD-Systemen häufig verschiedene Formate zur Auswahl. In dieser Arbeit wird das Format VRML (vgl. Abschnitt 4.3.2) verwendet. Zunächst wird auf die im CAD-System SolidWorks zur Verfügung stehenden Einflussparameter zur Immersionsgradsteigerung näher eingegangen. Daran schließt ein kurzer Einblick in die

Modellierungssprache für 3D-Objekte *VRML* und dessen mögliche Materialeinstellungen an. Die Parameter des CAD-Systems werden anschließend den *VRML*-Parametern gegenübergestellt, um die nutzbaren Parameter für eine Transformation zu definieren.

#### 4.3.1 CAD-Parameter

Da am Lehrstuhl für Produktionstechnologie und Produktentwicklung bereits weitgehende Kenntnisse des CAD-Systems *SolidWorks* (SW) existieren, und da dieses System zu den führenden CAD-Systemen im Bereich der mechanischen Konstruktion gehört, wird dieses System für den Einsatz zur Konzepterstellung ausgewählt. SW zeichnet sich im besonderen Maße sowohl durch seine intuitive Benutzeroberfläche, die eine einfache Erlernbarkeit ermöglicht, als auch durch weitere Softwarekomponenten für die Bereiche Fertigung und Simulation, die innerhalb von *SolidWorks* eingesetzt werden können, aus. Am Beispiel des CAD-Systems *SolidWorks* der Version 2007 wird die Zuweisung von entsprechenden Parametern näher erläutert.

Der Immersionsgrad ist hier grundsätzlich durch die Zuweisung von Materialeigenschaften wie Glanz, Spiegelung oder Transparenz (vgl. **Abbildung 4-15**) schon in der frühen Phase der Bauteil- oder Baugruppenerstellung im CAD-System zu steigern und kann damit in der Baugruppenansicht eine Unterscheidung von verschiedenen Werkstoffeigenschaften ermöglichen.

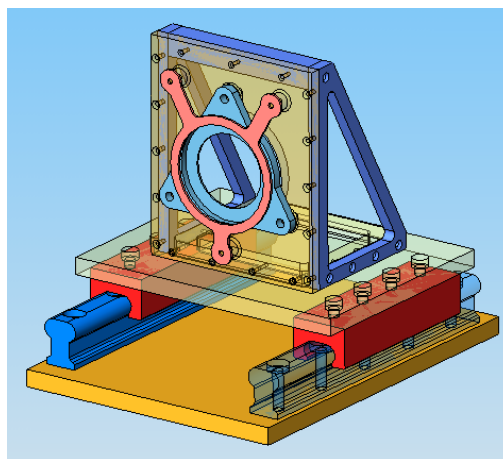


Abbildung 4-15: Baugruppe mit unterschiedlichen Materialeigenschaften [Sol07b]

Die Materialeigenschaften sind in *SolidWorks* getrennt nach Farben und optischen Eigenschaften. Die Farbeinstellungen (vgl. **Abbildung 4-16**) von Elementen können im RGB- oder im HSV-Farbsystem verändert werden. Der Wertebereich zur Änderung der

Farbparameter reicht für das RGB-System (vgl. Abschnitt 4.4.1) von 0 bis 255 und für das HSV-System (vgl. Abschnitt 4.4.1) von 0 bis 360.

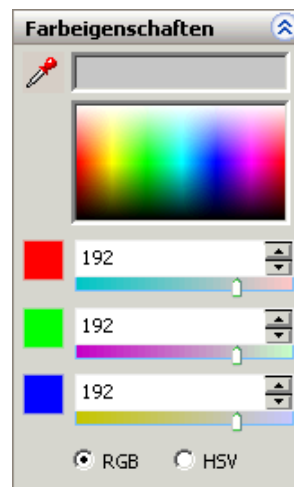


Abbildung 4-16: Eingabemaske für die Farbeinstellung in SW2007

Die optischen Eigenschaften sind in fünf Parameter untergliedert. Diese Parameter können die Werte von 0 bis 1 annehmen. In **Abbildung 4-17** ist die Eingabemaske für die optischen Eigenschaften zu erkennen.



Abbildung 4-17: Eingabemaske für die optischen Eigenschaften in SW2007

Insbesondere mit den optischen Eigenschaften (Materialeigenschaften) wird ein realer Eindruck der CAD-Objekte erreicht. Denn erst durch die Zuweisung von Materialeigenschaften erscheinen die Objekte echt. Die optischen Eigenschaften (vgl. Abbildung 4-17) werden nachfolgend erläutert [Sol07a]:

- **Transparenz:** Steuert den Grad, in dem Licht durch das Modell durchtritt.
- **Umgebungslicht:** Nähert die Lichtintensität einer Lichtquelle, die eine Oberfläche von allen Seiten ohne Attenuation (Dämpfung, Verminderung) oder Schattierung beleuchtet, an. Dabei wird unter Attenuation die Verringerung der Lichtintensität mit Zunahme des Abstands verstanden.
- **Diffusion.** Steuert die Lichtintensität auf der Oberfläche. Diffusion hängt von seinem Winkel zu einer Lichtquelle ab, ist aber unabhängig vom Blickwinkel des Betrachters.
- **Spiegeleffekt.** Steuert die Lichtintensität auf der Oberfläche. Der Spiegeleffekt hängt von der Position der Lichtquelle und des Betrachters ab.
- **Glanz.** Steuert die Reflexionskraft des Lichtes.
- **Emissionsgrad.** Steuert die Intensität des abgestrahlten Lichtes.

Neben den optischen Eigenschaften ist es auch möglich, den Objekten unterschiedliche Lichtquellen zu zuweisen. Die Lichtquellen lassen sich unterteilen in [Sol07a]:

- **Umgebungslicht:** Leuchtet das Modell gleichmäßig aus allen Richtungen aus. In einem Raum mit weißen Wänden gibt es viel Umgebungslicht, da das Licht von den Wänden und anderen Objekten reflektiert wird.
- **Gerichtetes Licht:** Kommt von einer Quelle, die unendlich weit vom Modell entfernt ist. In diesem Fall besteht das Licht aus parallelen Strahlen, die aus einer einzigen Richtung, wie etwa von der Sonne, kommen.
- **Punktlicht:** Kommt von einer sehr kleinen Lichtquelle, die sich an einer bestimmten Koordinate im Modellraum befindet. Diese Art von Lichtquelle emittiert Licht in alle Richtungen. Die Auswirkungen sind vergleichbar mit denen einer winzigen Glühbirne, die im Raum schwebt.
- **Scheinwerferlicht:** Kommt von einem begrenzten und gebündelten Licht in einem kegelförmigen Strahl, der in der Mitte am hellsten ist. Ein Scheinwerferlicht kann auf einen bestimmten Bereich des Modells gerichtet sein. So kann die Position und der Abstand der Lichtquelle relativ zum Modell sowie der Winkel, in dem sich der Strahl ausbreitet, angepasst werden.

Mit den unterschiedlichen Lichtquellen ist es möglich, die verschiedenen Objekte im CAD-System noch besser in Szene zu setzen und damit eine Steigerung des Realitätseindrucks zu gewinnen.

Neben der Auswahl von unterschiedlichen Lichtquellen können Texturen ebenfalls einen Beitrag zur Erhöhung des Realitätseindrucks leisten. Für die Auswahl an Texturen bietet *SolidWorks* eine große Datenbasis unterschiedlicher Texturen an. Neben Metallen und Kunststoffen, sind auch außergewöhnliche Texturen zu wählen wie beispielsweise eine Korbstruktur (vgl. **Abbildung 4-18**).

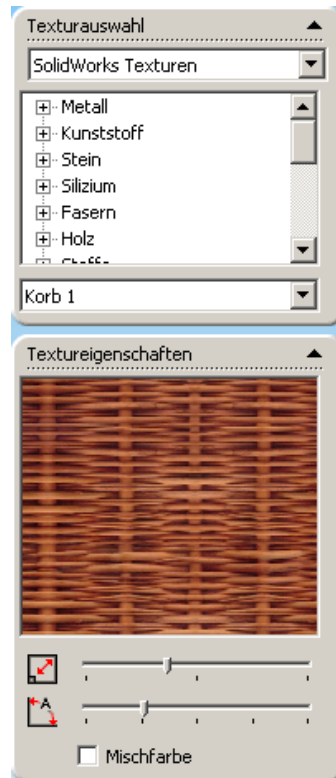


Abbildung 4-18: Eingabemaske für Texturen in SW 2007

Für den Export der CAD-Objekte stehen unterschiedliche Formate zur Auswahl. In diesem Konzept wird das *VRML*-Format (vgl. Abschnitt 4.3.2) verwendet. Bei der Transformation der CAD-Daten in das *VRML*-Format werden alle CAD-Objekte tesselliert. D.h. es erfolgt eine Annäherung der Objekte durch Polygone an die wirkliche Oberflächengeometrie (vgl. Abschnitt 2.5). Die Genauigkeit dieser Annäherung wird durch den Sehnenfehler angegeben. In *SolidWorks* hängt die Sehnenabweichung von der aktuellen Anzeigenauflösung ab. Soll die Anzeigenauflösung variiert werden, muss eine Änderung der gebotenen Bildqualität erfolgen. Die Anzeigenauflösung wird mit der in **Abbildung 4-19** dargestellten Eingabemaske eingestellt.

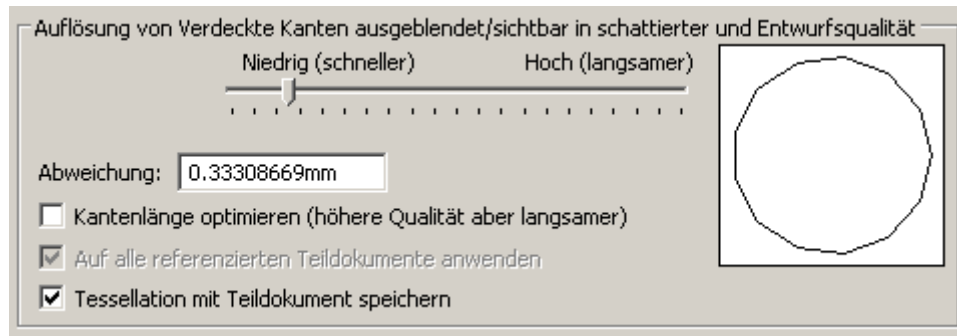


Abbildung 4-19: Eingabemaske für den Tessellierungsgrad in SW 2007

Mit dem Schieberegler wird die Einstellung für die maximale Sehnenabweichung vorgenommen. Der Wert wird im Feld *Abweichung* angezeigt. Es besteht zusätzlich die Möglichkeit, einen exakten Zahlenwert für die Sehnenabweichung in diesem Feld manuell einzugeben. Je höher die Auflösung gewählt wird, umso geringer ist der Sehnenfehler. Je geringer der Sehnenfehler gewählt wird, umso exakter wird die Polygonstruktur bei der Transformation der CAD-Objekte in das *VRML*-Format angenähert. Sollte der Maximalwert der Auflösung nicht ausreichen, wird zusätzlich das Feld *Kantenlänge optimieren* aktiviert. Damit wird eine noch höhere Bildqualität erreicht.

Bei Baugruppen ist es möglich, diese Einstellungen auf alle referenzierten Teile zu übernehmen. Zusätzlich kann die Tessellationsinformation mit dem Teildokument gespeichert werden.

Nachdem nun die unterschiedlichen Einstellungsparameter des CAD-Systems vorgestellt wurden, wird im nächsten Abschnitt auf das neutrale Datenformat zum Export der CAD-Modelle eingegangen.

### 4.3.2 Das systemneutrale Datenformat

Aus dem Bedürfnis heraus, ein Softwaresystem einzusetzen, welches in der Lage ist, die Visualisierung von virtuellen Welten mit Hilfe von dreidimensionalen Objekten zu realisieren, ist *VRML* entwickelt worden. Mit dieser Beschreibungssprache, auch Skriptsprache genannt, hat sich der Graphikstandard *VRML* ISO/IEC 14772 etabliert. Mit der Erklärung zum ISO Standard ist *VRML* zu einer weltweit akzeptierten Basis für Softwareentwicklungen geworden. Dabei waren die Unabhängigkeit des verwendeten Computersystems, eine beliebige Erweiterbarkeit und eine Nutzung auch bei geringen Übertragungsgeschwindigkeiten im Internet die Grundanforderungen an die neue Beschreibungssprache *VRML*.

Mit *VRML* besteht die Möglichkeit, statische und animierte Objekte in interaktiven 3D-Welten zu modellieren und durch Verknüpfungen mit anderen Medien wie Sound oder Video zu



verbinden. Die Erstellung des *VRML*-Codes kann mittels Autorenwerkzeuge wie *Cosmo Worlds* komfortabel in einer Softwareumgebung oder durch eine zeilenweise Programmierung per Hand erfolgen. Die Sicherung des *VRML*-Codes wird binär oder als ASCII-Datei durchgeführt. Für die Darstellung der *VRML*-Szenen werden Browser wie dem Internet Explorer oder Netcape mit entsprechenden Plug-Ins benötigt. Autorenwerkzeuge und Browser sind mittlerweile für viele Plattformen verfügbar.

Eine *VRML*-Szene besteht aus einer hierarchischen Anordnung der aufgelisteten Objekte. Die Hierarchie legt die Beziehung der Objekte untereinander fest und wird Szenengraph genannt. Die Objekte werden in *VRML* als Knoten bezeichnet und beinhalten neben der geometrischen Beschreibung der Körper u. a. die Definition von Materialeigenschaften. Die Struktur eines *VRML*-Codes wird an einem kurzen Beispiel erläutert. Dabei wird als Körper ein Würfel (vgl. **Abbildung 4-20**) mit definierten Materialeinstellungen verwendet.

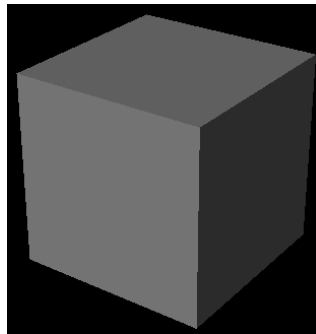


Abbildung 4-20: Würfel

Der *VRML*-Code des Würfels ist in **Abbildung 4-21** aufgelistet.

```
#VRML V2.0 utf8
Transform (
  children
  Shape (
    appearance Appearance {
      material Material {
        ambientIntensity 0.229508
        diffuseColor 0.61 0.61 0.61
        specularColor 0.77 0.76 0.76
        emissiveColor 0 0 0
        shininess 0.9
        transparency 0
      }
    }
    geometry Box (
    )
  }
  translation 0 0 0
)
```

Abbildung 4-21: *VRML*-Code - Würfel

Der Beginn der *VRML*-Datei ist in der Kopfzeile (engl. Header) durch das Rautezeichen (engl. hash) gekennzeichnet. Das Rautezeichen gibt an, dass es sich bei dieser Zeile um eine Kommentarzeile handelt. Des Weiteren wird in dieser Zeile angegeben, dass es sich bei dieser Version des *VRML*-Formates um die Version 2.0 handelt. Die Kennzeichnung „utf8“ bedeutet, dass der Quellcode mit Hilfe des utf8-Standardzeichensatzes erstellt ist. Die Abkürzung „utf8“ steht dabei für *Universal Character Set Transform Format*. Zur Beschreibung des Aufbaus einer *VRML*-Datei wird nur auf den in Abbildung 4-21 dargestellten Quellcode eingegangen. Weiterführende Hinweise zum Aufbau von *VRML*-Hierarchien sind im *VRML ISO/IEC 14772* Standard zu finden.

Die in *VRML* genutzten Objekte sind in drei Knotengruppen untergliedert:

- Gruppenknoten
- Blattknoten
- Untergeordnete Knoten

**Gruppenknoten** werden zur Gruppierung von mehreren Knoten eingesetzt. Häufig wird hier der Knoten *Transform* genutzt, um eine Verschiebung, Rotation oder eine Skalierung des gesamten Gruppenknotens vornehmen zu können. **Blattknoten** können zwar als eigenständige Knoten auftreten, kommt aber häufig nur als Kindknoten (*children*) in einem Gruppenknoten zur Anwendung. In dem Quellcodebeispiel (vgl. Abbildung 4-21) wird der Blattknoten *Shape* verwendet. Der *Shape*- oder auch Gestaltknoten beinhaltet die Beschreibung der geometrischen Gestalt und der Materialeigenschaften des Objektes. Die Gestalt und die Materialeigenschaften werden als **untergeordnete Knoten** *appearance* und *geometry* bezeichnet. Der *appearance*-Knoten beinhaltet dabei die folgenden Parameter zur Einstellung der Materialeigenschaften [Ame97]:

- ambientIntensity
- diffuseColor
- specularColor
- emissiveColor
- shininess
- transparency

Der Wert **ambientIntensity** beschreibt die Stärke der durch Umgebungsbeleuchtung reflektierten Objektfarbe. Diese hängt vom Winkel der Oberfläche in Bezug auf die Lichtquelle ab. Die **diffuseColor**-Parameter geben mit den drei RGB-Farbwerten an, welche Farbe das vom Objekt reflektierte Licht besitzt. Das Feld **specularColor** beinhaltet ebenfalls drei RGB-Farbwerte (Reflexionsgrad), die in Abhängigkeit vom Winkel zwischen der Oberfläche und Lichtquelle sowie vom Winkel zwischen Oberfläche und Betrachter in die Kalkulation der Farbverläufe einfließt. Die Stärke des vom Körper reflektierten Lichtes einer Lichtquelle wird für einen bestimmten Winkelbereich eingestellt. Die Werten der **emissiveColor**-Parameter können ein Leuchten von Objekten erzeugen. Von dem Objekt wird dabei kein Licht abgegeben und andere Objekte erfahren somit auch von diesem Objekt keine Beleuchtung. Der **shininess**-Parameter beschreibt die Glätte der Oberfläche eines Objektes. Während niedrige Werte eine Oberfläche rau und matt erscheinen lässt, bewirken höhere Werte eine eher glatte und glänzende Oberfläche durch schmale Lichtreflexionen. Mit Hilfe des **transparency**-Wertes, wird die Transparenz eines Objektes angegeben.

Der untergeordnete Knoten *geometry* beinhaltet die Geometriebeschreibung des Objektes. Im Beispiel des Würfels (vgl. Abbildung 4-21) wurde der Knoten *Box* (Würfel) genutzt. Wenn dort keine Werte vorhanden sind, sind stets die Standardwerte (engl. default) ausgewählt. In diesem Fall sind es die Werte [1 1 1] für die Kantenlängen des Würfels. Die Werte werden in der Scriptsprache *VRML* nach der Rechten-Hand-Regel angewendet [x y z]. Der *translation*-Knoten gibt als weiterer Kindknoten die Verschiebung des gesamten Gruppenknotens *Transform* an. Hier erfolgt jedoch keine Verschiebung, da die Werte [0 0 0] für eine Verschiebung festgelegt sind.

Die Grundlagen für ein Verständnis des neutralen Datenformates *VRML* sind damit geschaffen. Nun kann die Untersuchung zu den nutzbaren Parametern für die Transformation von CAD-Daten in das *VRML*-Format anschließen.

### 4.3.3 Nutzbare Parameter

Wie schon erläutert, ist das Gesamtkonzept derart gestaltet, dass die Zuweisung der Materialeigenschaften bereits im CAD-System erfolgt. Es ist nun zu prüfen, ob durch die im CAD-System vorgenommenen Zuweisungen von Beleuchtungseinrichtungen, spezifischen Materialparametern oder Texturen sichergestellt wird, dass die Parameter bei dem Export in das allgemeingültige Format *VRML* übertragen werden. Ziel ist die plattformunabhängige Einsetzbarkeit der abgeleiteten VR-Daten. Für die Untersuchung wird der in Abbildung 4-20 dargestellte Würfel verwendet. Dem Würfel werden im CAD-System die in **Tabelle 1** aufgelisteten *SolidWorks*-Parameter (vgl. Abbildung 4-17) und zusätzlich eine Textur in Korbstruktur (vgl. Abbildung 4-18) zugewiesen.

<b>SW - Parameter</b>				<b>VRML - Parameter</b>
<b>Farbeeigenschaften</b>				<b>Farbeinstellungen</b>
R	255		1	R
G	0		0	G
B	0		0	B
<b>Optische Eigenschaften</b>				<b>Materialeigenschaften</b>
Transparenz	0.75		0.75	transparency
Umgebung	0.30		0.30	ambientIntensity
Diffusion	0.90		0.90	diffuseColor
Spiegeleffekt	0.80		0.80	specularColor
Glanz	0.50		0.50	shininess
Emissionsgrad	0.70		0.70	emissiveColor

*Tabelle 1: Farb- und Materialparameter SW - VRML*

Die in *SolidWorks* vorhandenen Standard-Beleuchtungseinheiten sind als Default-Werte gesetzt. Der Würfel wird als *VRML*-Datei exportiert. Anschließend werden die in *SolidWorks* eingestellten Parameter denen aus der *VRML*-Datei extrahierten Parameter in der Tabelle 1 gegenübergestellt.

Nach der Transformation und dem Export der CAD-Daten in eine *VRML*-Datei wird festgestellt, dass weder die Parameter für die Beleuchtungseinrichtungen noch die Parameter für die zugewiesene Textur in die *VRML*-Datei transformiert wurden. Dies bedeutet, es ist mit dem genutzten CAD-System nicht möglich, Texturen oder Beleuchtungen in eine *VRML*-Datei zu übertragen. Aus der Tabelle 1 ist außerdem klar zu erkennen, dass alle Farb- und Materialparameter fehlerfrei in die *VRML*-Datei übernommen wurden. Während die Werte der optischen Eigenschaften keine Veränderung aufweisen, ist bei den RGB-Farbparametern zu sehen, dass sie sich deutlich unterscheiden. Dies ist damit zu erklären, dass in *SolidWorks* die Farbparameter nach einer Skala von 0-255 und in *VRML* von 0-1 eingestellt werden. Dabei entspricht der *SolidWorks* -Wert 255 dem *VRML*-Wert 1 und 0 entspricht 0. Auf eine einfache mit dem Dreisatz durchzuführende Umrechnung der Werte in das jeweilige Format soll jedoch hier nicht näher eingegangen werden. Eine Erklärung wird im Abschnitt 4.4.1 geliefert.

Da jetzt alle nutzbaren Transformationsparameter definiert sind, können die Untersuchungen zur Steigerung des Immersionsgrades folgen.

#### 4.4 Steigerung des Immersionsgrades

Zur Steigerung des Immersionsgrades finden grundsätzlich alle aufgeführten Parameter wie Texturen, Beleuchtungseinrichtungen, die Mechanismen der Raumwahrnehmung und Materialeigenschaften Anwendung. Die Ergebnisse der vorangegangenen Untersuchungen haben jedoch gezeigt, dass die Parameterauswahl zur Steigerung des Immersionsgrades bei diesem Konzept einzugrenzen ist. Texturen und Beleuchtungseinrichtungen sind auszuschliessen, da keine Möglichkeit besteht, die dafür notwendigen Parameter aus dem CAD-System zu exportieren. Es stehen also noch die Mechanismen der Raumwahrnehmung und die Materialeigenschaften als Parameter zur Verfügung.

Wie im Abschnitt 4.3.3 bereits festgestellt wurde, können alle in *SolidWorks* zur Verfügung stehenden Materialeigenschaften (Farbe, optische Eigenschaften) bei dem Konzept Verwendung finden.

Zur Steigerung der Raumwahrnehmung hat Hofmann [Hof02] bereits die egozentrischen Einflussparameter untersucht und nutzbare Ergebnisse geliefert. Diese Parameter sind also bereits zu nutzen. Auf die von Hofmann gelieferten Ergebnisse soll aber in dieser Arbeit nicht weiter eingegangen werden.

Zur Auswahl stehen nun die unterschiedlichen exozentrischen Einflussparameter zur Steigerung der Raumwahrnehmung. Zur Eingrenzung der Parameter wird nochmals auf die Definition des Einsatzbereiches dieses Konzeptes eingegangen. Ziel der stereoskopischen Visualisierung von CAD-Daten ist das schnelle Verständnis von komplexen dreidimensionalen Objekten. Dabei wird der Schwerpunkt auf typische Objekte aus dem Maschinenbau gelegt. Diese Objekte sind klar in ihren Abmaßen definiert und lassen sich kubisch begrenzen. Diese kubische Begrenzung (Würfel) muss nun von allen Seiten gleichermaßen zu betrachten sein. D.h. es muss eine jederzeit gute Raumwahrnehmung bei der Sicht auf das Objekt aus allen sechs Blickrichtungen möglich sein.

Da es sich nur um ein zu visualisierendes Objekt handelt, haben die exozentrischen Einflussparameter: Bewegungsparallaxe, relative Helligkeit, relative Größe, atmosphärische Perspektive, kinetische Tiefe, Höhe im Gesichtsfeld und Interposition keinen bedeutsamen Einfluss auf die Raumwahrnehmung eines kubischen Objektes in einem festgelegten Abstand zum Betrachter.

Als mögliche Mechanismen zur Steigerung der Raumwahrnehmung stehen noch die Farbe, die Schattierung und die lineare und die sphärische Perspektive zur Auswahl. Da das Objekt aus allen Raumrichtungen gleichermaßen betrachtet werden soll, kann die lineare Perspektive nicht zum Einsatz kommen. Damit stehen noch drei Parameter zur Verfügung:

- Farbe (farbiger Tiefeneffekt)
- Schattierung

- Sphärische Perspektive

Für weitere Untersuchungen zur Steigerung des Immersionsgrades sind somit die folgenden Einflussparameter näher zu betrachten:

- Farben
- Schattierung
- Materialeigenschaften
- Sphärische Perspektive

Zunächst werden nachfolgend mögliche Farbeinstellungen untersucht, die zu einer Steigerung des Immersionsgrades beitragen könnten.

#### 4.4.1 Farbe

Das Wort „Farbe“ wird im alltäglichen Gebrauch für unterschiedliche Bezeichnungen genutzt. An dieser Stelle wird der Begriff „Farbe“ mit Hilfe der Norm DIN 5033 Teil 1 genauer definiert:

*„Farbe im Sinne dieser Norm ist ein durch das Auge vermittelter Sinneseindruck, also eine Gesichtsempfindung. Die Farbe ist diejenige Gesichtsempfindung, durch die sich zwei aneinandergrenzende strukturlose Teile des Gesichtsfeldes bei einäugiger Beobachtung mit unbewegtem Auge allein unterscheiden können.“*

Der Farbeindruck wird durch Licht hervorgerufen. Das Licht wird physikalisch als eine Energieverteilung im elektromagnetischen Spektrum beschrieben. Das für den Menschen sichtbare Licht (vgl. **Abbildung 4-22**) ist elektromagnetische Energie im Spektralbereich mit einer Wellenlänge  $\lambda$  zwischen 380nm und 780nm [Sch00].

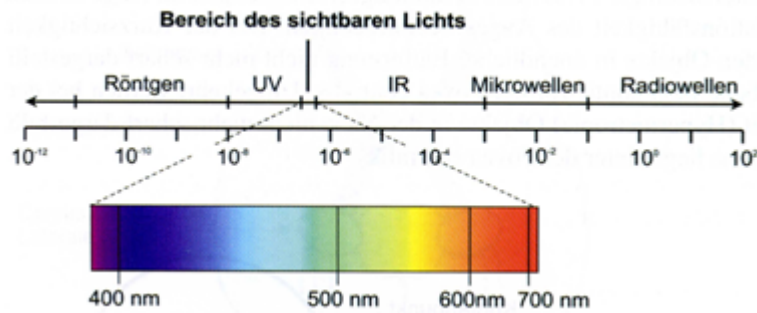


Abbildung 4-22: Spektrum des sichtbaren Lichtes [Sch00]

Das Licht wird durch die Absorption des in das Auge eintretenden Lichtes, welches auf die Netzhaut (Retina) trifft, wahrgenommen. Auf der Retina befinden sich lichtempfindliche Rezeptoren. Diese unterscheiden sich in Stäbchen und Zapfen. Während Stäbchen mit einer Anzahl von ca. 100-120 Millionen das Hell-Dunkel-Sehen gestatten, ermöglichen die Zapfen mit einer Anzahl von ca. 7-8 Millionen eine Farbwahrnehmung. Die Zapfen sind für unterschiedliche Wellenlängenbereiche besonders empfindlich. Nach der Tristimulustheorie gibt es auf der Netzhaut drei Arten von Farbsensoren (Zäpfchen), die jeweils bei rotem, grünem und blauem Licht ihre höchste Empfindlichkeit aufweisen [Fol94]. In **Abbildung 4-23** ist die Funktion der Spektralreaktion für die drei Arten von Zäpfchen auf der menschlichen Retina zu erkennen.

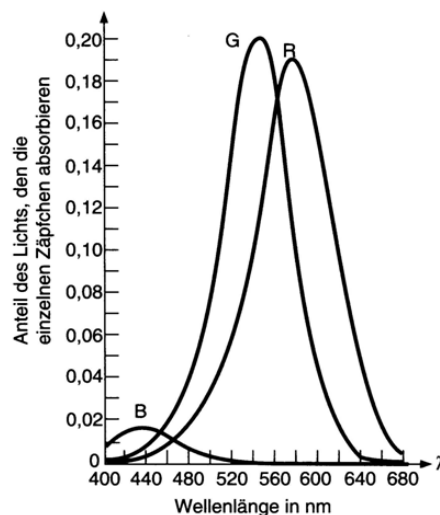


Abbildung 4-23: Spektralreaktion der drei Arten von Zäpfchen auf der Retina [Fol94]

Die Extremwerte der Reaktionen liegen für Blau bei ca. 440nm, für Grün bei ca. 545nm und für Rot bei ca. 580nm. Es ist deutlich zu erkennen, dass das menschliche Auge blaues Licht in einem wesentlich geringeren Maß absorbiert als rotes oder grünes Licht.

Die Fähigkeit, Licht verschiedenen Spektrums in unterschiedlicher Qualität wahrzunehmen, wird als Farbsehen bezeichnet. Eine Farbe entsteht durch die richtige Mischung von Licht bestimmter Intensitäten mit Wellenlängen des sichtbaren Spektrums [Bed03]. Bei der Wahrnehmung von Farben wird unterschieden zwischen chromatischen (bunten) Farben und achromatischen (unbunten) Farben.

Die gleichmäßige Mischung aller sichtbaren Wellenlängen wird als weiß bezeichnet. Eine Verringerung über Grautöne bis zu Schwarz wird durch die gleichmäßige Reduzierung der Intensität aller Wellenlängen erreicht. Achromatische Farben lassen sich allein durch ihre Helligkeit, also der Lichtintensität beschreiben.

Demgegenüber sind chromatische Farben durch den Farbton, die Farbsättigung und die Helligkeit gekennzeichnet. Während der Farbton die Farbe angibt, beschreibt die Sättigung die Reinheit der Farbe (Weiß- bzw. Schwarzanteil). Die drei unterschiedlichen Typen von Farbsensoren bilden die Grundlage der Drei-Farben-Theorie von T. Young (1773 - 1829) und H. von Helmholtz (1821 – 1894). Auf der Basis dieser Drei-Farben-Theorie wurden Farbmodelle konzipiert. Diese Farbmodelle beschreiben dreidimensionale Farbräume. In diesen Farbräumen wird unterschieden zwischen Primär- und Sekundärfarben. Farben, aus denen alle anderen Farben eines Farbraumes erzeugt werden können, werden als Primärfarbe bezeichnet. Alle Farben, die aus zwei Primärfarben erzeugt sind, werden als Sekundärfarbe bezeichnet.

Bei dem **HSV**-Farbmodell kommen die drei Parameter Farbton (**Hue**), Weißanteil in der Farbe (**Saturation**) und Helligkeit (**Value**) für die Farbdarstellung zum Einsatz. Gegenüber dem HSV-Farbmodell besitzt das **RGB**-Farbmodell die Primärfarben **Rot**, **Grün** und **Blau**. Jede andere Farbe *C* in diesem Farbmodell wird durch die relative Gewichtung (*r*, *g*, *b*) der drei Primärfarben (R, G, B) als Summe (vgl. **Gl. 4-1**) dargestellt:

$$C = rR + gG + bB \quad (\text{Gl. 4-1})$$

$$r, g, b \in [0,1]$$

Wenn für die digitale Speicherung der Faktoren eines Farbanteils jeweils 8 Bit zur Verfügung stehen, wird die Farbspezifikation häufig mit  $2^8 - 1 = 255$  angegeben. D.h., der Farbanteil kann die Werte zwischen 0 und 255 annehmen. In dem Datenformat *VRML* werden die Parameter für die Farbspezifikation mit Werten zwischen 0 und 1 beschrieben. Eine Umrechnung ist leicht möglich und wird hier nicht vertieft. Für die Farbparameter des RGB-Modells ergeben sich damit für die Primärfarben folgende Zahlentripel:

- Rot [1 0 0]



- Grün [0 1 0]
- Blau [0 0 1]

Als Beispiel werden drei Sekundärfarben mit dem Zahlentripel

- Gelb [1 1 0]
- Cyan [0 1 1]
- Magenta [1 0 1]

als RGB-Codierung dargestellt. Bei vollen Anteilen aller RGB-Farben wird die Farbe

- Weiß [1 1 1]

und mit keinem Anteil an RGB-Farben wird die Farbe

- Schwarz [0 0 0]

repräsentiert. Das additive RGB-Farbmodell ist in **Abbildung 4-24** zu sehen.



*Abbildung 4-24: Additives RGB-Farbmodell [Her04]*

In dieser Abbildung sind die drei Primärfarben und drei Sekundärfarben als Summe der Primärfarben zu sehen. Die Summe aller drei Primärfarben ist in der Mitte als weiß abgebildet. Als Mischung aus zwei mit jeweils vollem Anteil beteiligten Primärfarben werden

die Sekundärfarben Gelb, Magenta und Cyan repräsentiert. In **Abbildung 4-25** ist der RGB-Würfel zu erkennen.

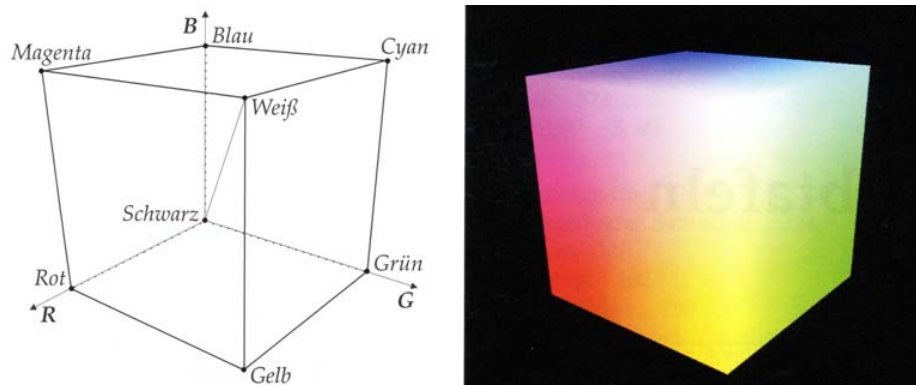


Abbildung 4-25: RGB-Würfel [Bed03]

Während die Primärfarben an den Koordinatenachsen aufgetragen sind, sind die Sekundärfarben in den jeweiligen Hauptebenen aufgetragen. Mit diesem dreidimensionalen RGB-Farbraum mit der Seitenlänge eins werden alle durch dieses Modell darstellbaren Farben repräsentiert. Die Raumdiagonale ausgehend vom Nullpunkt beschreibt dabei alle Grauwerte. Neben den Farbmodellen RGB und HSV existieren noch weitere Farbmodelle, auf die in dieser Arbeit aber nicht weiter eingegangen werden soll.

Die menschliche Wahrnehmung wird durch unterschiedliche Farbeffekte beeinflusst. Bei der räumlichen Anordnung mehrerer Farben können Phänomene auftreten, welche die Wahrnehmung von Einzelfarben beeinflusst. Als Beispiel wird hier der Farbglut- oder Helmholtz-Kohlrausch-Effekt genannt. Bei diesem Effekt wird ein chromatischer Farbreiz heller als ein weißer Reiz gleicher Leuchtdichte empfunden. Die wahrgenommene Helligkeit nimmt mit steigender Farbsättigung zu. Die Ausprägung dieses Effektes hängt zusätzlich vom gewählten Farbton ab. Farbige Farbreize glühen im Vergleich zu achromatischen Farbreizen [Sch00]. Gesättigte Farben gewinnen gegenüber „getrübten“ Farben an Leuchtkraft und treten daher stärker in den Focus des Betrachters. Dieser Effekt wird auch häufig als *Qualitätskontrast* bezeichnet.

Als weiterer Effekt wird der in Abschnitt 4.1 bereits beschriebene farbige Tiefeneffekt genannt. Dieses Phänomen ist jedoch abhängig von der Farbpaarung. Für Farbpaarungen bei denen dieser Effekt festgestellt wurde, gibt Grandis [Gra86] eine Liste an. Diese ist nach dem auftretendem Tiefeneffekt von stark nach schwach sortiert:

- gelb auf schwarz
- orange auf schwarz

- rot auf schwarz
- grün auf schwarz
- blau auf schwarz

Bei einem gelben Objekt auf schwarzen Hintergrund scheint ein gelbes Objekt stärker zu schweben als ein blaues Objekt auf ebenfalls schwarzem Hintergrund.

Zusammenfassend kann aufgrund der erläuterten Sachverhalte und mit dem Verweis auf die entsprechenden Literaturquellen, gesagt werden, dass eine Tiefenwirkung mit Hilfe einer Farbmodulation zu erzeugen ist. Insbesondere in dem von Grandis [Gra86] beschriebenen farbigen Tiefeneffekt wird Potential zur Steigerung der Raumwahrnehmung von virtuellen Prototypen vermutet. Der farbige Tiefeneffekt in Verbindung mit dem von Bender [Bed03] beschriebenen Effekt des *Kalt-Warm-Kontrastes*, der bei der Verwendung der Farben Rot, Orange und Gelb (warme Farbe) Objekte größer und näher erscheinen lässt und bei denen Objekte in den Farben Blau und Violett (kalte Farben) einen Verkleinerungs- bzw. Entfernungseffekt aufweisen, soll näher untersucht werden (vgl. **Abbildung 4-26**).



*Abbildung 4-26: Links warme Farben – rechts kalte Farben*

Dabei ist neben dem Helmholtz-Kohlrausch-Effekt, zusätzlich zu beachten, dass die Empfindlichkeit der Zäpfchen auf Farbreize (vgl. Abbildung 4-23) unterschiedlich stark ausgeprägt ist.

Neben den Primär- und Sekundärfarben existieren noch Komplementärfarben oder auch Gegenfarben. Die Komplementärfarben basieren für additive Farbräume darauf, dass sie sich bei einer Addition zu Weiß ergänzen. Komplementär ist eine Farbe stets zu einer anderen. Wenn sich zwei Farben diagonal gegenüberstehen, sind sie komplementär (vgl. **Abbildung 4-27**).

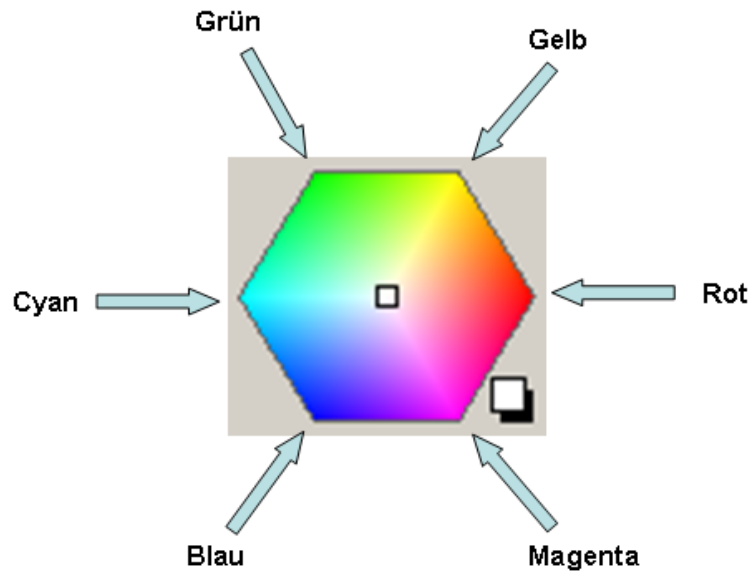


Abbildung 4-27: Komplementärfarben

Im RGB-Farbraum sind beispielsweise Blau zu Gelb, Rot zu Cyan und Grün zu Magenta komplementär. Bei der Auswahl von Farbkombinationen kommt dem *Komplementärkontrast* eine besondere Bedeutung zu. Sind zwei Farben komplementär, verstärken sie sich gegenseitig in ihrer Leuchtkraft. Der Komplementärkontrast wird als subjektiver Kontrast zwischen zwei komplementären Farben beschrieben. Dieser kann in einen *Simultankontrast* und einen *Sukzessivkontrast* unterschieden werden. Simultanwirkungen treten sofort nach Reizdarbietung auf, Sukzessivwirkungen dagegen mit zeitlicher Verzögerung [Skr96]. Mit Hilfe des Simultankontrastes und des Sukzessivkontrastes kann zudem eine Ermittlung von Komplementärfarben erfolgen [Hei61].

Die Komposition von Farben kann den chromatischen Kontrast verstärken. In **Abbildung 4-28** ist zu erkennen, wie die rote Fläche auf dem cyanfarbigen Hintergrund in einem höheren Maße leuchtender erscheint, als auf einem orangefarbenen Hintergrund.

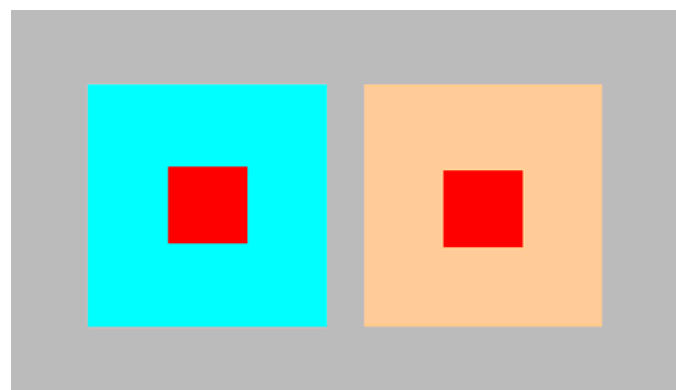


Abbildung 4-28: Simultankontrast

Dieses Phänomen wird *Simultankontrast* oder chromatische Induktion genannt. Insbesondere bei CAD-Anwendungen sind solche Effekte zu vermeiden, da sie den Benutzer durch auftretende Flimmereffekte übermäßig anstrengen [Skr96]. Neben diesem Verstärkungseffekt von Farbwahrnehmungen kann durch den Simultankontrast die Wahrnehmung einer farbigen Fläche durch die Farben der benachbarten Areale beeinflusst werden. Diese „Farbvermischung“ oder „Diffusion“ wird auch als *Color Bleeding* bezeichnet. Ein weiterer Effekt der chromatischen Induktion lässt den Eindruck entstehen, dass Farbtöne vor einem hellen Hintergrund an Helligkeit verlieren, dafür aber an Sättigung zunehmen. Mit einer exakten Abstimmung von Farbkombinationen ist es möglich, die menschliche Wahrnehmung gezielt zu beeinflussen.

Während der *Simultankontrast* bei räumlich angrenzenden Farben auftritt, wirkt sich der Sukzessivkontrast bei der zeitlichen Folge von Farben aus. Wird eine farbige Fläche auf weißem Grund und anschließend eine komplett weiße Fläche betrachtet, erscheint ein Nachbild aus der Komplementärfarbe der vorher fixierten Farbe. Als Beispiel soll eine rote Fläche betrachtet werden. Das Nachbild erscheint als Komplementärfarbe zu Rot in Cyan. Aufgrund der langen Betrachtungsdauer einer Farbe, ist das neuronale Farbsystem nicht mehr im Gleichgewicht. Die Folge ist eine im ersten Moment abweichende Farbwahrnehmung. Dieser Nacheffekt wird als *Sukzessivkontrast* oder auch als Nachbild bezeichnet.

Mit dem Begriff Kontrast werden häufig die Farben Schwarz und Weiß in Verbindung gebracht. Meistens ist aber der *Hell-Dunkel-Kontrast*, also der Helligkeitsunterschied zweier Farben gemeint. In diesem Zusammenhang wird häufig vom Remissionswert gesprochen. Das heißt, in welchem Maße eine Farbe Teile des Lichtspektrums absorbiert. Bei den achromatischen Farben besteht der größte Unterschied zwischen Weiß und Schwarz. Bei den chromatischen Farben besteht der größte Kontrast zwischen Gelb und Violett, da Gelb den höchsten und Violett den geringsten Remissionswert aufweist [Lie97]. Rot und Türkis haben demnach den schwächsten Hell-Dunkel-Kontrast (vgl. **Abbildung 4-29**).

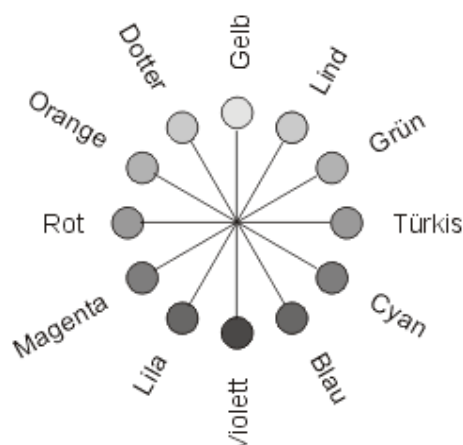


Abbildung 4-29: Hell-Dunkel-Kontrast [Lie97]

Mit dem *Hell-Dunkel-Kontrast* wird der räumliche Bildeindruck verbessert. Bei reduziertem Kontrastverhältnis wirkt eine dargestellte Szene in einem geringeren Maße räumlich als bei einem hohen Kontrastverhältnis.

Bei der Gestaltung einer Szene kommt den Größenverhältnissen von Farbflächen eine besondere Bedeutung zu. Das Größenverhältnis von Farbflächen hat einen erheblichen Einfluss auf die Szene und wird mit dem *Quantitätskontrast* beschrieben. Durch die ungleichmäßige Aufteilung von Farbflächenanteilen kann die Farbwahrnehmung einzelner Farben von der „echten“ Farbe abweichen. Das heißt, sie wirken leuchtender und treten in den Vordergrund oder sie wirken gedämpft und treten in den Hintergrund.

Bei der Kombination von reinen Primärfarben wird der *Farbe-an-sich-Kontrast* oder auch *Bunt-Unbunt-Kontrast* als stärkster Kontrast wahrgenommen. Die Farbkombinationen scheinen zu leuchten. Je mehr die beteiligten Farben abgeschwächt werden, das heißt, dass eine Beimischung von Weiß- und insbesondere von Schwarzanteilen (Graustufen der Farbe) erfolgt, umso geringer wirkt sich dieser Kontrast aus (vgl. **Abbildung 4-30**).

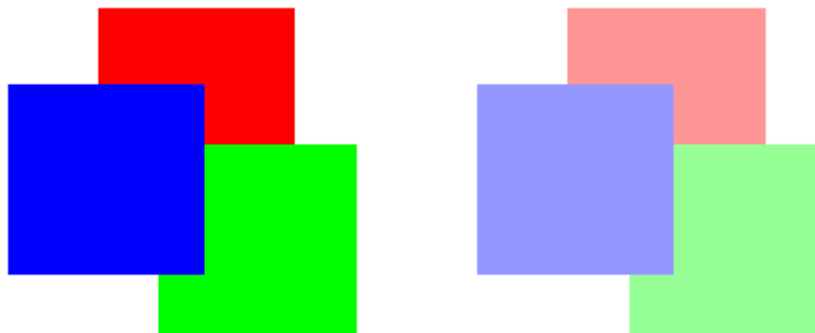


Abbildung 4-30: *Bunt-Unbunt-Kontrast*

Die bisher beschriebenen Effekte und Zusammenhänge bei der Farbwahrnehmung machen die Komplexität deutlich, die das Finden von Farbkombinationen für virtuelle Prototypen zur Steigerung des Immersionsgrades beinhaltet.

Für die Untersuchungen, wie mittels Farbvariation eine Erhöhung der Raumwahrnehmung erreicht wird, kommt ein einfacher Volumenkörper zur Anwendung. Da dieser Volumenkörper zusätzlich mit Materialeigenschaften versehen wird, ist es zunächst notwendig eine grundlegende Kenntnis über Beleuchtungsmodelle zu schaffen.

#### 4.4.2 Beleuchtungsmodelle

Um einen möglichst hohen Immersionsgrad bei der Darstellung von virtuellen Prototypen zu erreichen, können die visualisierten Objekte mit Materialeigenschaften versehen werden. Um den Objekten Materialeigenschaften zuzuweisen, kommen Modelle und Methoden der Computergrafik zur Anwendung, um eine möglichst getreue Nachbildung der Realität zu erzielen. Der Anforderung nach einer unverfälschten Realität steht die aufzuwendende Rechenzeit entgegen, da zahlreiche physikalische Gesetzmäßigkeiten der Lichtausbreitung bei der Berechnung der Bilder zu berücksichtigen sind. Es sind besondere Anforderungen an die Visualisierung gestellt, wenn die Interaktivität bei einer Echtzeitdarstellung gewährleistet sein soll.

Die Visualisierung von Computergrafiken erfolgt durch die Darstellung von Pixel auf Ausgabegeräten wie Monitoren oder Projektoren. Zur Berechnung der Intensität und zur Berechnung der Farbe dieser Pixel sind Beleuchtungsmodelle und Schattierungstechniken zu nutzen. Die Lichteffekte der räumlichen Szene (3D-Punkte) werden zunächst mittels Beleuchtungsmodelle ermittelt und im Anschluss daran die Schattierung durch Schattierungstechniken für die Darstellung auf den Ausgabegeräten (2D-Pixel) berechnet. Um eine deutliche Trennung zwischen Beleuchtung und Schattierung zu erreichen, wird formuliert, dass (3D-) Punkte beleuchtet und (2D-) Pixel schattiert werden.

Das Beleuchtungsmodell bestimmt die Intensität bzw. Farbe eines Punktes auf der Oberfläche eines Objekts. Für die Berechnung der Szenen verwenden die Beleuchtungsmodelle unter anderem die Gesetze der Strahlenoptik. Eines der wichtigsten Gesetze ist das Reflexionsgesetz. Das Reflexionsgesetz beschreibt die Richtung des gespiegelten Lichtstrahls. Gegenüber der ideal spiegelnden Reflexion, bei der ein Lichtstrahl ohne sich aufzuteilen in die nach dem Reflexionsgesetz bestimmte Richtung gespiegelt wird, zerstreut sich der Lichtstrahl bei der real oder unvollkommen spiegelnden Reflexion in unterschiedliche Richtungen. Bei der ideal diffusen Reflexion erfolgt die Reflexion durch eine absolut gleichmäßige Streuung des Lichts in alle Raumrichtungen mit gleicher Intensität.

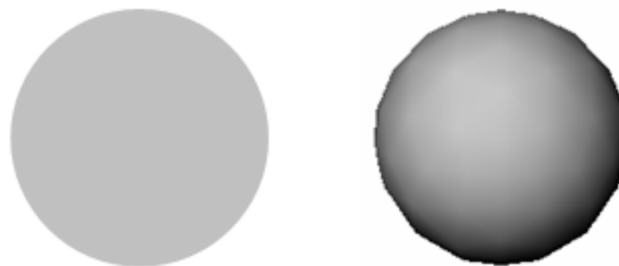
Die bekanntesten Beleuchtungsmodelle sind das Lambert- und das Phong-Beleuchtungsmodell. Während das Lambert-Beleuchtungsmodell die Beleuchtung einer ideal diffus reflektierenden Oberfläche (z.B. Kreide) simuliert, beschreibt das Phong-Beleuchtungsmodell eine unvollkommen spiegelnde Reflexion. Da bei den beiden Beleuchtungsmodellen nicht beleuchtete Flächen auch nicht wahrzunehmen sind, wird eine Hintergrundhelligkeit (ambiente Beleuchtung) definiert, die zu dem jeweiligen Beleuchtungsmodell addiert wird. Das Phong-Beleuchtungsmodell wird durch die drei RGB-Farbkomponenten und den jeweiligen diffusen, spiegelnden und ambienten Reflexionskoeffizienten beschrieben. Diese Reflexionskoeffizienten kennzeichnen die Materialeigenschaften, welche das Material des Objektes charakterisiert.

OpenGL verwendet eine abgewandelte Form des Phong-Beleuchtungsmodells. Dabei wird aus Effizienzgründen die Berechnung des idealen Reflexionsvektors vermieden. Das Phong-Beleuchtungsmodell gilt als Standard-Implementierung zur Beleuchtungsberechnung in der heutigen Grafikhardware [Bed03].

Neben dem Beleuchtungsmodell kommt dem Schattierungsverfahren eine wichtige Bedeutung zu, um den Realitätseindruck zu erhöhen.

#### 4.4.3 Schattierungsverfahren

Wie im Abschnitt 4.1 bereits angesprochen, wird mittels Schattierung (vgl. Abbildung 4-9) die Raumwahrnehmung gesteigert. Zu beachten ist jedoch, dass sich Schattierungen von Schattenwürfen erheblich unterscheiden. Schattierungstechniken werden zur Ermittlung der Farben aller Pixel im Bild verwendet, welche auf Grund von unterschiedlichen Orientierungen der beleuchteten Objektfläche relativ zur Lichtquelle und zum Betrachter auftreten. Eine Schattierung entsteht durch die graduelle Abweichung einer Lichtquelle zur Oberfläche. Dass allein durch Schattierung eine Raumwahrnehmung erfolgen kann, zeigt **Abbildung 4-31**.

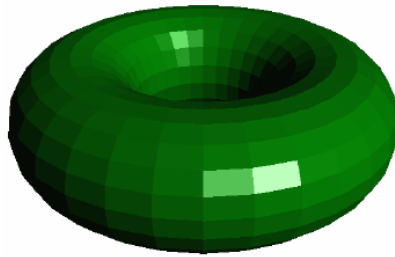


*Abbildung 4-31: Raumwahrnehmung durch Schattierung*

Während die linke Kugel ohne die Anwendung eines Schattierungsverfahrens dargestellt ist, ist die rechte Kugel mit einer Schattierung versehen. Bei der rechten Kugel ist deutlich zu erkennen, dass es sich um ein räumliches Objekt, also ein 3D-Objekt handelt, obwohl nur eine 2D-Darstellung verwendet wird.

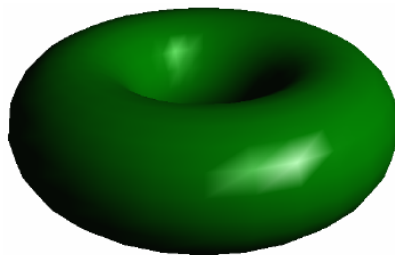
Die bekannten Schattierungsverfahren sind das Flat-, Gouraud- und das Phong-Shading. Das Flat-Shading variiert die Helligkeit einer Facette gemäß dem Winkelunterschied zwischen dem Normalenvektor und dem Lichteinfallswinkel. Das bedeutet, dass der Helligkeitswert nur in einem Punkt der Facettenoberfläche bestimmt wird und die gesamte Facette anschließend in der gleichen Farbe dargestellt wird. Aus diesem Grund wirken die mit dem Flat-Shading ermittelten Oberflächen, wie in **Abbildung 4-32** zu erkennen ist facettiert.





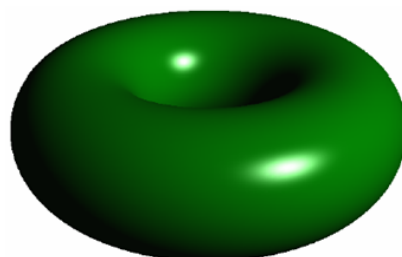
*Abbildung 4-32: Flat-Shading [Har07]*

Im Gegensatz zum Flat-Shading werden beim Gouraud-Shading die Farbwerte der angrenzenden Polygone interpoliert. Das heißt, dass beim Gouraud-Shading die Kanten angrenzender Polygone durch Glättung eine quasi Unsichtbarkeit erfahren. So wirken die Oberflächen realer. Im Vergleich zum Flat-Shading ist eine deutliche Verbesserung in der Darstellung zu erkennen (vgl. **Abbildung 4-33**).



*Abbildung 4-33: Gouraud-Shading [Har07]*

Beim Phong-Shading wird wie beim Gouraud-Shading eine Steigerung der Darstellungsgüte durch die Glättung der Facettenkanten erzielt (vgl. **Abbildung 4-34**).



*Abbildung 4-34: Phong-Shading [Har07]*

Es wird die Farbe jedes Pixels mit Hilfe einer Interpolation der Eckennormalen durch das Beleuchtungsmodell ermittelt. Da die Darstellungsparameter für jeden Pixel durch das Beleuchtungsmodell zu berechnen sind, benötigt dieses Modell erhebliche

Rechenkapazitäten. Aus diesem Grund findet das Phong-Shading in der interaktiven Echtzeitdarstellung von komplexen dreidimensionalen Anwendungen in Stereoprojektion kaum Anwendung.

Die heute vielfach verwendeten Verfahren zur Darstellung von Szenen mittels Computergrafik sind als Beleuchtungsmodell das Phong-Modell und als Schattierungsverfahren das Flat- bzw. das Gouraud-Shading in der jeweils notwendigen Kombination. Diese Verfahren finden auch in der heutigen Grafikhardware die häufigste Verwendung. Das Konzept dieser Arbeit sieht vor, das jeweils von der eingesetzten Hardware angebotene Schattierungsverfahren und Beleuchtungsmodell zu nutzen.

Um eine Aussage über die potentiellen Möglichkeiten zur Steigerung des Immersiongrades mittels Farb- und Materialkomposition zu erhalten, folgen nun dazu die notwendigen Untersuchungen.

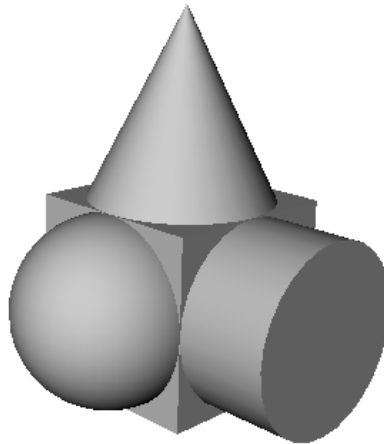
#### 4.4.4 Farb- und Materialparameter

Glänzende Oberflächen, d.h. mit entsprechenden Materialeigenschaften versehene Flächen, verbessern die Tiefenwahrnehmung aus Schattierungen [Tod83]. In besonderem Maße trifft dies bei der binokularen Visualisierung zu [Bla90]. Neben Materialeigenschaften liefern auch Farben einen Beitrag zur Steigerung der Tiefenwahrnehmung. Aus diesem Grund sind Untersuchungen notwendig, inwieweit diese Erkenntnisse genutzt werden können, um bei der Visualisierung von virtuellen Prototypen aus CAD-Daten eine Tiefenwirkung mittels Farbmodulation und Materialzuweisung zu erreichen. Da eine Vielzahl von Farbvarianten und Farbkombinationen möglich sind, wird der Betrachtungsraum durch eine Begrenzung der zu untersuchenden Farben erheblich eingeschränkt.

In einem ersten Schritt ist eine grundlegende Aussage über die Nutzbarkeit von Farben zur Erhöhung der Raumwahrnehmung zu treffen. Daher wird die Auswahl auf die nachfolgend aufgelisteten Farben eingegrenzt: Rot, Grün, Gelb, Magenta, Blau, Cyan, Weiß, Grau und Schwarz. Die Farben Rot, Grün, Blau und Grau werden jeweils in voller Sättigung und zusätzlich in zwei unterschiedlichen Graden von Weißanteilen untersucht. Die aufgestellte Untersuchungsmatrix mit der systematischen Bildnummerierung wird im Anhang in **Tabelle 7** gezeigt. Die Datenaufbereitung und die Visualisierung erfolgt nach der in Abschnitt 4.2 beschriebenen Verfahrensweise.

Um einen engen Zusammenhang zwischen den Untersuchungen und dem praktischen Einsatz von virtuellen Prototypen herzustellen, werden die Untersuchungen mit einem Volumenkörper vor einer Kulisse durchgeführt. Das 3D-Objekt wird in *SolidWorks* konstruiert und mit der höchst möglichen Facettenanzahl in eine *VRML*-Datei exportiert. Dabei wird der Volumenkörper derart gestaltet, dass er einfache Grundgeometrien beinhaltet. Damit wird ein

Bezug zu den in CAD-Systemen häufig verwendeten Körpern Würfel, Kugel, Kegel und Zylinder hergestellt (vgl. **Abbildung 4-35**).



*Abbildung 4-35: Einfacher Volumenkörper*

Insbesondere Bauteile aus dem Maschinenbau lassen sich häufig an diese Körper annähern. Nachdem das Volumenmodell in das Softwaresystem *VRED* importiert wurde, werden der Volumenkörper und die Kulisse gemäß Tabelle 7 eingefärbt. Der nachfolgend durchgeführte Screenshot wird als Bilddatei gespeichert und anschließend in eine weitere Matrix überführt. Die Materialeigenschaften werden für eine erste Untersuchung zunächst nicht berücksichtigt. Das bedeutet, dass alle nutzbaren Materialeigenschaften (vgl. *Abbildung 4-21*), bis auf den Farbwert *diffuseColor* mit dem die Farbeinstellung den Körpers vorgenommen wird, vorerst auf Null gesetzt sind. Die sich aus den oben genannten Farbkombinationen von Kulisse und Volumenmodell ergebende Untersuchungsmatrix ist in **Abbildung 4-36** dargestellt.

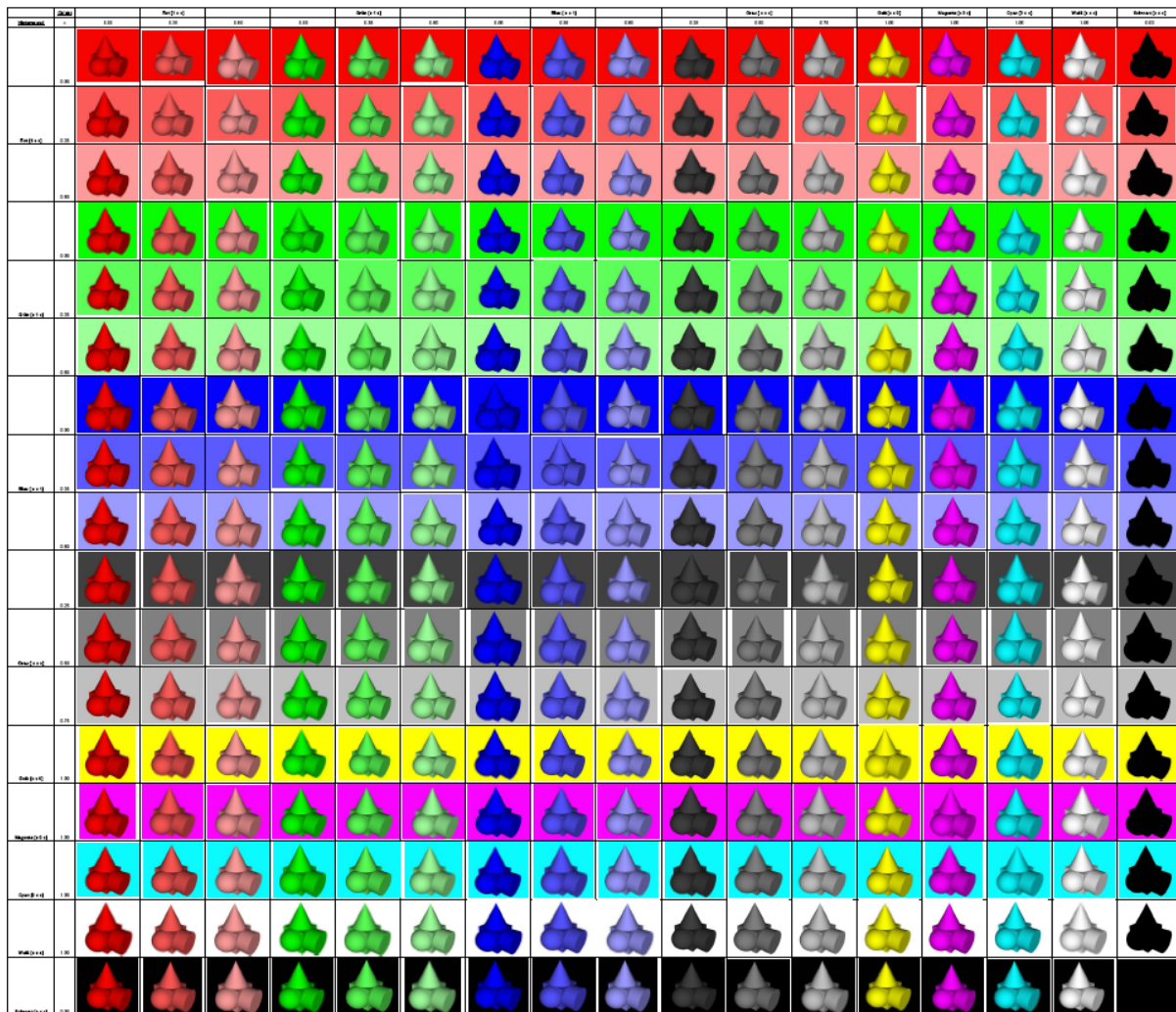


Abbildung 4-36: Untersuchungsmatrix - Farbe

Wie in Abbildung 4-36 zu erkennen ist, wird in den Zeilen die Farbe der Hintergründe und in den Spalten die Farben den Volumenkörpers variiert. Im Anschluss folgt nun die Untersuchung inwieweit die unterschiedlichen Farbkombinationen von Volumenobjekt und Kulisse bei dem Betrachter eine Tiefenwirkung hervorrufen.

### Untersuchung 1:

Die Untersuchung wird an vorhandenen Geräten (vgl. Abschnitt 4.2) mit einfachen Mitteln und einer geringen Anzahl von Probanden durchgeführt, da bei dieser konzeptionellen Arbeit eine tendenzielle Aussage über gute Farbkombination ausreicht. Bei der nachfolgenden Untersuchung werden die Bilder der *Untersuchungsmatrix - Farbe* auf einem Röhrenmonitor der Marke *iiyama* vom Typ *Vision Master Pro 512* visualisiert. Die Darstellung erfolgt nach dem Zufallsprinzip, d.h., es wird eine zufällige Abfolge von den in der *Untersuchungsmatrix - Farbe* geordneten Bildern visualisiert. Jedes Bild wird für die Dauer von 5 Sekunden gezeigt. Anschließend wird der Bildschirm für 10 Sekunden weiß eingefärbt, um die Augen der

Probanden zu neutralisieren. Bei der Untersuchung wird davon ausgegangen, dass männliche und weibliche Probanden sich in den Farbexperimenten ähnlich verhalten. In der Literatur sind jedenfalls keine Befunde bekannt geworden, die dagegen sprechen. Als Probanden stehen 4 männliche Mitarbeiter der Universität Duisburg-Essen im Alter zwischen 27 und 54 Jahren zur Verfügung.

Die visualisierten Abbildungen werden danach bewertet, inwieweit der in der Abbildung dargestellte und kolorierte Volumenkörper gegenüber dem farbigen Hintergrund eine räumliche Differenz (Raumwahrnehmung) empfinden lässt. Dabei wird unterschieden in positive, negative oder keine Raumwahrnehmung.

- Die Raumwahrnehmung wird als positiv bewertet, wenn der Volumenkörper auf dem Hintergrund als schwebend empfunden wird bzw. sich vom Hintergrund in Richtung des Betrachters abhebt. Die Bewertung erfolgt von 1 (geringer Effekt) bis 3 (starker Effekt).
- Die Raumwahrnehmung wird als negativ bewertet, wenn der Volumenkörper gegenüber der farbigen Kulisse in den Hintergrund rückt bzw. den Eindruck empfinden lässt, dass er sich in der Tiefe, also hinter der Kulisse befindet. Die Bewertung erfolgt von -1 (geringer Effekt) bis -3 (starker Effekt).
- Die Raumwahrnehmung wird als nicht vorhanden bewertet, wenn keine räumliche Differenz zwischen Volumenkörper und Hintergrund festzustellen ist. In diesem Fall wird eine Bewertung mit 0 vorgenommen.

Die Dokumentierung der subjektiven Messwerte der Probanden erfolgt durch Ankreuzen der entsprechenden Bewertungsskala auf dem Bewertungsbogen (vgl. Anhang **Abbildung 9-1**).

Die Auswertung der Ergebnisse erfolgt mit dem Tabellenkalkulationsprogramm Microsoft Excel. Für die Auswertung wird zunächst eine Zuordnung aller Einzelwerte zu den entsprechenden Bildnummern vorgenommen. Anschließend werden alle Einzelwerte der Probanden summiert und aus der Summe der arithmetische Mittelwert nach Gl. 4-2 [Dub00] gebildet.

$$x_{\text{arithm}} = \frac{1}{n} \sum_{i=1}^n x_i \quad (\text{Gl. 4-2})$$

Um Aussagen darüber treffen zu können, welche Farbkombinationen bei den Probanden eine positive Tiefenwahrnehmung hervorruft, wird eine Gruppierung der Ergebnisse durchgeführt.

Da die Farbkombinationen, die einen geringeren arithmetischen Mittelwert als 1,5 aufweisen, eine schlechte oder negative Raumwahrnehmung hervorrufen, werden sie für die weiteren

Betrachtungen nicht weiter herangezogen. Die Einordnung erfolgt daher in drei Gruppen A, B und C wie folgt:

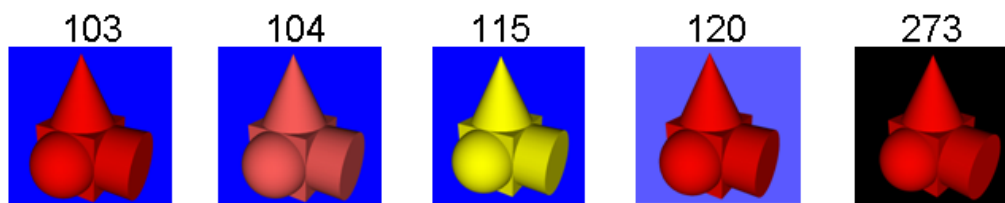
- Gruppe A: [2,5; 3,0]
- Gruppe B: [2,0; 2,5)
- Gruppe C: [1,5; 2,0)

In **Tabelle 2** sind die Bildnummern zu den Gruppen A, B und C zugeordnet. Es ist eine deutliche Gewichtung zwischen den Gruppen zu erkennen.

Gruppe	Bildnummer									
A	103	104	115	120	273					
B	86	99	106	107	114	116	118	121	184	188
	277									
C	32	82	84	87	105	108	112	117	124	125
	132	134	135	154	155	161	201	208	229	257
	260	274	275	276	281	285	286	287	288	

*Tabelle 2: Gruppierung der Ergebnisse*

Nach der Analyse der Ergebnisse lassen sich für die Gruppe A nur 5 Bilder, für die Gruppe B 11 Bilder und für die Gruppe C 29 Bilder ermitteln. Die besten Ergebnisse lieferten die Bilder der Gruppe A (vgl. **Abbildung 4-37**).



*Abbildung 4-37: Gruppe A – Positive Tiefenwirkung*

Es ist eine deutliche Tendenz zu einem blauen Hintergrund zu sehen. Zudem stellt sich heraus, dass in der Überzahl rote Körper neben einem gelben Körper von den Probanden auf einem blauen Hintergrund als schwebend empfunden wird. Die Bilder der Gruppe B sind in **Abbildung 4-38** dargestellt.

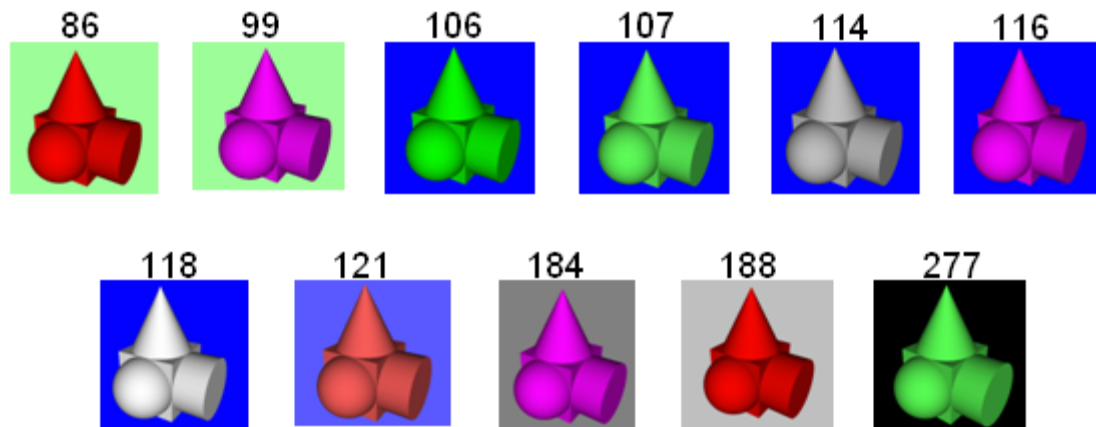


Abbildung 4-38: Gruppe B – Positive Tiefenwirkung

Auch hier fällt eine deutliche Tendenz zu einem blauen Hintergrund auf. Gegenüber der Gruppe A sind in der Gruppe B zusätzlich auch grüne, graue und ein schwarzer Hintergrund als positiv bewertet worden. In der Gruppe B fällt außerdem auf, dass auch andere Objektfarben wie Grün, Grau, Weiß und Magenta neben Rot und Gelb eine positive Bewertung erhalten haben.

Nach der Bestimmung der Standardabweichung  $s$  für die Bildbewertungen nach Gl. 4-3 [Dub00], wird festgestellt, dass die Probanden eine annähernd gleiche Bewertung bei der Beurteilung der Bilder vorgenommen haben.

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - x_{arithm})^2} \quad (Gl. 4-3)$$

Zudem ist auffällig, dass die Probanden eine signifikante Differenz zwischen den Bildern in der Tiefenwahrnehmung bemerken. Dies lässt vermuten, dass der farbige Tiefeneffekt einen Betrag zur Erhöhung der Tiefenwahrnehmung leisten kann. Die gesamte Auswertung inklusive der Gruppe C ist als Farbcodierung: Gruppe A – Rot, der Gruppe B – Blau und der Gruppe C – Schwarz im Anhang in **Tabelle 7** dargestellt.

Die Ergebnisse der Untersuchung 1 werden nun diskutiert, um Empfehlungen für die Parametereinstellungen ableiten zu können. Mit Hilfe der in Abschnitt 4.4.1 aufgeführten Farbphänomene wird versucht Zusammenhänge zu erkennen und herzustellen, um die Erkenntnisse in eine optimale Parameterempfehlung einfließen zu lassen. Ziel dieser Empfehlungen ist es, den farbigen Tiefeneffekt so zu beeinflussen, dass der Betrachter aufgrund der Farbmodulation von Objekt und Hintergrund die Empfindung hat, dass sich das Objekt vom Hintergrund abhebt und in seine Richtung zu schweben scheint.

### **Diskussion der Untersuchungsergebnisse 1:**

Die ersten vier Bilder der Gruppe A (vgl. Abbildung 4-37) können mit dem *Kalt-Warm-Kontrast* in Verbindung gebracht werden, da die Objekte mit den Farben Rot und Gelb und der Hintergrund in Blau eingefärbt sind. Aufgrund dieser Farbkombination der Szene wirkt das Objekt warm und nah und der Hintergrund kalt und fern. Der Volumenkörper scheint sich deutlich vor dem Hintergrund abzuheben und zu schweben. Neben dem *Kalt-Warm-Kontrast* kann auch der *Hell-Dunkel-Kontrast* eine Tiefenwirkung erzeugen, da die Farben Rot und Gelb heller sind als Blau (vgl. Abbildung 4-29) und damit stärker in den Focus des Betrachters gelangen. Zudem kann die erhöhte Präsenz des Objektes durch die Tristimulustheorie gestützt werden, denn die Zäpfchen reagieren auf das rote Objekt wesentlich stärker als auf den blauen Hintergrund (vgl. Abbildung 4-23). Zwar sind die Farben Rot und Blau in voller Sättigung dem *Bunt-Unbunt-Kontrast* zuzuordnen, es wird jedoch vermutet, dass sich dieser Effekt hier nicht auf die Raumwahrnehmung auswirkt. Der von Grandis [35] beschriebene *farbige Tiefeneffekt* stimmt zwar mit dem Bild 273 überein, das vermutete hohe Potential zur Steigerung der Raumwahrnehmung kann jedoch auf der Grundlage dieser Ergebnisse nicht bestätigt werden, da lediglich ein Bild diesem Effekt eindeutig zuzuordnen ist. Der *Simultankontrast* kann, abgesehen von Bild 115, ausgeschlossen werden, da die Farben Rot und Blau bzw. Schwarz nicht komplementär sind. In dem Bild 115 verstärken sich zwar die Farben Gelb und Blau aufgrund des *Simultankontrastes* in ihrer Leuchtkraft, ein Einfluss auf die Raumwahrnehmung wird darin jedoch nicht vermutet.

Zu den Bildern der Gruppe B (vgl. Abbildung 4-38) trifft die oben erläuterte Argumentation gleichermaßen zu. Die Bilder 106 und 107 kombinieren zwar die kalten Farben Blau und Grün, aufgrund des *Hell-Dunkel-Kontrastes* tritt das grüne Objekt jedoch stärker in den Vordergrund.

Die Bilder 86 und 99 sind durch den *Qualitätskontrast* gekennzeichnet, da der Hintergrund gegenüber den Objekten in der Farbsättigung reduziert ist und damit an Leuchtkraft verliert und die Beachtung durch den Betrachter verringert. Die Bilder 114 und 118 folgen dieser Argumentation. Die Bilder 184 und 188 könnten mit dem *Helmholtz-Kohlrausch-Effekt* erklärt werden, da ein chromatischer Farbreiz heller als ein grauer Farbreiz empfunden wird.

Festzuhalten bleibt, dass der farbige Tiefeneffekt nicht die erhofften Ergebnisse geliefert hat. Dies könnte darauf zurückzuführen sein, dass Grandis [Gra86] bei seinen Untersuchungen keine schattierten Objekte eingesetzt hat. Es wird vermutet, dass durch die auf dem Objekt vorhandenen Schattierungen eine gewisse Unschärfe in den Farbübergängen von Objekt und Hintergrund entsteht und auf Grund dessen der farbige Tiefeneffekt in seiner Wirkung negativ beeinflusst wird. Die Untersuchungsergebnisse liefern zudem das Ergebnis, dass die Kombination von Hintergründen insbesondere in Blautönen mit Objekten in kontraststarken Farben, insbesondere der Farben Grün bis Magenta eine erhöhte Tiefenwahrnehmung beim



Betrachter erzeugen. Hinzuzufügen ist noch, dass auch Hintergründe in hellen Grautönen in Kombination mit chromatischen Objekten eine gute Tiefenwahrnehmung hervorrufen.

Aufgrund der gewonnenen Erkenntnisse wird die folgende Auswahl an Hintergrund- und Objektfarben für die nachfolgenden Untersuchungen getroffen:

#### Hintergrundfarben:

- Grün [0.60 1 0.60]
- Blau [0 0 1]
- Blau [0.35 0.35 1]
- Grau [0.5 0.5 0.5]
- Grau [0.75 0.75 0.75]
- Schwarz [0 0 0]

#### Objektfarben:

- Rot [1 0 0]
- Rot [1 0.35 0.35]
- Grün [0 1 0]
- Grün [0.35 1 0.35]
- Grau [0.75 0.75 0.75]
- Gelb [1 1 0]
- Magenta [1 0 1]
- Weiß [1 1 1]

Nachdem der Farbeinfluss bei der Darstellung von virtuellen Modellen untersucht wurde, wird nun der Einfluss der Materialeigenschaften näher betrachtet.

#### **Untersuchung 2:**

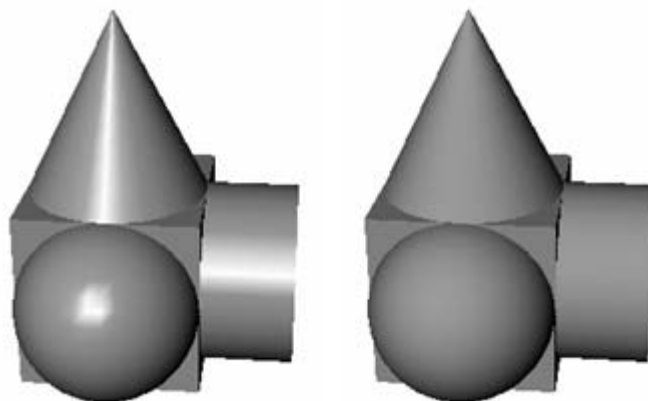
Neben den Farbvariationen können auch definierte Materialeigenschaften einen Betrag zur Steigerung der Raumwahrnehmung und damit zur Steigerung des Immersionsgrades leisten. Dem als Untersuchungsgegenstand entworfenen Volumenkörper wird eine metallische Materialeigenschaft zugewiesen. Da in *SolidWorks* für die Definition von

Materialeigenschaften nur die Möglichkeit einer Texturzuweisung besteht, wird auf die vorhandenen Parameter in der Materialbibliothek von *Cosmo Worlds* zurückgegriffen. Das aus *SolidWorks* exportierte Objekt wird als *VRML*-Datei in *Cosmo Worlds* importiert und mit den gewünschten Materialeigenschaften versehen. Es werden zwei Varianten erstellt. Der Variante 1 sind Parameter zugewiesen, die einen guten Eindruck einer metallischen Oberfläche wiedergeben. Die Variante 2 wird mit dem gleichen Farbton ohne spezielle Materialeigenschaften belegt (vgl. **Tabelle 3**).

<b>Farbeigenschaften (diffuseColor)</b>	Variante 1	Variante 2
R	0.61	0.61
G	0.61	0.61
B	0.61	0.61
<b>Optische Eigenschaften</b>		
Transparency (Transparenz)	0.00	0.00
ambientIntensity (Umgebung)	0.23	0.00
diffuseColor (Diffusion)	0.61	0.61
specularColor (Spiegeleffekt)	0.77	0.00
shininess (Glanz)	0.90	0.00
emissiveColor (Emissionsgrad)	0.00	0.00

*Tabelle 3: Materialzuweisung*

Die zweite Variante erhält demnach nur den entsprechenden Diffuse-Wert für die Farbrepräsentation. Anschließend werden beide Varianten mit *VRED* visualisiert (vgl. **Abbildung 4-39**).



*Abbildung 4-39: Objekt mit und ohne metallische Materialeigenschaften*

Im Vergleich der beiden Varianten erscheint das mit metallischen Materialeigenschaften versehene Objekt wesentlich realer als das Objekt ohne spezielle Materialeigenschaften. Die monokularen Parameter der Materialeigenschaften sollen aufgrund dieses Ergebnisses

vorerst nicht weiter untersucht werden, da eine deutliche Steigerung des Realitätsgrades durch die Zuweisung von speziellen Materialeigenschaften und damit ein erhöhter Realitätseindruck deutlich erkennbar ist.

Auf Grund der Variation der Farbeinstellungen von Objekt und Kulisse und einer Zuweisung von Materialparametern zum Objekt, sind klare Hinweise zu erkennen, dass damit möglicherweise eine Erhöhung der Tiefenwirkung und eine Steigerung des Immersionsgrades zu erreichen ist.

Nun sind in einem weiteren Schritt die Ergebnisse der Farb-, und Materialuntersuchungen miteinander zu verknüpfen und bei der stereoskopischen Visualisierung der optimierten Gesamtszene in Bezug auf mögliche Ghosting-Effekte zu untersuchen.

#### **4.4.5 Ghosting-Reduktion**

Einen großen Betrag zu einem hohen Immersionsgrad liefert die fehlerfreie stereoskopische Darstellung einer Szene. Dies bedeutet, dass die in Abschnitt 2.7.2 beschriebenen Crosstalk-Effekte (vgl. Abbildung 2-19) und der damit vom Betrachter verbundenen Wahrnehmung von Ghosting-Effekten, eine erhebliche Reduzierung des Immersionsgrades bewirken können und aus diesem Grund von besonderer Wichtigkeit sind.

In dieser Arbeit wird ein neuer Lösungsansatz aufgezeigt, um Ghosting-Effekte durch Farbmodulation des Prototypen und des Raumes zu reduzieren, oder sogar komplett zu eliminieren. Damit könnte ein bedeutender Beitrag zur Steigerung des Immersionsgrades geleistet werden. Es ist zu untersuchen, ob sich durch bestimmte Farbkombinationen die Perzeption so beeinflussen lässt, dass Ghosting-Effekte nicht mehr wahrzunehmen sind. Die Perzeption wird hier als die Totalität der Wahrnehmungs- und Empfindungsprozesse beim Menschen definiert. Dabei wird auf die in Abschnitt 4.4.1 beschriebenen Effekte über den Einfluss auf die individuelle Wahrnehmung von Farbnuancen beim Menschen zurückgegriffen.

Ziel ist es, die durch unvollständige Polarisation auftretenden Crosstalk-Effekte und die dadurch hervorgerufenen Ghosting-Effekte beim Betrachter einer stereoskopischen Visualisierung mit Hilfe einer geeigneten Farbmodulation zu minimieren. Das heißt, die aufgrund mangelnder Polarisation nach wie vor vorhandenen Crosstalk-Effekte sollen nur mit Hilfe einer geschickten Farbkomposition vom Betrachter nicht mehr als Ghosting-Effekte bemerkt werden.

Ausgehend von den Ergebnissen der Untersuchung 1 zur Steigerung der Tiefenwirkung mittels Farbmodifikation wird untersucht, inwieweit die als positiv bewerteten Parametereinstellungen Ghosting-Effekte hervorrufen. Bei der Untersuchung wird der in Abbildung 4-35 dargestellte Volumenkörper verwendet. Die Materialeigenschaften des

Objektes sind nach Tabelle 3 Variante 2 justiert. Für die stereoskopische Visualisierung wird das im Abschnitt 4.2 beschriebene Projektionssystem genutzt.

### **Untersuchung 3:**

Zunächst werden die als positiv bewerteten Parametereinstellungen (vgl. Abschnitt 4.4.4) der Untersuchung 1 zu einer neuen Matrix zusammengeführt. Um die Vergleichbarkeit der Untersuchungsergebnisse zu erleichtern, wird der Szene, bei der die gleichen Parametereinstellungen wie in der Untersuchung 1 (vgl. Tabelle 7) vorgenommen sind, auch die gleiche Nummerierung zugeordnet. Die so aufgestellte Untersuchungsmatrix mit den zugeordneten Bildnummern ist in **Tabelle 4** zu erkennen.

	Objekt	Rot [1 x x]		Grün [x 1 x]		Grau [x x x]	Gelb [x x 0]	Magenta [ x 0 x]	Weiß [x x x]
Hintergrund	X	0.00	0.35	0.00	0.35	0.75	1.00	1.00	1.00
Grün [x 1 x]	0.60	69	70	72	73	80	81	82	84
Blau [ x x 1]	0.00	103	104	106	107	114	115	116	118
	0.35	120	121	123	124	131	132	133	135
Grau [ x x x]	0.50	171	172	174	175	182	183	184	186
	0.75	188	189	191	192	199	200	201	203
Schwarz [x x x]	0.00	273	274	276	277	284	285	286	288

*Tabelle 4: Untersuchungsmatrix - 2*

Nach dem Import des Volumenkörpers in *VRED*, werden die Farbparameter des Körpers und des Hintergrundes nach der *Untersuchungsmatrix - 2* eingefärbt und der anschließend durchgeführte Screen-Shot in der *Untersuchungsmatrix – Ghosting-Effekt* (vgl. **Abbildung 4-40**) zusammengeführt.

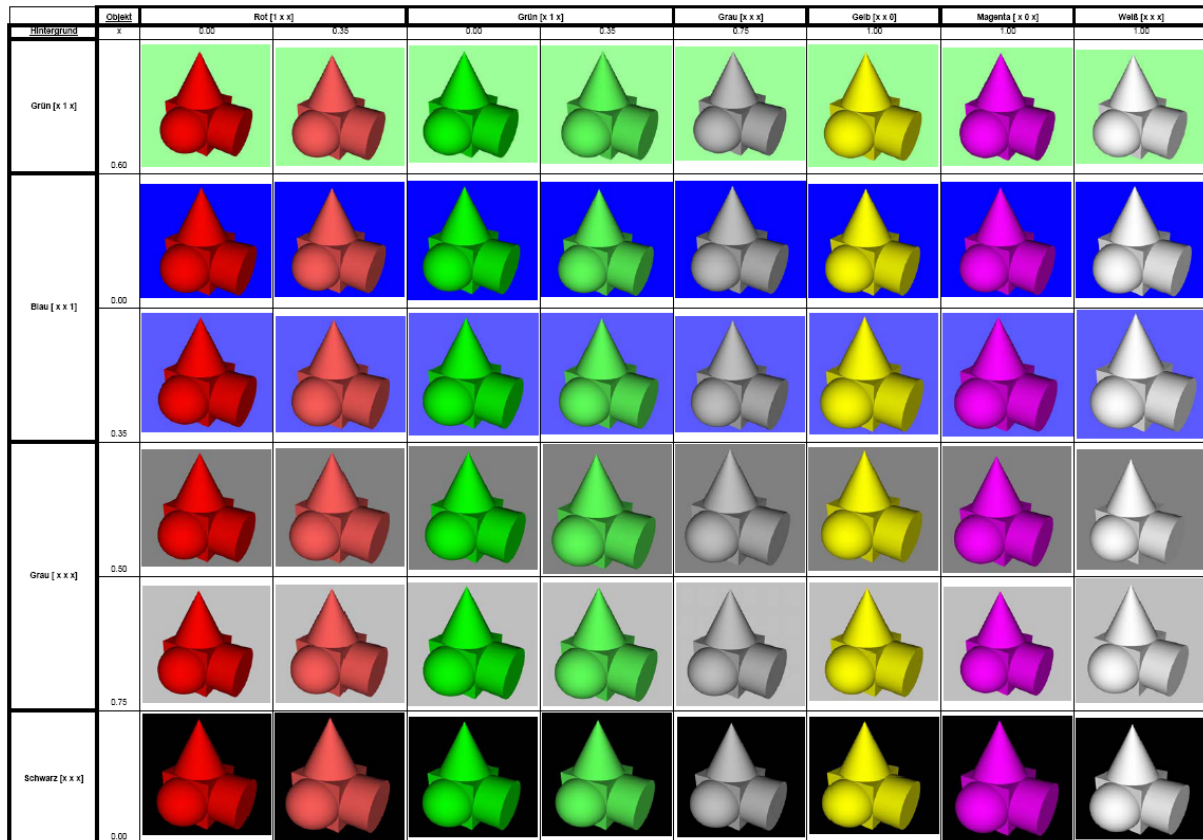


Abbildung 4-40: Untersuchungsmatrix - Ghosting-Effekt

Die Visualisierung der Szenen erfolgt wieder nach dem Zufallsprinzip, d.h., es wird eine zufällige Abfolge von den in der Untersuchungsmatrix geordneten Bildern stereoskopisch visualisiert. Während der Darstellung bleibt der Volumenkörper in seiner Position fixiert. Die Gesamtvisualisierungsdauer der dargestellten Szene beträgt ca. 10 Sekunden. Für die Bewertung der Szenen stehen die gleichen Probanden wie in der Untersuchung 1 zur Verfügung.

Die visualisierten Szenen sind danach zu bewerten, wie stark ein Ghosting-Effekt zwischen Objekt und Hintergrund wahrgenommen wird. Die Bewertung erfolgt von 0 (kein Ghosting-Effekt) bis 3 (starker Ghosting-Effekt). Die subjektiven Messwerte der Probanden werden durch Ankreuzen der entsprechenden Bewertungsskala auf dem Bewertungsbogen (vgl. **Abbildung 9-2**) dokumentiert. Die Auswertung erfolgt nach der in Untersuchung 1 beschriebenen Weise. Die Ergebnisse sind wie folgt gruppiert:

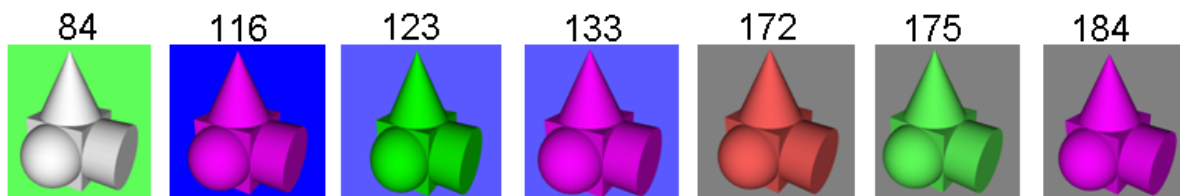
- Gruppe A: [0; 0,5]
- Gruppe B: (0,5; 1]
- Gruppe C: (1, 3]

Da die Gruppe C alle Darstellungen beinhaltet, die starke Ghosting-Effekte hervorrufen, werden diese für die weiteren Betrachtungen nicht weiter herangezogen. In **Tabelle 5** sind die Bildnummern zu den Gruppen A und B zugeordnet.

Gruppe	Bildnummer								
A	84	116	123	133	172	175	184		
B	80	104	121	124	131	191	199	200	203

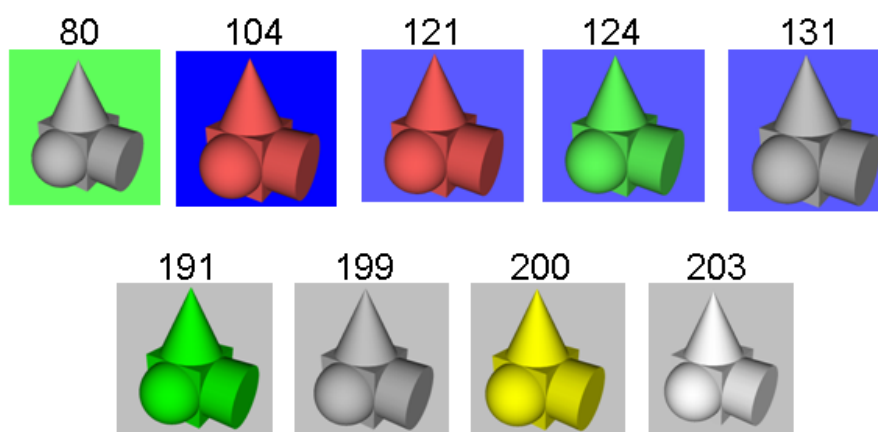
*Tabelle 5: Gruppierung der Ghosting-Ergebnisse*

Nach der Analyse der Ergebnisse lassen sich für die Gruppe A 7 Bilder und für die Gruppe B 9 Bilder ermitteln. Die geringsten Ghosting-Effekte wurden bei den Bildern der Gruppe A (vgl. **Abbildung 4-41**) bemerkt.



*Abbildung 4-41: Gruppe A – Sehr geringe Ghosting-Effekte*

Wie die Bilder der Gruppe A zeigen, ist hier eine erhöhte Tendenz zu einem blauen und einem grauen Hintergrund zu erkennen. Außerdem ist zu sehen, dass die Farben Rot, Grün und Magenta als Objektfarben dominieren. Das Bild 84 ist gesondert zu betrachten, da hier ein weißer Körper vor einem hellgrünen Hintergrund dargestellt wird. Die Szenenbilder der Gruppe B sind in **Abbildung 4-42** zu sehen.



*Abbildung 4-42: Gruppe B – Geringe Ghosting-Effekte*

In der Gruppe B fällt ebenfalls eine deutliche Tendenz zu blauen und grauen Hintergründen auf. Im Vergleich zu der Gruppe A sind in der Gruppe B auch drei graue und ein gelbes Objekt vorhanden.

Bei dieser Untersuchung kann ebenfalls festgestellt werden, dass die Probanden eine annähernd gleiche Bewertung bei der Beurteilung der Szenen in Bezug auf Ghosting-Effekte vorgenommen haben. Außerdem fällt auf, dass die Probanden eine deutliche Differenz in der Stärke des Ghosting-Effekts zwischen den unterschiedlichen Szenen bemerkt haben. Die Ergebnisse führen zu der Annahme, dass es möglich sein kann, eine Ghosting-Reduktion mittels Farbmodulation zu erreichen.

Die Ergebnisse der Untersuchung 3 werden im Anschluss diskutiert, um Empfehlungen für eine optimale Farbeinstellung und Farbkombination von Objekt und Hintergrund abzuleiten. Auf der Grundlage der Untersuchungsergebnisse wird versucht Zusammenhänge zu erkennen, um die Erkenntnisse in eine optimale Farbkomposition einfließen zu lassen. Ziel dieser Empfehlungen ist es, Ghosting-Effekte bei der stereoskopischen Visualisierung von virtuellen Prototypen weitestgehend zu vermeiden.

### **Diskussion der Untersuchungsergebnisse 3:**

Alle Farbkombinationen der Gruppe A in Abbildung 4-41 weisen nur sehr geringe (fast keine) Ghosting-Effekte auf. Die Bilder der Gruppe A zeigen, dass die Hintergrundfarben Blau und Grau mit einer Anzahl von jeweils drei Bildern gleich gute Ergebnisse liefern. Diese Hintergrundfarben werden daher als Favoriten eingestuft. Der blaue Hintergrund eignet sich im Besonderen für Objektfarben in voller Sättigung wie Magenta und Grün (Bild 116 und 133). Auf grauem Hintergrund neigen die Objektfarben Rot, Grün und Magenta zu sehr geringen Ghosting-Effekten (Bild 172, 175, 184). Rot und Grün sind in diesen Fällen jedoch mit Weißanteilen versetzt, Magenta hingegen nicht. Mit dem Bild 84, welches ein weißes Objekt vor einem grünen Hintergrund zeigt, ist kein Zusammenhang zu den übrigen Bildern der Gruppe A herzustellen.

Die Farbkombinationen der Gruppe B in Abbildung 4-42 weisen geringe Ghosting-Effekte auf. Diese Effekte sind als stärker einzustufen als die Effekte der Gruppe A. Sie sind jedoch im Vergleich zu den stets vorhandenen Ghosting-Effekten bei einer Stereoprojektion noch als sehr akzeptabel einzustufen und reduzieren hier nicht den Immersionsgrad. Auch in der Gruppe B muss der blaue und der graue Hintergrund als Favorit genannt werden. In Kombination mit einem blauen Hintergrund sind die roten Objekte und das grüne Objekt jedoch mit Weißanteilen versetzt (Bild 104, 121, 124). Das graue Objekt (Bild 131) ist ebenfalls als mit Weißanteilen versetzt zu bezeichnen. Vor einem grauen Hintergrund sind die Objektfarben Grün und Gelb (Bild 191, 200) als nutzbar zu nennen. Graue bzw. weiße

Objekte zeigen vor einem grauen Hintergrund erwartungsgemäß nur eine geringfügige Neigung zu Ghosting-Effekten (Bild 199, 203). Zu bemerken ist, dass ein graues Objekt vor einem grünen Hintergrund (Bild 80) als gute Kombination empfunden wird.

Aufgrund der gewonnenen Erkenntnisse aus der Untersuchung 3 wird die folgende Empfehlung für Hintergrund- und Objektfarben getroffen. Da für die Hintergrundfarbe Blau zwei Varianten zur Verfügung stehen, wird die Variante mit den meisten positiven Bewertungen herangezogen. Für die Hintergrundfarbe Blau [0.35 0.35 1] sollten die aufgelisteten Objektfarben Verwendung finden:

- Magenta [1 0 1]
- Grün [0 1 0]
- Grün [0.35 1 0.35]
- Rot [1 0.35 0.35]
- Grau [0.75 0.75 0.75]

Für die Hintergrundfarbe Grau stehen ebenfalls zwei Varianten zur Auswahl. Da das Gesamtkonzept dieser Arbeit jedoch eine Erhöhung der Tiefenwirkung mittels Farbmodulation anstrebt, wird die Farbe Grau [0.5 0.5 0.5] als Hintergrundfarbe ausgewählt. Diese Hintergrundfarbe ermöglicht eine Kombination mit dem Großteil der für eine Tiefenwirkung empfohlenen chromatischen Objektfarben. Nachfolgend sind die Objektfarben aufgelistet, die in Kombination mit der grauen Hintergrundfarbe geringe Ghosting-Effekte hervorrufen:

- Rot [1 0.35 0.35]
- Grün [0.35 1 0.35]
- Magenta [1 0 1]

Im Anschluss soll nun untersucht werden, inwieweit die Tiefenwirkung bei der stereoskopischen Betrachtung von virtuellen Prototypen zusätzlich zu steigern ist.

#### **4.4.6 Sphärische Perspektive**

In Abschnitt 4.1 wird erläutert, wie eine monokulare Tiefenwirkung durch die Anwendung einer perspektivischen Darstellung erzielt wird. Ein weiteres Ziel dieses Konzeptes (vgl.



Abschnitt 4.4) ist es nun, die Raumwahrnehmung bei der Betrachtung eines Volumenkörpers mittels sphärischer Perspektive (vgl. Abbildung 4-11) zu steigern. Aufbauend auf den vorangegangenen Erkenntnissen wird bei der Untersuchung zur Nutzbarkeit eines sphärischen Raumes zur Erhöhung der Tiefenwirkung von virtuellen Prototypen wie folgt vorgegangen: Zu Beginn werden Räume konzipiert, die möglicherweise eine erhöhte Raumwahrnehmung bei der Betrachtung des Prototypen bewirken können. Nachfolgend wird der Volumenkörper im Zentrum des Raumes positioniert und die beim Betrachter hervorgerufene Raumwahrnehmung beurteilt.

Grundsätzlich soll der Prototyp mit einer frei beweglichen Kamera zu betrachten sein. Dies bedeutet, dass bei einer Blickwinkeländerung der Kamera eine möglichst gleich bleibende sphärische Perspektive zu gewährleisten ist. Aus diesem Grund muss ein zu gestaltender Raum nahezu rotationssymmetrisch aufgebaut sein.

In einem ersten Ansatz wird eine Tiefenwirkung durch stabförmige Elemente, die parallel zur Blickachse des Betrachters angeordnet sind, erzeugt. Auf die stabförmigen Elemente wird zukünftig nur noch mit der Bezeichnung „Elemente“ Bezug genommen. Um eine regelmäßige Anordnung der Elemente zu gewährleisten, entscheidet sich der Autor vorerst für ein kubisches Arrangement. Die kubische Gestalt bietet zudem den Vorteil, dass hier eine einfache und schnelle Änderung der Elemente mit Hilfe eines CAD-Systems möglich ist, welches die Parametrik-Technik nutzt. Dies ist bei der Erstellung zahlreicher Varianten von erheblichem Vorteil. Es ist also ein würfelförmiges Volumenobjekt zu entwerfen, in dem stabförmige Elemente in einer definierten Aufstellung positioniert sind. Die Elemente sollen dabei einen Hohlraum in der Mitte des Würfels begrenzen, in dem das Volumenobjekt zu positionieren ist. Der Hohlraum muss derart gestaltet sein, dass der Kamerablickwinkel weder durch das Objekt, noch durch die Elemente bei einer rotatorischen Umrundung des Objektes gestört wird. Die Elemente sind anschließend hinsichtlich der nachfolgenden Einflussparameter zu untersuchen und zu optimieren:

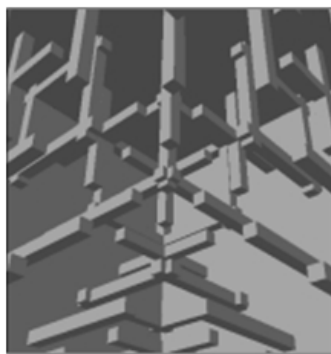
- Elementlänge
- Elementdichte
- Elementquerschnittsgeometrie

Die Gestaltung der für die Untersuchung verwendeten kubischen Raummodelle mit den jeweiligen Stabgeometrien erfolgt in dem CAD-System *SolidWorks*. Diese werden dann in das neutrale Datenformat *VRML* exportiert. Für die nachfolgenden Untersuchungen wird der in Abbildung 4-35 dargestellte Volumenkörper in *CosmoWorlds* importiert. Um von den Abmessungen des Volumenkörpers unabhängig zu sein, wird eine Normierung (Skalierung) des Objektes auf die maximale Kantenlänge eins durchgeführt. Da sich durch eine

Skalierung der Parameter auch die Position des Objektes verändert, muss eine Koordinatentransformation anschließen. Die Koordinatentransformation soll den Volumenkörper wieder in dem Koordinatenursprung positionieren. Erst jetzt wird der Volumenkörper mit dem Raummodell vereinigt, als Gesamtmodell gespeichert und anschließend mit dem Softwaresystem *VRED* visualisiert.

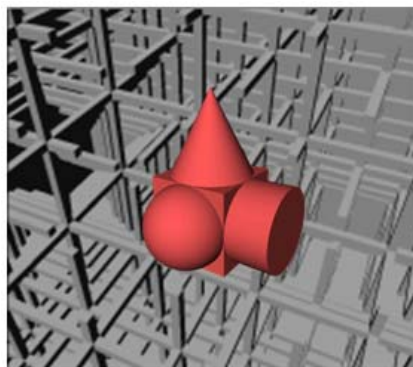
Nach der Visualisierung des Gesamtmodells wird eine Beurteilung vorgenommen. Die subjektive Einschätzung einer möglichen Steigerung der Raumwahrnehmung erfolgt anhand des Vergleichs zwischen dem Körper vor einem neutralen Hintergrund und dem Körper innerhalb des Raumes.

Als einfache Elementquerschnittsgeometrie für die Elemente wird zu Beginn ein Quadrat gewählt. Die Anordnung erfolgt zunächst in einem regelmäßigen Abstand. Bei einer ungünstigen Ausführung und Anordnung der Elemente in dem Würfel, kann es zu einer Verschachtelung der Elemente kommen (vgl. **Abbildung 4-43**).



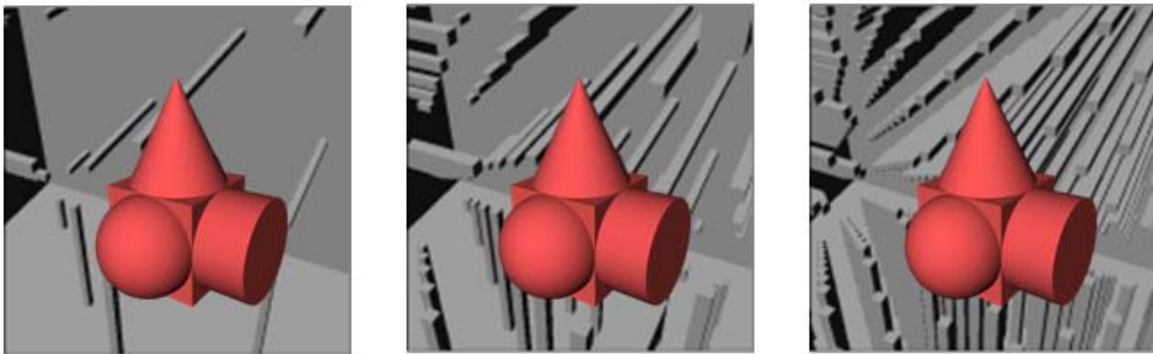
*Abbildung 4-43: Verschachtelung der Elemente*

Diese Verschachtelung erzeugt zwar einen räumlichen Eindruck, in Verbindung mit dem Objekt wird dessen Tiefenwirkung jedoch im Gesamtkontext nicht sehr gesteigert (vgl. **Abbildung 4-44**).



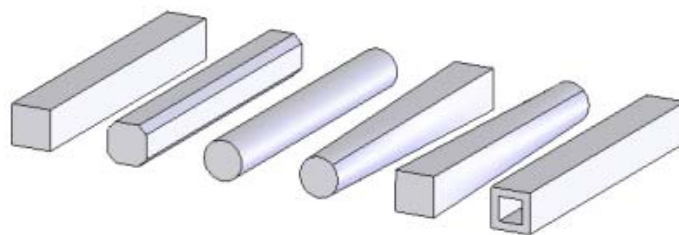
*Abbildung 4-44: Objekt und verschachtelte Elemente*

Daher wird die Elementanordnung so vorgenommen, dass eine Berührung untereinander ausgeschlossen ist. Die Elementanordnung erfolgt regelmäßig, um den Focus des Betrachters nicht durch Unregelmäßigkeiten des Hintergrundes vom Objekt abzulenken. Da eine gewisse Anzahl von Elementen vorhanden sein muss, um eine Tiefenwirkung zu erzeugen, darf der Elementabstand nicht zu groß gewählt werden. Sind die Elemente zu dicht nebeneinander angeordnet, wird ein Großteil der Würfelwand verdeckt und die Tiefenwirkung geht verloren. In **Abbildung 4-45** ist links eine zu geringe, rechts eine zu hohe und in der Mitte eine optimale Elementdichte zu erkennen.



*Abbildung 4-45: Elementdichte*

Für die Elementquerschnittsgeometrien werden nur symmetrische Geometrien verwendet, um sicherzustellen, dass der Betrachter aus unterschiedlichen Blickrichtungen einen annähernd gleichen Effekt beobachtet. In Voruntersuchungen [Raa05] zu dieser Arbeit wurden die Elementquerschnittsgeometrien (vgl. **Abbildung 4-46**) im Zusammenhang mit der Tiefenwirkung untersucht.



*Abbildung 4-46: Elementquerschnittsgeometrien [Raa05]*

Dort wurde festgestellt, dass die Elementquerschnittsgeometrie einen vernachlässigbaren Einfluss auf die Tiefenwirkung ausübt. Die quadratische Querschnittsgeometrie wird daher beibehalten. In **Abbildung 4-47** ist der aufgrund der gesammelten Erkenntnisse konzipierte Raum mit einer optimierten Anordnung und Gestaltung der Elemente zu erkennen.

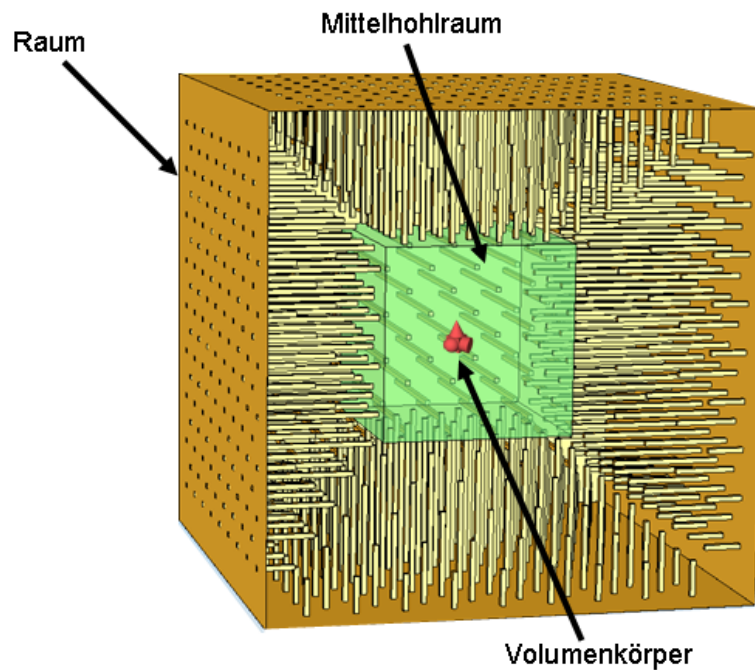


Abbildung 4-47: Raumgestaltung

Der im Zentrum vorhandene Mittelhohlraum ist so gestaltet, dass der Volumenkörper im Koordinatenursprung positioniert werden kann. Zum abschließenden Vergleich wird nun der Prototyp vor einem neutralen Hintergrund und in dem neu konzipierten Raum dargestellt (vgl. **Abbildung 4-48**).

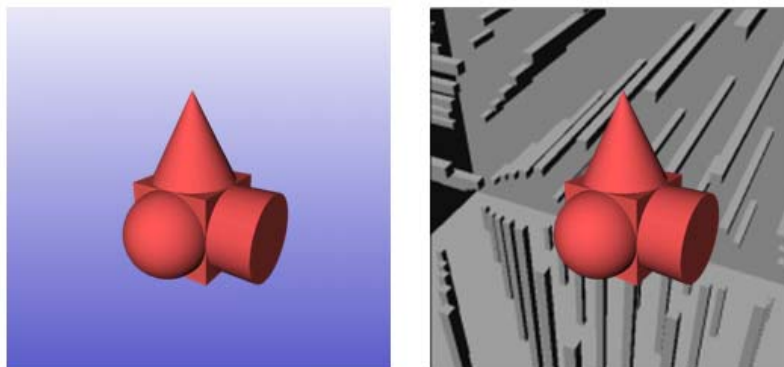


Abbildung 4-48: Objekt mit neutralem Hintergrund und mit neuem Raum

Im rechten Bild ist eine deutliche Steigerung der monokularen Tiefenwirkung gegenüber dem linken Bild zu erkennen. In der Abbildung 4-48 ist zu sehen, dass die Tiefenwirkung von virtuellen Prototypen mittels sphärischer Perspektive zu steigern ist.

#### 4.4.7 Fazit

Zum Thema der menschlichen Wahrnehmung existieren umfangreiche empirische Untersuchungen, welche die verschiedenen Einflüsse beschreiben, die zu einer räumlichen Wahrnehmung beitragen. Auf der Basis dieser gesicherten wissenschaftlichen Erkenntnis wurden einzelne Aspekte im Kontext der Virtual Reality aufgegriffen, um Optimierungspotenziale für die Darstellung von virtuellen Prototypen zu finden.

Es wurden Ansätze aufgezeigt, wie der Immersionsgrad bei der Darstellung von virtuellen Prototypen gesteigert werden könnte. Es haben sich Hinweise ergeben, wie mit einer Farboptimierung von Modell und Hintergrund eine Erhöhung der Tiefenwirkung zu erreichen wäre. Weitere Untersuchungen weisen darauf hin, dass definierte Materialeigenschaften ebenfalls einen Beitrag zur Erhöhung des Immersionsgrades leisten könnten. Ein möglicher neuer Lösungsansatz zur Steigerung des Immersionsgrades ist durch die Reduktion von Ghosting-Effekten mittels Farbmodulation vorgestellt worden. Darauf aufbauend wurde aufgezeigt, wie durch den Einsatz eines speziell gestalteten Raumes, in den der Prototyp importiert wird, eine zusätzliche Steigerung der Tiefenwirkung möglich wäre.

Die hier vorgestellten Ansätze zur Steigerung des Immersionsgrades wurden an Hand von Untersuchungen bewertet, die den Schluss nahe legen, dass die hier formulierten Maßnahmen zu einer deutlichen Steigerung des Immersionsgrades führen. Für eine Verifikation der in dieser Arbeit ausgewählten Maßnahmen sind weiterführende empirische Untersuchungen mit einer repräsentativen Anzahl von Probanden erforderlich. Ein statistisch belastbarer Nachweis der gefundenen Ansätze bildet jedoch nicht den Kern dieser Arbeit und würde den Umfang an dieser Stelle sprengen.

Aus den aufgeführten Untersuchungsergebnissen sind Empfehlungen für die Parametereinstellungen abgeleitet worden, die für eine optimale Darstellung der Modelle eine einfache Farbeinstellung bereits im CAD-System ermöglichen.

Die Prozesskette zur Aufbereitung der CAD-Daten und zur stereoskopischen Visualisierung der Gesamtszene musste jedoch stets manuell durchlaufen werden.

Daher wird in der Folge auf der Basis der erarbeiteten Parametereinstellungen die gesamte Prozesskette der Prototypenentstehung, ausgehend von dem CAD-System, weiter betrachtet, um den Gesamtprozess, angefangen vom Export der CAD-Daten bis zur Visualisierung der Szene zu automatisieren.

## 5 SOFTWARETECHNISCHE REALISIERUNG

Nach den Untersuchungen zur Steigerung des Immersionsgrades bei der stereoskopischen Darstellung von CAD-Objekten fließen die gewonnenen Ergebnisse in die anschließende softwaretechnische Realisierung des Konzeptes ein. Das Konzept sieht neben der Integrationsmöglichkeit in unterschiedliche CAD-Systeme eine weitestgehende Systemunabhängigkeit vor. Aus diesem Grund wird die Umsetzung in zwei Teilkonzepte untergliedert. In dem ersten Teil wird die Integration in ein CAD-System beschrieben. Der zweite Teil beinhaltet den Entwurf eines Visualisierungsmoduls, welches zum einen direkt aus dem CAD-System angesprochen wird, zum anderen aber auch eigenständig ohne den Einsatz eines CAD-Systems nutzbar ist. Bei der Konzeptumsetzung wird ein hoher Automatisierungsgrad des Visualisierungsprozesses angestrebt. Bevor die softwaretechnische Ausarbeitung erfolgt, wird die Architektur des Gesamtkonzeptes vorgestellt (vgl. **Abbildung 5-1**).

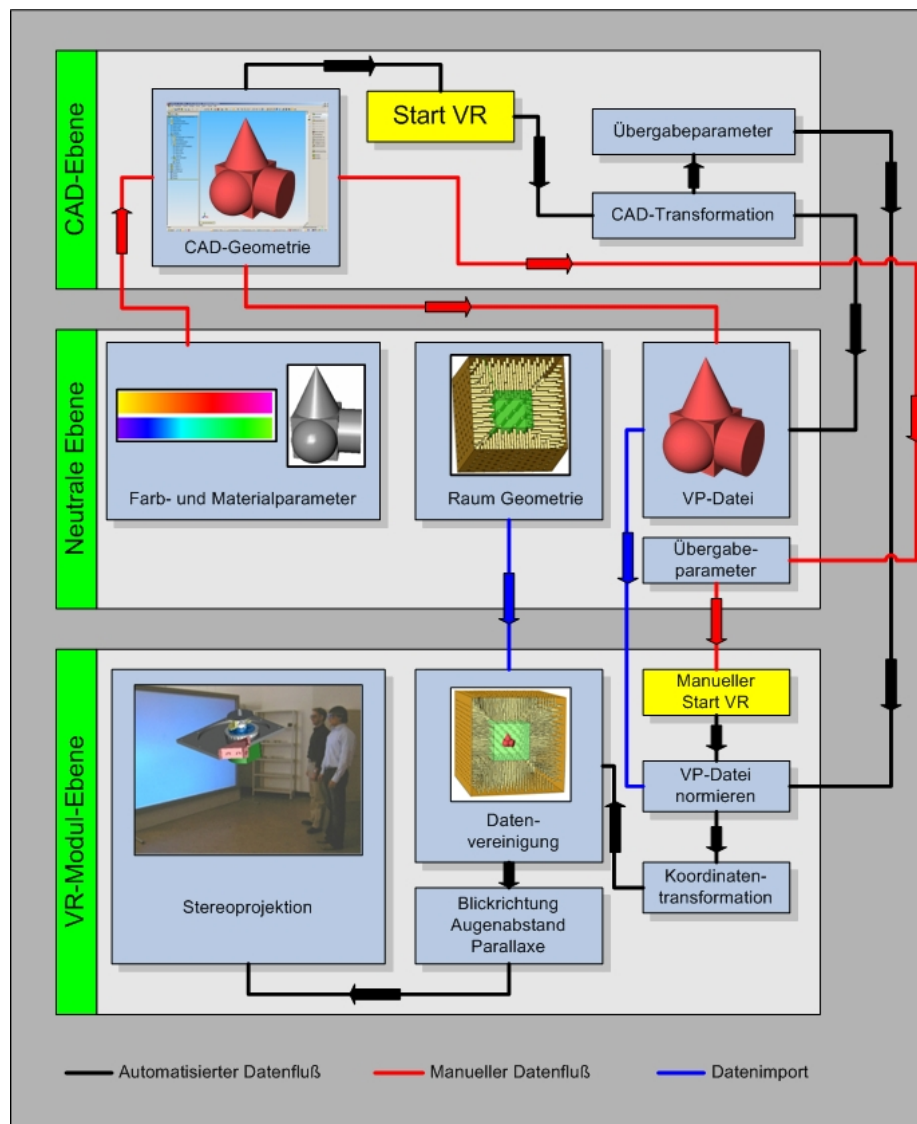


Abbildung 5-1: Architektur des Gesamtkonzeptes

Das Gesamtkonzept basiert auf einem Drei-Ebenen-Prinzip:

- CAD-Ebene
- Neutrale Ebene
- VR-Modul-Ebene

Die **CAD-Ebene** beinhaltet das CAD-System und notwendige Zusatzanwendungen zur Aktivierung eines automatisierten Visualisierungsprozesses. Das Konzept sieht vor, die Ergebnisse der Farb- und Materialparameter (**Neutrale Ebene**) den entsprechenden Bauteilen im CAD-System manuell zuzuweisen. Nach der Aktivierung einer im CAD-System implementierten Zusatzanwendung wird die Transformation der CAD-Objekte in ein systemneutrales Datenformat durchgeführt und als eigenständige VP-Datei auf die **neutrale Ebene** exportiert. Dann erfolgt die Ermittlung der Übergabeparameter. Die Übergabeparameter werden im Anschluss mit dem Startbefehl als Startparameter an das VR-Modul gesendet (**VR-Modul-Ebene**). Das VR-Modul importiert zunächst die neutrale VP-Datei und führt vor der Koordinatentransformation die Normierung durch. Nach dem Import der Raum-Geometrie in das VR-Modul werden beide Datensätze miteinander vereinigt. Vor der Darstellung werden Blickrichtung, Augenabstand und Parallaxe mit Vorgabewerten belegt. Zum Schluss wird die optimierte Gesamtszene stereoskopisch visualisiert. Damit ist das Drei-Ebenen-Prinzip automatisiert und wesentlich einfacher gestaltet als die im Abschnitt 2.7.3 aufgezeigte Visualisierungspipeline.

Der Einsatz des VR-Moduls ohne ein CAD-System benötigt einen VP-Datensatz, einen Datensatz der Raum-Geometrie und entsprechende Übergabeparameter. Der VP-Datensatz kann manuell aus einem CAD-System abgeleitet werden und muss als eigenständige Datei vorliegen. Zum Starten des VR-Moduls ist es notwendig in einem ebenfalls manuellen Prozessschritt die Übergabeparameter für den VP-Datensatz zu ermitteln. Das Konzept sieht weiterhin vor, dass Modul mit einer einfachen Kommandozeile zu starten, die den Namen der VR-Datei und die Startparameter beinhaltet. Sobald das Modul gestartet ist, geschieht der oben beschriebene Ablauf auf der **VR-Modul-Ebene** nach dem gleichen Ablaufschema.

Bevor auf die softwaretechnische Umsetzung des Konzeptes eingegangen wird, werden die grundlegenden Anforderungen an die Softwarearchitektur formuliert. Die Anforderungen sind nach Sommerville [Som07] in folgende Teilbereiche zu gliedern:

- Funktionale Anforderungen
- Nichtfunktionale Anforderungen
- Problembereichsanforderungen

- Benutzeranforderungen
- Systemanforderungen
- Schnittstellenspezifikation

Im Rahmen des Konzeptes werden zwei Softwarebausteine generiert, die über eine geeignete Schnittstelle miteinander kommunizieren. Daher müssen die Anforderungen an die Softwarebausteine teilweise in verschiedener Ausführung formuliert werden. Der erste Baustein beinhaltet die Integration einer Zusatzanwendung in ein CAD-System und der zweite Baustein enthält ein weitestgehend systemunabhängiges Visualisierungsmodul.

Mit den **funktionalen Anforderungen** werden einerseits die Inhalte der Funktionen des Softwaresystems beschrieben (*Was passiert wann?* oder *Wie lautet die spezielle Aufgabe?*) und andererseits müssen die dafür notwendigen Eingabe- und Ausgabemöglichkeiten exakt definiert werden, einschließlich eventuell auftretender Ausnahmesituationen. Die Spezifikationen der funktionalen Anforderungen an ein System müssen prinzipiell vollständig und auch konsistent sein. Wenn die Spezifikationen alle vom Benutzer benötigten Dienste festlegen, ist dies als vollständig zu bezeichnen. Die Konsistenz hingegen verlangt, dass die Anforderungen keine widersprüchlichen Festlegungen beinhalten. Die Systemfunktionen werden deshalb wie folgt festgelegt:

- CAD-Integration: Der CAD-Benutzer kann mithilfe eines Aktivierungsbuttons die stereoskopische Ausgabe der aktuell dargestellten CAD-Daten starten. Als Eingabewerte werden die CAD-Daten verwendet. Die Ausgabewerte sind die Übergabeparameter in Form von Bauteilabmessungen und die abgeleitete VP-Datei.
- VR-Modul: Das VR-Modul ist im Fall 1 aus dem CAD-System nach Button-Aktivierung automatisch zu starten. Im Fall 2 ist ein manueller Start des Moduls durch den Benutzer möglich. Als Eingabewerte sind jeweils die Übergabeparameter, die VP-Datei und die Raumgeometrie zu nutzen. Die stereoskopische Visualisierung kennzeichnet den Ausgabewert.

Die **nichtfunktionalen Anforderungen** ergeben sich aus den Beschränkungen für die durch das zum Einsatz kommende Hard- und Software-System angebotenen Dienste und Funktionen. Diese beziehen sich in erster Linie auf die Zuverlässigkeit, den Speicherbedarf, die Benutzerfreundlichkeit, die Wartbarkeit, die Portierbarkeit und die Leistungsfähigkeit von E/A-Geräten und die durch die Systemschnittstelle genutzten Datendarstellungen. Im Einzelnen sind dies:



- Die Zuverlässigkeit und die Verfügbarkeit des Systems müssen möglichst zu jeder Zeit sichergestellt sein. Der Speicherbedarf wird durch die Größe des zu visualisierenden Modells bestimmt.
- Die Benutzerfreundlichkeit zeichnet sich durch eine sehr einfache Bedienung aus.
- Für die Weiterentwicklung des Konzeptes spielt die Wartbarkeit eine wichtige Rolle. Die Wartbarkeit darf nicht nur das Beheben von Fehlern in möglichst kurzer Zeit und das Einbinden der jeweils aktuellen Technologien ermöglichen, sondern muss durch eine klar gegliederte Struktur des Programms die Einsparungen von Zeit und Kosten gewährleisten.
- Die Portierbarkeit kann für das CAD-System nicht formuliert werden, da das CAD-System häufig nicht systemneutral ausgelegt ist. Die Portierbarkeit ist jedoch durch eine neutrale Datenebene und ein weitestgehend systemneutrales VR-Modul zu gewährleisten. Auf der neutralen Datenebene sollen die Farb- und Materialparameter, die Raumgeometrie und die VP-Daten zur Verfügung stehen. Bei dem manuellen Einsatz werden zudem die Übergabeparameter auf der neutralen Datenebene hinterlegt. Das heißt, es sollen keine plattformabhängigen Anweisungen bzw. Daten in der neutralen Ebene oder in der VR-Modul-Ebene zum Einsatz kommen.
- Die Datendarstellungen der Farb- und Materialparameter können tabellarisch in Papierform oder elektronisch aufgelistet sein. Die Dokumentation der Übergabeparameter kann in der gleichen Form erfolgen. Die 3D-Datensätze der Raum-Geometrie und des VP-Datensatzes liegen in der Form einer VRML-Datei im ASCII-Format vor.
- Die Leistungsfähigkeit von E/A-Geräten des Systems ist eng mit der Latenzzeit verknüpft. Deshalb muss diese so ausgelegt sein, dass keine negativen Einflüsse auf den Immersionsgrad wirken.

Die Komponenten wie Rechner und Projektoren unterliegen einer rasanten Weiterentwicklung. Daher wird an dieser Stelle keine Festlegung auf Systemkomponenten (vgl. Abschnitt 4.2) vorgenommen. Da die Größe des zu visualisierenden Modells die Rechnerkonfiguration bestimmt, ist die ausgewählte Rechnerkonfiguration mithilfe eines Benchmarks dahingehend zu testen, bei welcher Polygonanzahl eine flüssige Echtzeitdarstellung der Modelle nicht mehr gegeben ist. Es ist bei der Darstellung der Modelle daher darauf zu achten, dass die vorher bestimmte maximale Anzahl an Polygonen stets unterschritten wird.

Die **Problembereichsanforderungen** beschreiben die internen Anforderungen, die sich aus der Anwendung des Systems ergeben und deren Inhalte unbedingt in dem noch zu entwickelnden Konzept Beachtung finden müssen. Wenn diese Anforderungen nicht erfüllt

werden, ist das System möglicherweise nicht, oder nicht zufrieden stellend einsetzbar. Hier ist die Systemunabhängigkeit zu nennen, das heißt, dass Ergebnis des Konzeptes erfordert es, dass es an möglichst viele Betriebssysteme und Softwaresysteme koppelbar ist.

Die **Benutzeranforderungen** beschreiben die externen Anforderungen für die Anwender des Systems hinsichtlich der Nutzung des Systems. Hierbei sind die Auswirkungen für den Benutzer, die sich durch das jeweilige Verhalten des Systems in den speziellen Bearbeitungssituationen ergeben, so allgemeinverständlich wie möglich zu formulieren:

- Alle notwendigen Parametereinstellungen sind mit der vorhandenen CAD-Funktionalität vorzunehmen. In dem eingesetzten CAD-System ist ein Aktivierungsbutton zu implementieren, damit ein automatisierter Visualisierungsprozess der CAD-Daten ausgelöst werden kann.
- Zum Start des manuellen Visualisierungsprozesses ist nur die Eingabe einer Kommandozeile notwendig.
- Die Raum-Geometrie muss frei wählbar sein. Sie muss jedoch unbedingt den in Abschnitt 4.4.6 definierten Anforderungen entsprechen.

Die **Systemanforderungen** sind erweiterte Varianten der Benutzeranforderungen und stellen die Grundlage für den Systementwurf dar. Sie ergänzen die Benutzeranforderungen mit Einzelheiten und erklären, wie diese vom System umzusetzen sind. Bei dem geplanten Systementwurf werden die zwei Teilbereiche CAD-Integration und VR-Modul getrennt voneinander betrachtet.

- CAD-Integration: Nach dem Betätigen des Aktivierungsbuttons wird eine Zusatzanwendung gestartet. Diese transformiert zunächst die aktuelle CAD-Geometrie in ein VRML-Modell, dessen Inhalt anschließend als eigenständige Datei in einen Ordner XY exportiert wird. Im Anschluss werden die kubischen Abmaße der aktuellen CAD-Geometrie ermittelt und als Zahlenwerte in einem Speicher hinterlegt. Mit dem Starten eines externen Programms werden die im Speicher hinterlegten Werte simultan als Startparameter an das zu startende VR-Modul übergeben.
- VR-Modul: Nach dem Start des VR-Moduls mit dem Startbefehl aus der CAD-Zusatzanwendung werden zunächst mithilfe der übergebenen Abmessungen des CAD-Objektes die Skalierungs- und die Koordinatentransformationsparameter berechnet. Die VP-Datei wird aus dem Ordner XY eingelesen und um die neu berechneten Parameter ergänzt. Danach wird die Raum-Datei eingelesen und an die

VP-Datei angehängt. Zuletzt wird die VP-Datei mit der Raum-Datei als vereinigte Datei stereoskopisch visualisiert. Die manuelle Aktivierung des Moduls unter Verwendung der aus dem CAD-System ermittelten Abmaße des CAD-Objektes wird mittels einer Eingabekonsole mit einer einfachen Kommandozeile gestartet. An die Kommandozeile müssen die Abmaße des CAD-Objektes angehängt werden. Der sich anschließende Prozess läuft dann wie bereits beschrieben ab.

Für das Zusammenarbeiten von verschiedenen Softwaresystemen müssen präzise definierte **Schnittstellenspezifikationen** existieren. Für die Kommunikation zwischen der CAD-Implementierung und dem VR-Modul wird die Schnittstelle folgendermaßen spezifiziert:

Die aus der CAD-Geometrie ermittelten Parameter werden beim Aufruf des VR-Moduls (Programms) aus der CAD-Implementierung auf Betriebssystemebene übergeben. Die Parameter werden dabei nach dem Programmnamen aufgeführt. Diese Parameter werden als Kommandozeilenparameter bezeichnet und als Zeichenfelder (*C-String*) übergeben [Bre97]. Folgende Parameter sind vorgesehen:

- Der Name des zu startenden VR-Moduls
- Ausdehnung der CAD-Geometrie in X-Richtung
- Ausdehnung der CAD-Geometrie in Y-Richtung
- Ausdehnung der CAD-Geometrie in Z-Richtung
- X-Wert der Ursprungskoordinate des CAD-Objektes
- Y-Wert der Ursprungskoordinate des CAD-Objektes
- Z-Wert der Ursprungskoordinate des CAD-Objektes

Nach der Übergabe der Parameter erfolgt die Weiterverarbeitung im VR-Modul.

Nachdem die Anforderungen an die Softwarearchitektur allgemein formuliert sind, kann nun die softwaretechnische Umsetzung der CAD-Implementierung am Beispiel des CAD-Systems *SolidWorks* vorgenommen werden.

## 5.1 CAD-Integration

Um den Funktionsumfang zu erweitern, bietet *SolidWorks* die Möglichkeit Zusatzanwendungen zu erstellen. Aus einer für *SolidWorks* erzeugten Zusatzanwendung können Befehle auf Betriebssystemebene abgesetzt werden. Die dafür notwendigen

Parameter sind den Befehlen als Kommandozeilenparameter angehängt. Damit erfüllt *SolidWorks* die im Abschnitt 3.2 formulierten Anforderungen an die Erstellung von Zusatzanwendungen.

Einfache Zusatzanwendungen sind mithilfe von *SolidWorks-Makros* zu erzeugen. Alle Funktionen, die über die Benutzeroberfläche erreichbar sind, können mit *SolidWorks-Makros* ausgeführt werden. Makros stellen so eine Möglichkeit zur Verfügung, um wiederkehrende Tätigkeiten, die über die Benutzeroberfläche zu aktivieren sind, aufzuzeichnen und zu einem späteren Zeitpunkt als automatisierte Routine auszuführen. Dadurch wird der manuelle Aufwand erheblich reduziert. Das bedeutet aber, dass *SolidWorks-Makros* nur sequenzielle Prozesse ausführen können, die durch die Aktivierung von Funktionen der Bedienoberfläche aufgezeichnet wurden. Mit aufgezeichneten Makros sind also nur Prozesse zu automatisieren, für die auch eine entsprechende Funktion in der Bedienoberfläche vorhanden ist. Insbesondere komplexe Routine, bei denen unter anderem ein Zugriff auf die interne Datenstruktur des CAD-Systems notwendig ist, können mit Makros nicht erstellt werden. Für diese speziellen Zusatzanwendungen bietet *SolidWorks* ein *Application-Programming-Interface (API)* an, welches auf der COM<sup>10</sup>-Technologie basiert.

Dabei handelt es sich nicht um ein spezielles Softwaremodul, sondern um ein von Microsoft entwickeltes Standardverfahren. Die COM-Technologie ist ein Protokoll, welches ein Softwaremodul mit einem anderen Softwaremodul verbindet und sich danach zurückzieht. Nachdem eine Verbindung aufgebaut wurde, kommunizieren die beiden Module anschließend über Schnittstellen miteinander. Die COM-Kommunikation basiert auf einem Client- / Server-Prinzip. Das erstellte Objekt wird als Server bezeichnet und stellt die angebotenen Funktionen zur Verfügung. Das Programm, welches die Funktionen des COM-Objektes nutzt, wird Client genannt. Mit diesem Prinzip wird eine Wiederverwendbarkeit der Syntax und eine Reduzierung der Entwicklungszeit gewährleistet.

Mit der *SolidWorks-API* können somit individuelle Zusatzanwendungen entworfen werden, die Zugriff auf die interne Datenstruktur des CAD-Systems haben. Die API-Funktionen können von Visual Basic 6.0, **Visual Basic for Applications (VBA)**, Visual Basic .NET, Visual C# .NET, Visual C++ 6.0, Visual C++ .NET und *SolidWorks* Makrodateien aufgerufen werden. Für die API-Funktionen steht eine umfassende Dokumentation zur Verfügung.

Die Erstellung der CAD-Implementierung für das Softwarekonzept sieht eine möglichst schnelle und einfache Umsetzung der definierten Anforderungen vor. Daher ist zu überlegen, wie die Anforderungen mit einem möglichst geringen Aufwand umzusetzen sind. Eine einfache und elegante Lösungsvariante zur Erstellung einer Zusatzanwendung, ist das bereits genannte Aufzeichnen von Makros. Bei der Aufzeichnung von *SolidWorks-Makros* wird der Quellcode automatisch für die Programmiersprache Visual Basic generiert. Der Vorteil liegt hier in einer erheblichen Verkürzung der Entwicklungszeit. Ein weiterer Vorteil

---

<sup>10</sup> COM - Component Object Model

liegt darin begründet, dass *SolidWorks-Makros* leicht per Drag & Drop in die Benutzeroberfläche von *SolidWorks* in Form eines Aktivierungsbuttons einzufügen sind.

Diesen Vorteilen steht der Nachteil gegenüber, dass nur Funktionen aufzuzeichnen sind, die in der Bedienoberfläche betätigt werden. Es ist jedoch möglich, dem Grundgerüst eines *SolidWorks-Makros* weitere Funktionen aus der API in Form von Visual Basic - Anweisungen hinzuzufügen. Mit dieser Systematik ist eine schnelle und einfache Grundstruktur der notwendigen Syntax zu erstellen, ohne auf die Potentiale der API verzichten zu müssen. Diese Vorgehensweise wird für den weiteren Verlauf zur Konzeption der CAD-Implementierung genutzt.

Da die *SolidWorks-API* direkt von *SolidWorks-Makros* angesprochen werden kann, ohne eine COM-Client-Server-Architektur erstellen zu müssen, wird hier auf eine Vertiefung der COM-Technologie verzichtet.

VBA ist ein Basic Dialekt, der von Microsoft zur Steuerung von Programmabläufen für Microsoft-Office-Komponenten entwickelt wurde. Die VBA-Syntax ist zu einem Großteil mit der Syntax von Visual Basic identisch, VBA ist jedoch gegenüber Visual Basic in seinen Funktionen und in der Leistungsfähigkeit seiner Entwicklungsumgebung eingeschränkt. Da VBA keine objektorientierten Konzepte wie Vererbung und Polymorphie unterstützt, wird VBA vor allem für die prozedurale Programmierung eingesetzt. Mit Hilfe der Methode `CreateObject()` können COM-Verbindungen zu registrierten COM-Servern hergestellt werden. Dadurch besteht die Möglichkeit Funktionalitäten fremder Programme nutzen zu können. Bei diesem Konzept wird die Kommunikation mit der *SolidWorks-API* über diese Methode hergestellt. Im Folgenden wird auf die VBA spezifische Notation jedoch nicht weiter eingegangen. Mit dem VBA-Editor (vgl. **Abbildung 5-2**) ist eine komfortable Bearbeitung der *SolidWorks-Makros* möglich.

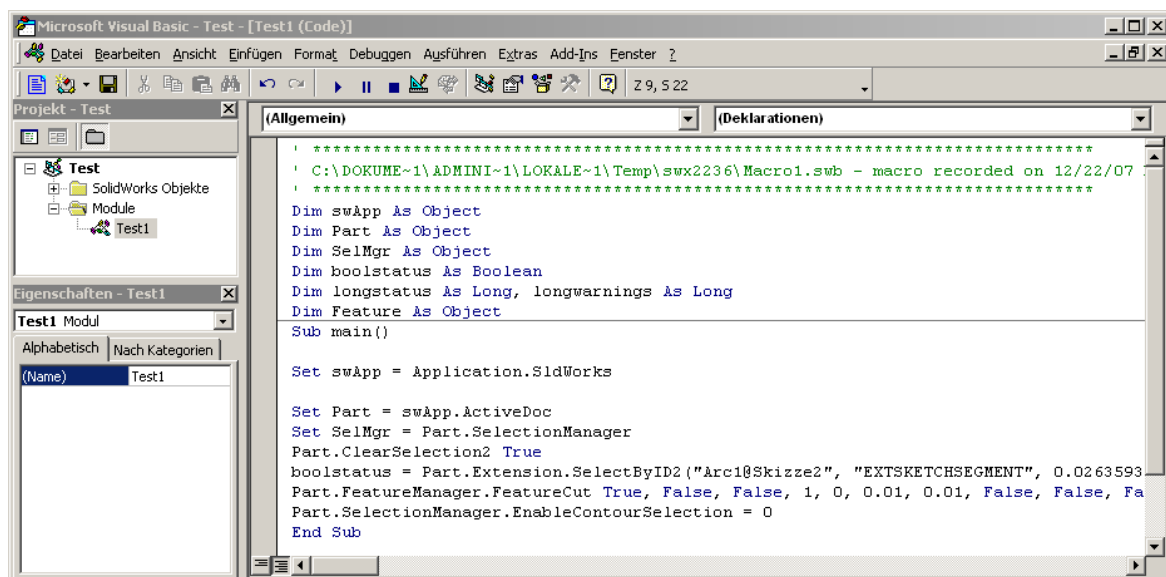


Abbildung 5-2: VBA-Editor

Neben dem Erstellen von Programmcode ist es mit dieser Entwicklungsumgebung auch möglich, Bedienoberflächen (Formulare) zu gestalten und in das VBA-Projekt zu integrieren. Das VBA-Entwicklungssystem kompiliert (übersetzt) den erstellten Programmtext anschließend in eine für den Computer verarbeitbare Form. Diese werden dann als VBA-Projektdateien (Makro-Dateien) auf der Systemebene gespeichert.

Basierend auf der beschriebenen Vorgehensweise zur Erstellung und Bearbeitung von SW-Makros wird im folgenden die Umsetzung des Programmmoduls beschrieben. Dabei werden detailliert die Teilfunktionen

- Ermittlung der Übergabeparameter
- Export von CAD-Daten
- Start des VR-Moduls

vorgestellt. Die Beschreibung dieser Modulararchitektur wird dahingehend begrenzt, dass zum Verständnis der Ablaufroutine nur für die wichtigsten Funktionen und Abläufe eine nähere Erläuterung erfolgt. Es wird davon abgesehen, die VBA-Notation und die Syntax in allen Einzelheiten, wie beispielsweise die Deklaration von Variablen darzustellen, da diese Programmstruktur in ähnlicher Weise auch für andere CAD - API's und in anderen Programmiersprachen umsetzbar sein muss. Zur Erklärung werden jedoch Programmabschnitte auszugsweise herangezogen.

Die Programmstruktur ist folgendermaßen gegliedert:

- Deklaration der Variablen
- Ermittlung der minimalen und der maximalen Begrenzungen des aktuellen Objektes in X-, Y- und Z-Richtung
- Lage des Objektmittelpunktes bezogen auf den Koordinatenursprung
- Bestimmung der Längenausdehnung des Objektes in X-, Y- und Z-Richtung
- Umwandlung der ermittelten Zahlenwerte in Zeichenketten
- Speichern der *VRML*-Datei
- Start des VR-Moduls und Übergabe der Parameter

Nachdem die für das Ausführen einer Makrodatei notwendigen globalen Variablen deklariert wurden, werden zu Beginn die X-, Y- und Z-Begrenzungen des aktuellen *SolidWorks*-Objektes mit einer For-Next-Schleife ermittelt (vgl. **Abbildung 5-3**).

```

' -----
' Ermittlung der minimalen und maximalen Begrenzungen
' in X, Y und Z-Richtung des aktuellen Objektes
' -----
vChild = swRootComp.Children
For i = 0 To UBound(vChild)
    Set swChildComp = vChild(i)

    If swChildComp.Visible = swComponentVisible Then
        vBox = swChildComp.GetBox(False, False)
        X_max = GetMax(vBox(0), vBox(3), X_max)
        X_min = GetMin(vBox(0), vBox(3), X_min)
        Y_max = GetMax(vBox(1), vBox(4), Y_max)
        Y_min = GetMin(vBox(1), vBox(4), Y_min)
        Z_max = GetMax(vBox(2), vBox(5), Z_max)
        Z_min = GetMin(vBox(2), vBox(5), Z_min)
    End If
Next i

```

*Abbildung 5-3: Ermittlung der Objektausdehnungen*

Anschließend liefert die Objektreferenz „**swRoot.Comp.Children**“ das Objekt „**vChild**“. Dieses Objekt bietet Zugriff auf alle in einer Baugruppe enthaltenen Komponenten, wobei es sich wiederum um Baugruppen oder Teile handeln kann. Die folgende Schleife wird solange durchlaufen, bis alle Objekte untersucht wurden. Die Funktion „**Set**“ weist der Variablen „**swChildComp**“ das momentan zu untersuchende Objekt zu. Wenn dieses Objekt sichtbar ist, liefert die Methode „**swChildComp.GetBox**“ die kubische Begrenzung dieser Komponente. Während der Variablendeklaration sind zu Beginn des Programmdurchlaufes den drei Raumkoordinaten X, Y und Z die

Maximalwerte 1.79769313486231E+308 und die

Minimalwerte -1.79769313486231E+308

zugewiesen. Diese Maximal- bzw. Minimalwerte werden mit dem jeweiligen Begrenzungswert der Koordinatenachse an die Funktionen „**GetMax**“ und „**GetMin**“ übergeben (vgl. **Abbildung 5-4**).

```

'-----
' Findet das Maximum von 3 Werten
'-----
Function GetMax _
(ByVal Val1 As Double, ByVal Val2 As Double, ByVal Val3 As Double) As Double

    GetMax = Val1
    If Val2 > GetMax Then
        GetMax = Val2
    End If
    If Val3 > GetMax Then
        GetMax = Val3
    End If

End Function

'-----
' Findet das Minimum von 3 Werten
'-----
Function GetMin _
(ByVal Val1 As Double, ByVal Val2 As Double, ByVal Val3 As Double) As Double
' Finds minimum of 3 values
    GetMin = Val1

    If Val2 < GetMin Then
        GetMin = Val2
    End If

    If Val3 < GetMin Then
        GetMin = Val3
    End If

End Function

```

Abbildung 5-4: Bestimmung von Max. bzw. Min. dreier Werte

Diese Funktionen liefern nach einem Vergleich der drei Werte den größten oder kleinsten Wert als Antwort. Diese Antwort wird dann der entsprechenden Begrenzungsvariablen zugewiesen (vgl. Abbildung 5-3). Durch diesen Algorithmus ist es möglich, die Begrenzungen in den jeweiligen Raumrichtungen zu ermitteln. Mithilfe dieser Werte sind nun die Mittelpunktkoordinaten der kubischen Begrenzung und die Längenausdehnungen für die drei Raumrichtungen der globalen Baugruppe zu bestimmen (vgl. **Abbildung 5-5**).

```

'-----
' Bestimmung der Mittelpunktkoordinaten für die kubische
' Begrenzung des aktuellen Objektes bezogen auf den
' Koordinatenursprung
'-----
CenterX = Center(X_max, X_min) 'Mittelpunkt auf X-Ebene
CenterY = Center(Y_max, Y_min) 'Mittelpunkt auf Y-Ebene
CenterZ = Center(Z_max, Z_min) 'Mittelpunkt auf Z-Ebene

'-----
' Bestimmung der Längenausdehnungen des aktuellen
' Objektes in X, Y und Z-Richtung
'-----
LaengeX = Laenge(X_max, X_min) 'Baugruppenlänge auf X-Ebene
LaengeY = Laenge(Y_max, Y_min) 'Baugruppenlänge auf Y-Ebene
LaengeZ = Laenge(Z_max, Z_min) 'Baugruppenlänge auf Z-Ebene

```

Abbildung 5-5: Bestimmung der Mittelpunktkoordinaten und der Längenausdehnung



Mit der Werteübergabe für die Objektbegrenzungen der jeweiligen Raumrichtungen wird durch die Funktion „**Center**“ die Mittelpunktkoordinate für die entsprechende Raumrichtung berechnet. Der berechnete Wert wird dann den Variablen für die Mittelpunktbeschreibung zugewiesen. Mit der Funktion „**Laenge**“ wird gleichermaßen verfahren. Die hier berechneten Werte werden ebenfalls den Variablen für die Beschreibung der Längenausdehnungen in den drei Raumrichtungen zugewiesen. Damit stehen alle benötigten Parameter für die Übergabe zur Verfügung. Die Übergabeparameter liegen momentan als Datentyp *double*<sup>11</sup> vor. Laut der Schnittstellenspezifikation müssen diese jedoch als Zeichenkette zur Parameterübergabe vorliegen. Die Umwandlungsfunktion „**DoubletoString**“ wandelt die benötigten Parameter vom Datentyp *double* in den Datentyp *string*<sup>12</sup> um (vgl. **Abbildung 5-6**).

```

'-----
' Umwandlung der Double-Werte zu einem
' String mit der Funktion DoubletoString
'-----
' Mittelpunktkoordinaten (X, Y, Z)
CenterX_str = DoubletoString(CenterX)
CenterY_str = DoubletoString(CenterY)
CenterZ_str = DoubletoString(CenterZ)
' Längenwerte (X, Y, Z)
LaengeX_str = DoubletoString(LaengeX)
LaengeY_str = DoubletoString(LaengeY)
LaengeZ_str = DoubletoString(LaengeZ)

```

Abbildung 5-6: Datentypumwandlung

Jetzt entsprechen die Parameter der Schnittstellenspezifikation. In dem nächsten Schritt wird das gesamte CAD-Objekt als *VRML*-Datei auf dem Laufwerk *C* im Ordner *stereo* mit dem Namen *solid.WRL* gespeichert (vgl. **Abbildung 5-7**).

```

'-----
' Speichern des CAD Datensatzes als VRML Datei
'-----
swModel.SaveAs2 "c:\stereo\solid.WRL", 0, True, False

```

Abbildung 5-7: VRML-Exportbefehl

Zum Abschluss der CAD-Integration muss der Startbefehl auf Systemebene mit den Übergabeparametern für den Start des VR-Moduls abgesendet werden. Um ein

<sup>11</sup> Bezeichnung für einen Gleitkommawert, welcher acht Byte belegt.

<sup>12</sup> Bezeichnung für eine Folge von Zeichen aus einem definierten Zeichensatz.

ausführbares Programm auf Systemebene zu starten, bietet VBA eine Shell<sup>13</sup>-Funktion an. Der Shell-Funktion sind die notwendigen Argumente anzuhängen (vgl. **Abbildung 5-8**).

```
'-----  
' Start des VR-Moduls und Übergabe der Parameter  
'-----  
Ergebnis = Shell("C:\stereo\Stereo-Projekt.EXE  
c:\stereo\solid.WRL" & " " & LaengeX_str & " " &  
LaengeY_str & " " & LaengeZ_str & " " & CenterX_str &  
" " & CenterY_str & " " & CenterZ_str, 1)
```

Abbildung 5-8: Start des VR-Moduls

Dabei ist der Programmpfad bei dieser prototypischen Umsetzung vorerst festgelegt. Bei erfolgreichem Start des VR-Moduls liefert die Shell-Funktion die Task-ID<sup>14</sup> des gestarteten Programms zurück. Andernfalls wird eine Null zurückgeliefert. Um auf die CAD-Ebene zurückzugelangen, muss das VR-Modul zuvor zwingend beendet sein. Die zurückgesendeten Werte der Task-ID können in der *SolidWorks*-Integration für weitere Programmschritte Verwendung finden. Dies ist bei diesem Konzept nicht vorgesehen. Die erstellte Syntax wird nachfolgend als Makrodatei gespeichert und steht dem CAD-System *SolidWorks* als Integrationsmodul zur Verfügung. Analoge Zusatzanwendungen können auch in ähnlicher Weise für andere CAD-Systeme mit einem vergleichbaren Entwicklungsaufwand realisiert werden. Das erstellte *SolidWorks*-Makro wird mit einem Menübutton verknüpft und so in die Bedienoberfläche von *SolidWorks* implementiert (vgl. **Abbildung 5-9**).

---

<sup>13</sup> Eine Shell-Funktion ermöglicht das Interpretieren und Ausführen von Kommandos, beispielsweise das Starten eines Programms.

<sup>14</sup> Die Task-ID ist eine Zahl, die das ausgeführte Programm eindeutig kennzeichnet.

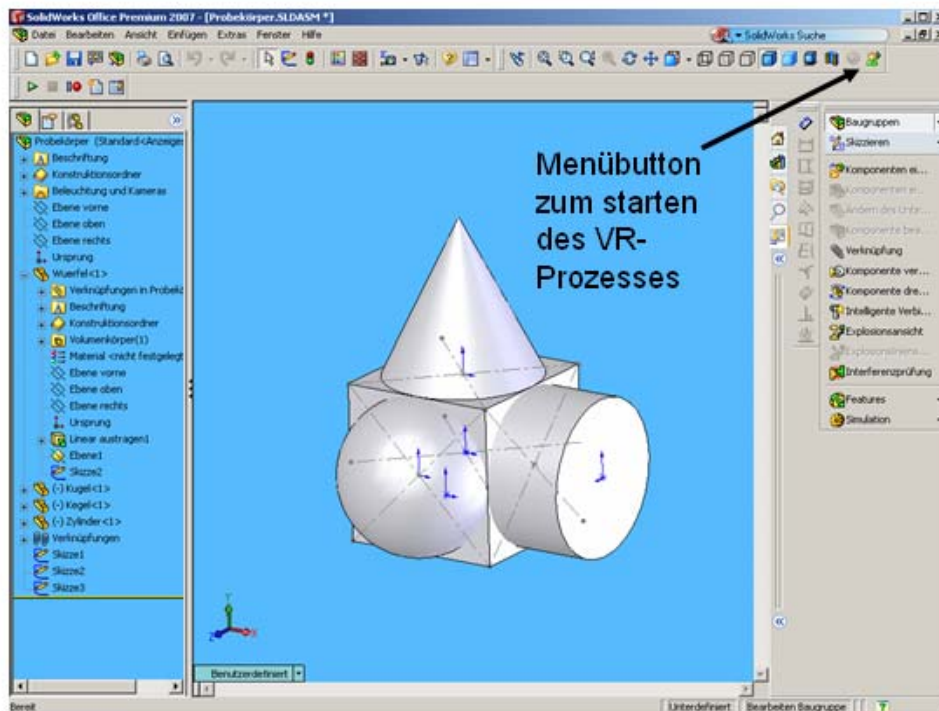


Abbildung 5-9: Menübutton zum Starten des VR-Prozesses

Beim Betätigen des Buttons wird die CAD-Implementierung gestartet. Nach dem Durchlauf der Programmroutine wird das VR-Modul aktiviert und die erforderlichen Parameter übergeben. Das VR-Modul führt im Anschluss die Prozesskette fort. Im nächsten Abschnitt wird nun die softwaretechnische Umsetzung des VR-Moduls mithilfe eines Szenengraphensystems beschrieben.

## 5.2 VR-Modul

Im vorangegangenen Abschnitt ist die Umsetzung einer CAD-Implementierung zur automatisierten Ableitung von CAD-Objekten am Beispiel des CAD-Systems *SolidWorks* dargestellt worden. Bei der Aktivierung des VR-Prozesses im CAD-System liefert die neue Zusatzanwendung einen systemneutralen Datensatz der Objektgeometrie für eine neutrale Ebene. Gleichzeitig stehen die benötigten Übergabeparameter zur Verfügung. Diese Daten dienen als Eingangsgrößen für die nächsten Prozessschritte. Es gilt nun ein Konzept für die VR-Modul-Ebene zu gestalten, welches die neutralen Objektdaten mithilfe der Übergabeparameter stereoskopisch darstellt. Das VR-Modul muss dabei die in Abschnitt 3.4 definierten Anforderungen erfüllen.

Die stereoskopische Visualisierung erfolgt über die vorhandene Grafikhardware. Diese Hardwareebene ist mit Computergrafikalgorithmen anzusprechen, um die gewünschte Bildausgabe zu erhalten. Diese Computergrafikalgorithmen werden heute durch moderne

Softwareschichten in Form von Grafik-API's als Bibliotheken gekapselt und ermöglichen damit einen Zugriff auf die Funktionalitäten der Grafikkhardware durch hardwarenahe Schnittstellen.

Die zwei meistgenutzten Grafik-API's sind *Microsoft Direct3D* und *OpenGL*. Mit diesen API's ist es Programmierern möglich, Grafikanwendungen zu entwerfen. *Direct3D* zeichnet sich dadurch aus, eine hervorragende Unterstützung jeder neuen Grafikkarte zu leisten. Da *Direct3D* jedoch eng mit der Windows-Plattform verknüpft ist, kann *Direct3D* deshalb nicht plattformunabhängig zum Einsatz kommen. Gegenüber *Direct3D* basiert die Bibliothek *OpenGL* auf der Graphics Library GL des Herstellers Silicon Graphics. Nachdem *OpenGL* von Microsoft als integraler Bestandteil der verschiedenen Windows-Versionen ausgeliefert wurde, hat sich für *OpenGL* ein enormer Markt eröffnet. *OpenGL* kann als weitestgehend systemunabhängig und portabel bezeichnet werden, da *OpenGL* neben *Windows* für *MAC-OS*, *Linux* und alle kommerziellen *UNIX*-Systemen zur Verfügung steht.

Mit Grafik-API's lassen sich einzelne grafische Objekte sehr gut darstellen. VR-Applikationen müssen jedoch häufig überaus viele Objekte in einem engen Zusammenhang untereinander visualisieren, so dass eine Umsetzung mit Hilfe einer Grafik-API nicht sinnvoll ist. Aus diesem Defizit heraus wurde eine neue Abstraktionsebene zwischen der VR-Applikation und der Grafik-API geschaffen (vgl. **Abbildung 5-10**).

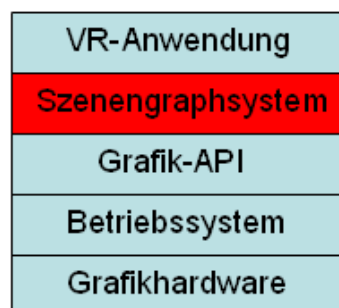


Abbildung 5-10: Grafik-Abstraktionsebenen

Diese neue Abstraktionsebene speichert Hierarchien die mit gerichteten azyklischen Bäumen (directed acyclic graph, **DAG**) die Objekte einer darzustellenden Szene beschreiben. Für aktuelle API's hat sich diese Datenstruktur als Standard etabliert. Dieses Konzept für eine Datenstruktur wird Szenengraphsystem genannt.

Ein Szenengraphsystem führt die im DAG gespeicherten Operationen aus, bildet diese auf der darunter liegenden Grafik-API ab und stellt die Objekte mit Hilfe der Grafik-API dar. In **Abbildung 5-11** ist ein einfacher Aufbau eines Szenengraphen zu erkennen.

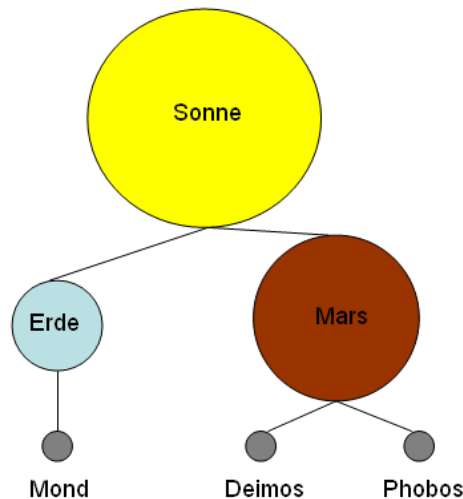


Abbildung 5-11: Abstrahiertes Sonnensystem

Erde und Mars müssen in einem unterschiedlichen Abstand und mit einer anderen Rotationsgeschwindigkeit um die Sonne rotieren. Ebenso müssen die Monde von Erde und Mars jeweils um den Planeten und gleichzeitig um die Sonne rotieren. Wird also ein Planet rotiert, muss sich die Rotation auch auf den dazugehörigen Mond übertragen. Solche Hierarchien sind mit DAG's relativ einfach zu realisieren. In Knoten liegt die entsprechende Geometrie der Planeten mit der dazugehörigen Transformation, die auf den Planeten anzuwenden sind. Diese Transformation wird dann auch auf die entsprechenden Kindknoten der Monde angewendet und damit weitervererbt. So sind in einer Szene Objekte sehr effizient wiederverwendbar.

Beispiele für Szenengraphen sind *OpenInventor*, *Java3D*, *VRML97* und *OpenSG*. Diese Szenengraphsysteme werden neben der Erstellung von Echtzeit-Graphikprogrammen unter anderen für Anwendungen der Virtuellen Realität eingesetzt.

*OpenSG* ist ein Open Source (**LG**PL - Lesser **G**eneral **P**ublic **L**icense) Echtzeit-Rendering-System, welches auf *OpenGL* basiert und es zeichnet sich besonders dadurch aus, dass es multiviewport- bzw. clusterfähig ist. Dies bedeutet, dass mit *OpenSG* auf unterschiedlichen Computern mit verschiedenen Grafikkarten die gleiche Anwendung laufen kann. Insbesondere bei dem Einsatz von High-End VR-Systemen ist der Einsatz eines Clusters von besonderer Bedeutung, da beispielsweise bei der Nutzung einer CAVE mehrere Ausgabekanäle synchron anzusprechen sind. Hier kann *OpenSG* einen erheblichen Beitrag leisten, die verteilten Informationen effizient zu gestalten.

Des Weiteren zeichnet sich *OpenSG* dadurch aus, dass es multithreadfähig ist. Unter Multithreading wird dabei das zeitgleiche Abarbeiten mehrerer Threads verstanden. Als Thread wird die Reihenfolge bei der Abarbeitung einer Software bezeichnet. Während einzelne Threads schnell auf zeitkritische Abläufe reagieren, können andere lang anhaltende Berechnungen durchführen. Dies kann die Qualität einer Szene enorm steigern. Andere

Szenengraphen verfügen nicht über eine Clusterfähigkeit mit der gleichzeitigen Möglichkeit des Multithreading. Da *OpenSG* auf der Grundlage der Programmiersprache C++ konzipiert wurde, entspricht es den objektorientierten Vorgaben nach hoher Flexibilität, Effizienz und Erweiterbarkeit. Zudem kann *OpenSG* als weitestgehend betriebssystemunabhängig bezeichnet werden, da es auf den am häufigsten genutzten Systemen wie IRIX, Linux und Windows einsetzbar ist und auf OpenGL basiert. *OpenSG* wird daher für die Umsetzung des VR-Moduls als der am besten geeignete Szenengraph bewertet und aus diesem Grund als Kernkomponente für dieses Konzept ausgewählt.

*OpenSG* wurde Anfang 2000 als Open-Source-Projekt ins Leben gerufen und wird durch das *OpenSG* Forum in der Weiterentwicklung überwacht und gelenkt. Das *OpenSG* Forum ist ein Zusammenschluss von Forschungsinstituten, Universitäten und Industriepartnern.

Aus Platzgründen soll die Erklärung des Aufbaus und der Verwendung von *OpenSG* nicht weiter vertieft werden. Hier wird auf die entsprechende Literatur verwiesen [Rei02]. Anhand der Architektur des VR-Moduls wird jedoch mit Hilfe von Syntaxausschnitten der konzeptbezogene Einsatz von *OpenSG* erläutert. Da für *OpenSG* die Programmiersprache C++ eingesetzt wird, wird aus diesem Zusammenhang heraus im folgenden Abschnitt auf die wichtigsten Elemente dieser Sprache eingegangen und damit eine Hilfe für das Verständnis der Modulgestaltung darstellen.

Unter der Federführung von Bjarne Stroustrup wurde C++ von AT&T als objektorientierte Weiterentwicklung von C entwickelt. Im Jahre 1998 wurde für C++ eine einheitliche Sprachspezifikation vom ANSI/ISO Komitee, die ISO/IEC 14882:1998 verabschiedet. Da C++ eine Weiterentwicklung von C ist, wird C++ als Sprachhybrid bezeichnet, der eine prozedurale und eine objektorientierte Programmierung erlaubt [Jos01]. Die Kennzeichen der objektorientierten Programmierung (OOP) lassen sich auf vier Begriffe zurückführen:

- Klassen
- Datenkapselung
- Vererbung
- Polymorphie

Der objektorientierte Ansatz sieht vor, die reale Struktur von Systemen für die Informationsverarbeitung so natürlich wie möglich abzubilden. Jedes System lässt sich dazu als eine Anzahl von bestimmten Objekten darstellen, die miteinander in Verbindung stehen. Der interne Aufbau und das mögliche Verhalten, d.h. die formale Beschreibung von Objekten wird mit Hilfe einer Klasse beschrieben. Jede Klasse besitzt dabei bestimmte Eigenschaften und Funktionen, die von der Klassenzugehörigkeit bestimmt werden. Beispielsweise kann das Verhalten von Objekten einer Klasse spezifiziert sein: wie sie zu

erzeugen sind, wie sie zu verändern sind und wie sie zu zerstören sind [Str98]. Eine C++ Klasse wird als ein abstrakter Datentyp, der ein Objekt beschreibt, bezeichnet. Das Definieren von allgemein zu nutzenden oder auch anwendungsspezifischen Datentypen ist eine der wichtigsten und grundlegendsten Aufgaben in C++ [Str98]. Die von einer Klasse erzeugten Variablen werden in der objektorientierten Programmierung Instanzen genannt [Bre07]. Instanzen bestehen aus den in der Klasse beschriebenen Daten (Membervariablen) und sind mit den dort definierten Funktionen (Memberfunktionen) zu manipulieren.

Mit Hilfe von Klassen können unzulässige Zugriffe auf Daten durch Datenkapselung vermieden werden. Nicht gekapselte Daten sind zu jeder Zeit von jedem Programmierer veränderbar. Dies kann dazu führen, dass bei der Anwendung einer Klasse Inkonsistenzen und Fehler entstehen können. Bei der Nutzung der Datenkapselung kann der Zugriff auf die Daten nur über definierte Schnittstellen erfolgen. Der Zugriff auf interne Daten eines Objektes kann damit nur über Operationen des entsprechenden Datentyps erfolgen, die für den öffentlichen Zugriff vorgesehen sind.

Um die Übersichtlichkeit von komplexen Programmen zu erleichtern und gleichzeitig das weitestgehende Auftreten von Redundanzen zu vermeiden, wird die Vererbung eingesetzt. Sie fördert zudem die Wiederverwendbarkeit des erstellten Programmcodes. Die Vererbung ermöglicht alle Elemente und Methoden einer Klasse einer anderen Klasse zu vererben. Hiermit erhält die neue Klasse alle Daten und Funktionen der alten Klasse und kann zudem um weitere ergänzt oder überschrieben werden. Mit Hilfe der zusätzlichen Elemente und Methoden wird die neue Klasse genauer spezifiziert. Während die alte Klasse als Basisklasse bezeichnet wird, wird die neue Klasse und die damit entstehende Struktur aufgrund der Vererbung Kind-Klasse genannt.

Eine weitere Möglichkeit der objektorientierten Programmierung die Natürlichkeit der Problemabstraktion zu unterstützen ist der Polymorphismus. Dieses Prinzip besagt, dass verschiedene Objekte auf dieselbe Nachricht unterschiedlich reagieren können, indem die jeweilige Operation vom Objekt selbst interpretiert wird. Dies kann unter Umständen erst zur Laufzeit geschehen, da beim Kompilieren nicht unbedingt bekannt ist, welche Operation gemeint ist. Dies bedeutet, dass jedes Objekt seine Klassenzugehörigkeit kennt, und daher in der Lage ist, die entsprechende Methode auszuwählen.

Nach einem kurzen Überblick über die wichtigsten Kennzeichen der OOP wird nun die Umsetzung des VP-Moduls näher erläutert. Um das Verständnis für die VR-Modul-Struktur zu erleichtern wird zunächst ein Überblick über die Bausteine des Moduls und deren Zusammenwirken gegeben. Der Programmcode eines Hauptprogramms ist durch das Schlüsselwort „main“ gekennzeichnet. Aus diesem Grund repräsentiert „main“ das Hauptprogramm in **Abbildung 5-12**.

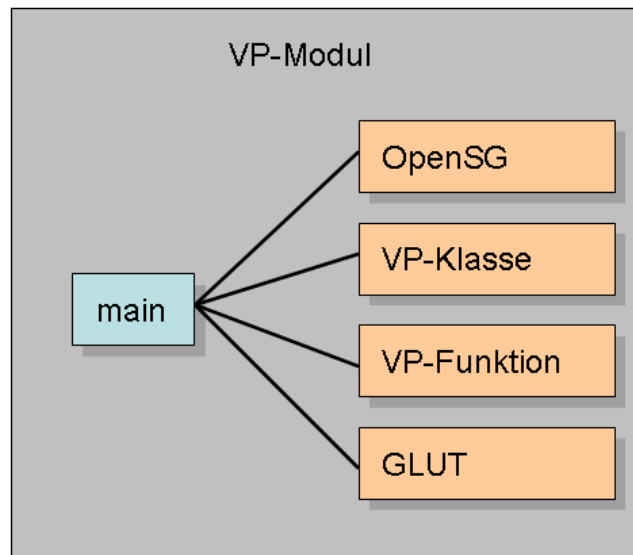


Abbildung 5-12: Bausteine des VP-Moduls

Das Hauptprogramm benötigt für die stereoskopische Visualisierung der virtuellen Prototypen die Komponenten des Szenengraphen *OpenSG*, eine neu gestaltete *VP-Klasse*, eine *VP-Funktion* und das **OpenGL Utility Toolkit GLUT**.

Wie bereits erwähnt, dient *OpenSG* zur stereoskopischen Echtzeitrenderung der Gesamtszene. Um die Übersichtlichkeit der Syntax zu gewährleisten und eine möglichst einfache Wartbarkeit der Programmstruktur zu unterstützen wird eine neue *VP-Klasse* erstellt. Zur Reduktion und besseren Strukturierung des Programmcodes werden für wiederkehrende Aufgaben Memberfunktionen entworfen. Zusätzlich sind alle Aufgaben, die mit der Generierung der kombinierten Gesamtszene als *VRML*-Datei in Verbindung stehen, zusammengefasst und in eine *VP-Funktion* ausgegliedert. Für die plattformunabhängige Ausgabe der Visualisierungsdaten und für Tastaturein-, -ausgaben und Mauskommandos wird *GLUT* eingesetzt. *GLUT* stellt dabei die Verbindung zwischen *OpenGL* und den betriebssystemspezifischen Funktionen und Bibliotheken her.

Bevor die Arbeitsweise des Hauptprogramms erläutert wird, werden zunächst die *VP-Klasse* und anschließend die *VP-Funktion* näher betrachtet, um die einzelnen Komponenten später in einem Gesamtkontext miteinander zu verbinden.

Für die Gesamtprozedur des *VR-Moduls* sind in der *VP-Klasse* die notwendigen Daten als Membervariablen zu deklarieren. Für einen ersten Konzeptentwurf der *VP-Klasse* wird auf Datenkapselung vorerst verzichtet. Neben den Daten für die Dateinamen beinhaltet die *VP-Klasse* zusätzlich die Membervariablen der von der *SolidWorks*-Zusatzanwendung übergebenen Parameter. Für die Typumwandlung der vom *CAD-Zusatzmodul* übergebenen *C-Strings* (vgl. Abschnitt 5.1), wird eine Memberfunktion benötigt, die diese Daten in *String*- bzw. *Double*-Werte umwandelt. Zusätzlich werden zwei weitere Memberfunktionen für die Berechnung des Skalierungsfaktors und der Transformationsparameter (vgl. Abschnitt 4.4.6)



definiert. In **Tabelle 6** sind die Membervariablen und die Memberfunktionen der VP-Klasse zusammengefasst dargestellt.

<b>Membervariable</b>	<b>Beschreibung</b>
Filename_Objekt	Objektdatei
Filename_Raum	Raumdatei
Filename_Gesamt	Neue fusionierte Datei
X_Ausdehnung	Ausdehnung des Objektes in X-Richtung
Y_Ausdehnung	Ausdehnung des Objektes in Y-Richtung
Z_Ausdehnung	Ausdehnung des Objektes in Z-Richtung
X-Koordinate	Mittelpunktcoordinate in X-Richtung
Y-Koordinate	Mittelpunktcoordinate in Y-Richtung
Z-Koordinate	Mittelpunktcoordinate in Z-Richtung
Skalierungsfaktor	Skalierungsfaktor
X-Transformation	Transformationsparameter für X-Koordinate
Y-Transformation	Transformationsparameter für Y-Koordinate
Z-Transformation	Transformationsparameter für Y-Koordinate
<b>Memberfunktion</b>	<b>Beschreibung</b>
StringToDouble	Wandelt einen String in einen Double-Typ
Skalierung	Funktion zur Ermittlung des Skalierungsfaktors
Transformation	Berechnung des Transformationsparameters

*Tabelle 6: Membervariablen und Memberfunktionen der VP-Klasse*

Der ausgegliederten VP-Funktion stehen mit der VP-Klasse nun alle Funktionalitäten für die Erstellung der mit dem Hauptprogramm darzustellenden Gesamtszene zur Verfügung. Das grob skizzierte Ablaufdiagramm der VP-Funktion ist in **Abbildung 5-13** zu erkennen.

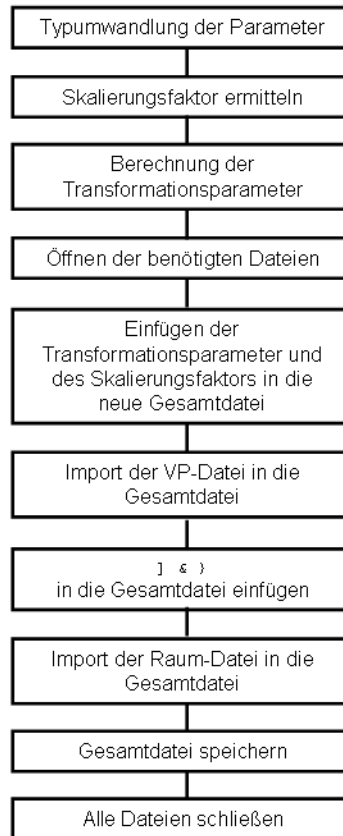


Abbildung 5-13: Ablaufdiagramm der VP-Funktion

Zu Beginn hat eine Aufbereitung der vom CAD-Zusatzmodul erhaltenen Parameter zu erfolgen. Dies bedeutet, dass die übergebenen Werte vom Typ *C-String* zunächst in *String* bzw. *Double* Typen zur Weiterverarbeitung umzuwandeln sind. Anschließend wird mit den nun zur Verfügung stehenden Zahlenwerten der X-, Y- und Z-Objektausdehnungen mit Hilfe der Funktion „Skalierung“ der Skalierungsfaktor berechnet. Dieser ist für die Normierung des Objektes notwendig. Es folgt die Bestimmung der Transformationsparameter für die X-, Y- und Z-Richtung mittels Skalierungsfaktor, um das normierte Objekt wieder in den Koordinatenursprung zu „verschieben“.

Im Anschluss daran, wird für die Fusion der Visualisierungsdaten von Objekt und Raum eine Öffnung aller notwendigen Dateien durchgeführt. In eine neue *VRML*-Gesamtdatei werden im Verlauf der VP-Funktion die Transformationsparameter und der Skalierungsfaktor mithilfe eines *Transform*-Knotens (vgl. Abschnitt 4.3.2) als Textzeilen eingefügt. Danach wird ein Kind-Knoten geöffnet, an den nun die Objektdatei (VP-Datei) der abgeleiteten CAD-Daten angehängt wird (vgl. **Abbildung 5-14**).

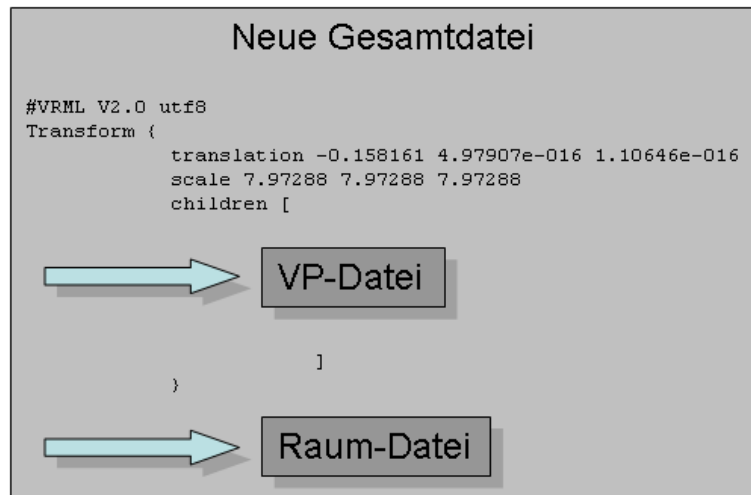


Abbildung 5-14: Struktur der VRML-Gesamtdatei

Nachdem die VP-Datei angehängt wurde, sind der Kind- und der *Transform*-Knoten durch die vorgegebene Klammer-Syntax zu schließen. Erst jetzt erfolgt das Hinzufügen der Raumdatei. Mit dem abschließenden Speichern der Gesamtdatei sind alle geöffneten Dateien geschlossen.

Nachdem die Komponenten des Hauptprogramms jetzt erläutert sind, wird der Ablauf der Hauptroutine näher betrachtet. Zunächst muss eine Möglichkeit gegeben sein, um mit der Aufrufumgebung zu kommunizieren. Bei dem in dieser Arbeit vorgestellten Konzept wird die Kommunikation durch den Programmaufruf aus der CAD-Zusatzanwendung mit der synchronen Übergabe von Umgebungsvariablen gestartet (vgl. Abbildung 5-8). Die Schnittstelle wird im VP-Modul mit der Funktion „*main*“ wie folgt deklariert:

```
int main (int argc, char* argv[] )
```

Mit dem Aufruf des VR-Moduls, werden die Argumente bzw. Übergabeparameter als Feld in `argv` als *C-Strings* übergeben. Die Anzahl der im Feld `argv` vorhandenen Elemente steht in `argc`.

Um das Verständnis, für den nachfolgend erläuterten Prozessablauf des Hauptprogramms zur Visualisierung der Szene zu erleichtern, wird zunächst die Prozesskette mit Hilfe eines groben Ablaufdiagramms in **Abbildung 5-15** dargestellt.

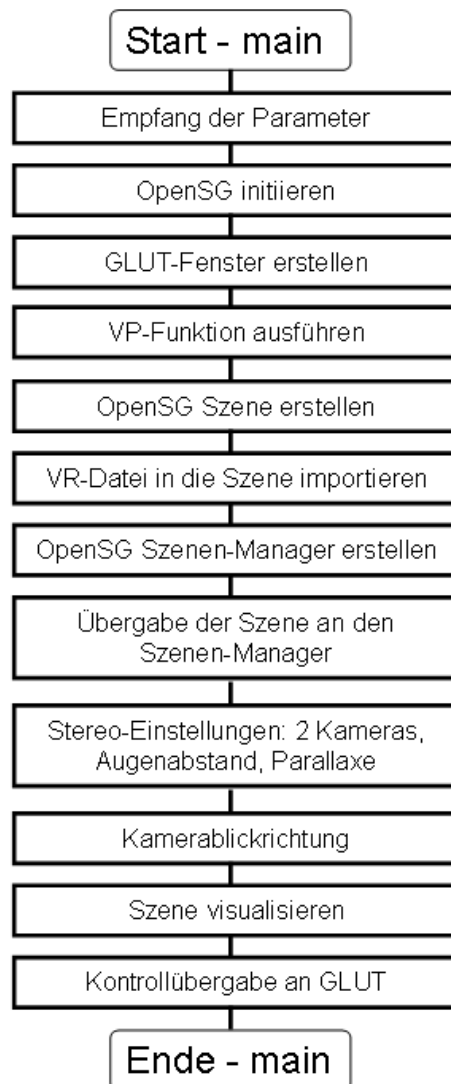


Abbildung 5-15: Ablaufdiagramm des Hauptprogramms

Nach dem Start des VR-Moduls erfolgt zunächst die Initiierung des *OpenSG* Systems. Anschließend wird ein Ausgabefenster für die Visualisierungsdaten als *GLUT*-Window erstellt, indem die Szene mit *OpenGL* gerendert wird. Gleichzeitig wird damit die Verbindung zu den Eingabegeräten hergestellt (vgl. **Abbildung 5-16**).

```
int setupGLUT(int *argc, char *argv[])
{
    glutInit(argc, argv);
    glutInitDisplayMode(GLUT_RGB | GLUT_DEPTH | GLUT_DOUBLE | GLUT_STEREO );
    int winid = glutCreateWindow("VR - Virtual Reality");
    glutFullScreen();
    glutReshapeFunc(reshape);
    glutDisplayFunc(display);
    glutIdleFunc(display);
    glutMouseFunc(mouse);
    glutMotionFunc(motion);
    glutKeyboardFunc(keyboard);
    return winid;
}
```

Abbildung 5-16: Verbindungsherstellung zu GLUT

Die im vorausgegangenen Abschnitt erläuterte VP-Funktion kommt nun zum Einsatz und generiert aus der Objektdatdatei und der Raumdadei eine neue *VRML*-Gesamtdadei. Für die Darstellung der Gesamtdadei mit dem Szenengraphsystem ist im nächsten Schritt eine *OpenSG*-Szene zu generieren. In dieser Szene erfolgt im Anschluss der Import der Gesamtdadei.

Für die Grundkonfiguration einer Szene wird der *OpenSG* Szenen-Manager genutzt. Er stellt die Szene dar, fügt neben den Navigationsfunktionalitäten Licht und eine Kamera in die Szene ein und stellt die Verknüpfung dieser Komponenten sicher. Dem Szenen-Manager wird dann die vorher definierte Szene zur Darstellung übergeben.

Für die stereoskopische Visualisierung der Szene sind noch weitere Konfigurationen vorzunehmen. Dazu gehört zunächst die Einbindung zweier Kameras in die Szene. Die Kameraeinstellungen müssen mit den Parametern des Augenabstandes und der Parallaxeeinstellung justiert sein.

Für den automatisierten Visualisierungsprozess ist es von Vorteil, die Szene stets aus der gleichen Blickrichtung darzustellen. Dies bedeutet, dass eine Voreinstellung des Kamerablickwinkels für die Szene zu definieren ist, die bei jedem Start des VR-Moduls vorerst gleich ist.

Die Szene wird im nächsten Schritt mit dem Szenen-Manager dargestellt und zum Abschluss erfolgt die Kontrollübergabe an *GLUT*. Mittels *GLUT* wird die Darstellung an *OpenGL* übergeben und die Bildausgabe berechnet.

Damit ist die Erklärung zur Architektur des Gesamtkonzeptes zur automatisierten Visualisierung von CAD-Daten in Stereoprojektion abgeschlossen. Nun kann die Verifizierung am Beispiel eines CAD-Objektes anschließen.

## 6 BEISPIELHAFTE VERIFIKATION

Im Folgenden wird die Anwendung des entwickelten Gesamtkonzeptes anhand von zwei Beispielen verdeutlicht. Die exemplarische Verifikation umfasst dabei zum einen die Visualisierung eines manuell aus dem CAD-System *SolidWorks* abgeleiteten Volumenkörpers, der mit dem VR-Modul durch das Absenden einer Kommandozeile auf Betriebssystemebene dargestellt wird. Zum anderen wird ein Universalgelenk aus dem gleichen CAD-System direkt mit Hilfe der automatisierten Prozesskette visualisiert.

Im ersten Beispiel wird an Hand eines einfachen Objektes die globale Einsetzbarkeit des Gesamtkonzeptes, insbesondere im Kontext zu verteilten Unternehmensstandorten verdeutlicht. Es wird gezeigt, dass das Konzept im Rahmen von Simultaneous Engineering für die synchrone Abstimmung von Sachverhalten an unterschiedlichen Standorten von besonderer Bedeutung sein kann.

Im zweiten Beispiel wird gezeigt, wie durch die Integration des Konzeptes in die Bedienoberfläche eines CAD-Systems in Form einer CAD-Zusatzapplikation eine äußerst einfache Anwendung möglich ist. Die Wirksamkeit des Konzeptes wird hierbei durch die automatisierte Prozesskette zur stereoskopischen Darstellung von CAD-Objekten besonders deutlich. Darüber hinaus wird gezeigt, wie durch Farbmodulation und den Einsatz eines sphärischen Raumes, der Tiefeneindruck gesteigert werden könnte. Außerdem lassen die Ergebnisse vermuten, dass aufgrund von Farbmodulation der Ghosting-Effekt reduzierbar ist. Im Gesamtkontext sind daher deutliche Hinweise zu erkennen, wie die Erhöhung des Tiefeneindrucks und die Reduzierung der Ghosting-Effekte einen Beitrag zur Steigerung des Immersionsgrades leisten könnten

### 6.1 Einsatz bei verteilten Standorten

Für die Darstellung von CAD-Objekten wird davon ausgegangen, dass das Objekt bereits als aktive Baugruppe in einem CAD-System geöffnet ist. Zu Beginn sind manuelle Tätigkeiten auf der CAD-Ebene (vgl. Abbildung 5-1) notwendig. Nach der Konzeptsystematik werden zunächst die Farbwerte nach Abschnitt 4.4.5 und die Materialparameter (Optische Eigenschaften) nach Tabelle 3 auf das Objekt angewendet (vgl. **Abbildung 6-1**).

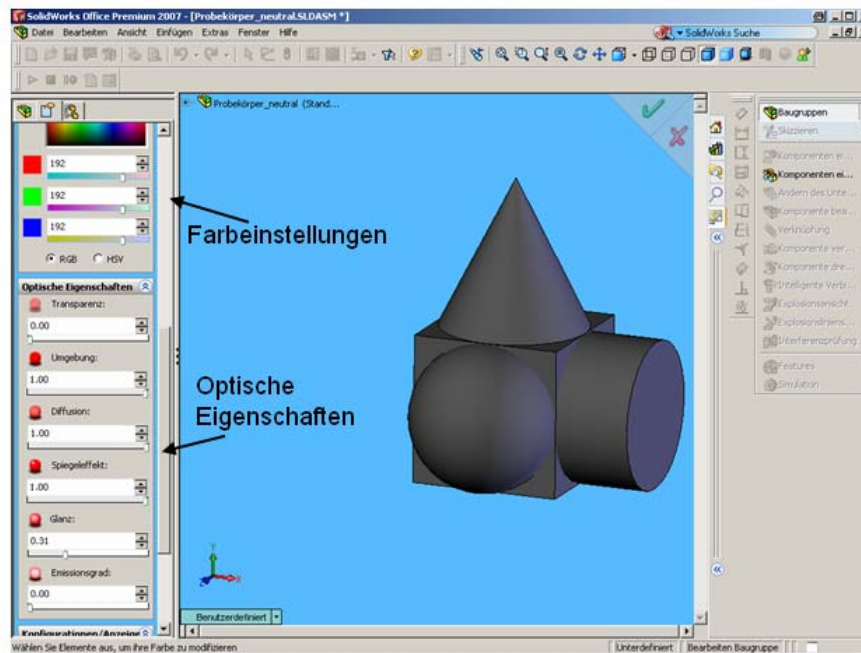


Abbildung 6-1: Einstellung der Farb- und Materialparameter im CAD-System

Diese Parameter befinden sich auf der neutralen Ebene und sind somit allgemeingültig anwendbar. Wenn das in dieser Arbeit vorgestellte Konzept in den Produktentwicklungsprozess integriert werden soll, sind diese Einstellungen bereits in der Konstruktionsphase, dem Kernprozess der Produktentwicklung, vorzunehmen. Damit entfällt ein späterer Konvertierungsprozess in Form von Farb- und Materialzuweisungen.

Das so vorbereitete Objekt wird nun als *VRML*-Datei mit dem Namen *solid.wrl* aus dem CAD-System in den Ordner *c:\stereo\* exportiert. Vor dem Export ist darauf zu achten, dass die gewünschte Darstellungsgüte in Form des Tessellierungsgrades (vgl. Abbildung 4-19) eingestellt ist. Des Weiteren müssen vor dem Export die Exportoptionen nach **Abbildung 6-2** gewählt sein. Nach dem Export steht die *VRML*-Datei zur Visualisierung auf der neutralen Ebene (vgl. Abbildung 5-1) zur Verfügung.

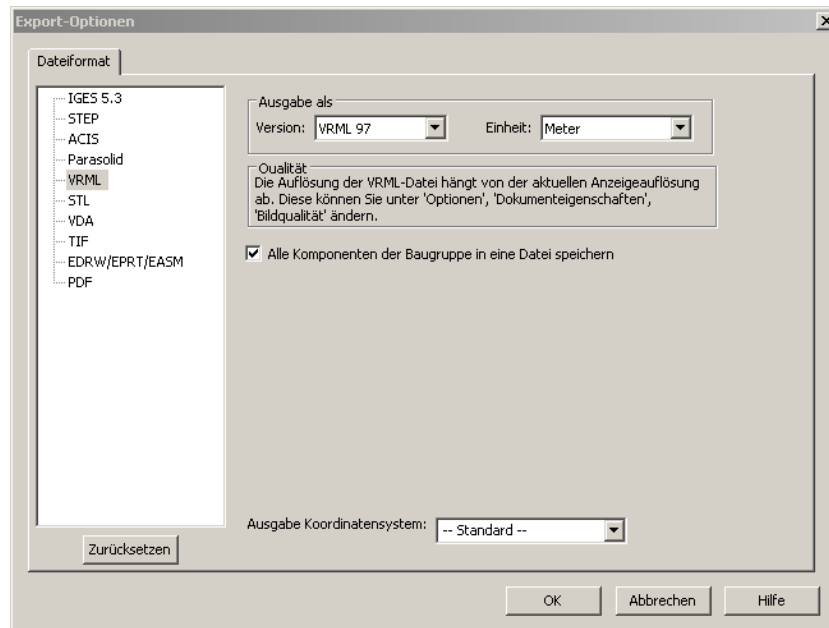


Abbildung 6-2: Export-Optionen

Zum Start der Visualisierung sind noch die folgenden Startparameter zu definieren:

- Länge des CAD-Objektes in X-Richtung [m]
- Länge des CAD-Objektes in Y-Richtung [m]
- Länge des CAD-Objektes in Z-Richtung [m]
- Mittelpunktordinate X des Objektes gemessen vom Ursprung [m]
- Mittelpunktordinate Y des Objektes gemessen vom Ursprung [m]
- Mittelpunktordinate Z des Objektes gemessen vom Ursprung [m]

Diese Werte sind manuell aus dem CAD-System zu ermitteln. In diesem Beispiel sind die Werte wie folgt festgestellt worden:

- Länge des CAD-Objektes in X-Richtung [m] : 0,0015
- Länge des CAD-Objektes in Y-Richtung [m] : 0,002
- Länge des CAD-Objektes in Z-Richtung [m] : 0,0015
- Mittelpunktordinate X des Objektes gemessen vom Ursprung [m] : 0,00025
- Mittelpunktordinate Y des Objektes gemessen vom Ursprung [m] : 0,0005
- Mittelpunktordinate Z des Objektes gemessen vom Ursprung [m] : 0,00025



Nun sind alle notwendigen Parameter vorhanden und sie stehen auf der neutralen Ebene für den weiteren Einsatz zur Verfügung. Als Ausführungsverzeichnis für das VR-Modul wird in diesem Beispiel `c:\stereo\` gewählt. Die Voraussetzungen für die Darstellung der Daten sind in Abschnitt 5.2 zusammengefasst.

Um den Visualisierungsprozess anzustoßen, wird jetzt die Kommandozeile ausgeführt. Als Dateiname wird für das VR-Modul `Stereo-Projekt.exe` gewählt. Die Kommandozeile wird auf dem Betriebssystem Windows XP in einem DOS-Fenster (vgl. **Abbildung 6-3**) eingegeben. Damit wird der Visualisierungsprozess zur stereoskopischen Darstellung der abgeleiteten CAD-Daten gestartet. Mit der im Abschnitt 4.2 vorgestellten Hardware können die Bilddaten anschließend projiziert werden.

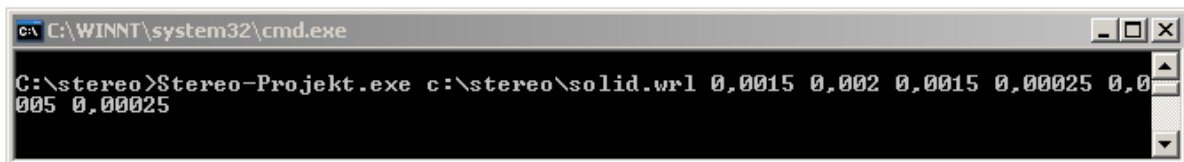


Abbildung 6-3: Absenden der Kommandozeile

Nach diesem Beispiel zur „manuellen“ Visualisierung von CAD-Daten wird im nächsten Beispiel auf die automatisierte Visualisierung näher eingegangen.

## 6.2 Automatisierter Visualisierungsprozess

Für die exemplarische Verifikation des automatisierten Visualisierungsprozesses wird zur Verdeutlichung der Ergebnisse, die bei der Anwendung des Gesamtkonzeptes erreicht werden können, zunächst eine Variante 1 des Universalgelenkes (vgl. **Abbildung 6-4**) ohne Optimierungsmaßnahmen direkt als *VRML*-Datei aus dem CAD-System exportiert und mit dem Softwaresystem *VRED* in Stereoprojektion dargestellt.

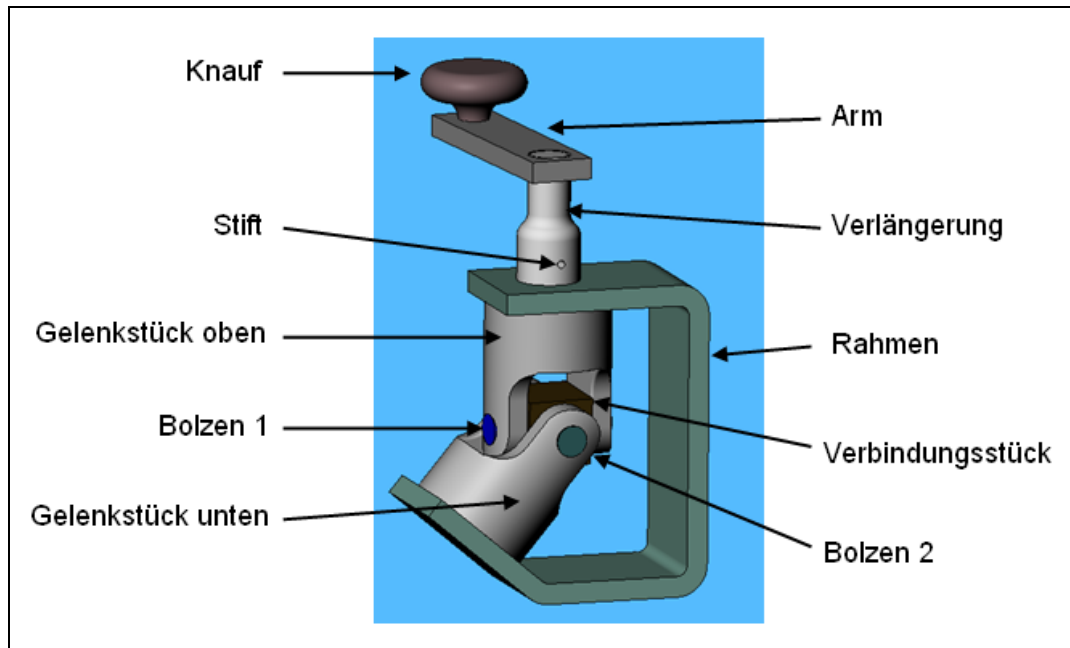


Abbildung 6-4: Universalgelenk [Sol07b]

Im Folgenden werden zwei weitere Varianten abgeleitet, die jedoch für eine stereoskopische Visualisierung mittels Polarisationsverfahren optimiert sind. Sie werden mit den Farb- und Materialparametern versehen, die sich als besonders tauglich herausgestellt haben. Dadurch wird die Wirksamkeit dieses Konzeptes veranschaulicht.

Die Variante 2 des Universalgelenkes wird mit den Farbparametern versehen, die sich in Kombination mit einem blauen Hintergrund als besonders geeignet erwiesen haben (vgl. Abschnitt 4.4.5). Für den Raum werden die RGB-Farbparameter [0.35 0.35 1] eingesetzt. Für die Bauteile des Universalgelenkes werden die folgenden Parameter verwendet:

- Magenta [1 0 1]
- Grün [0 1 0]
- Grün [0.35 1 0.35]
- Rot [1 0.35 0.35]
- Grau [0.75 0.75 0.75]

Demgegenüber wird die Variante 3 mit den RGB-Farbparametern versehen, die sich in Kombination mit einem grauen Hintergrund als besonders geeignet herausgestellt haben. Dies sind für den Raum [0.5 0.5 0.5] und für die Bauteile des Universalgelenkes:

- Rot [1 0.35 0.35]

- Grün [0.35 1 0.35]
- Magenta [1 0 1]

Wenn die Parametereinstellungen (vgl. Abschnitt 6.1) definiert sind, werden die unterschiedlichen Varianten des Universalgelenkes stereoskopisch visualisiert. Die automatisierte Darstellung der Modelle erfolgt durch die Aktivierung des Buttons für die CAD-Zusatzanwendung (vgl. Abbildung 5-9). Da der Benutzer bei der automatisierten Visualisierung nur einen Button zu betätigen hat, ist eine Fehlbedienung nahezu ausgeschlossen. Die exportierte *VRML*-Datei steht danach auf der neutralen Ebene zur Verfügung. Die Übergabeparameter werden zum Programmstart an das VR-Modul unmittelbar übergeben. Anschließend erfolgt die Stereoprojektion des CAD-Objektes durch das VR-Modul. Die Ergebnisse der stereoskopischen Projektion werden als 2D-Bilder in der **Abbildung 6-5** gezeigt.

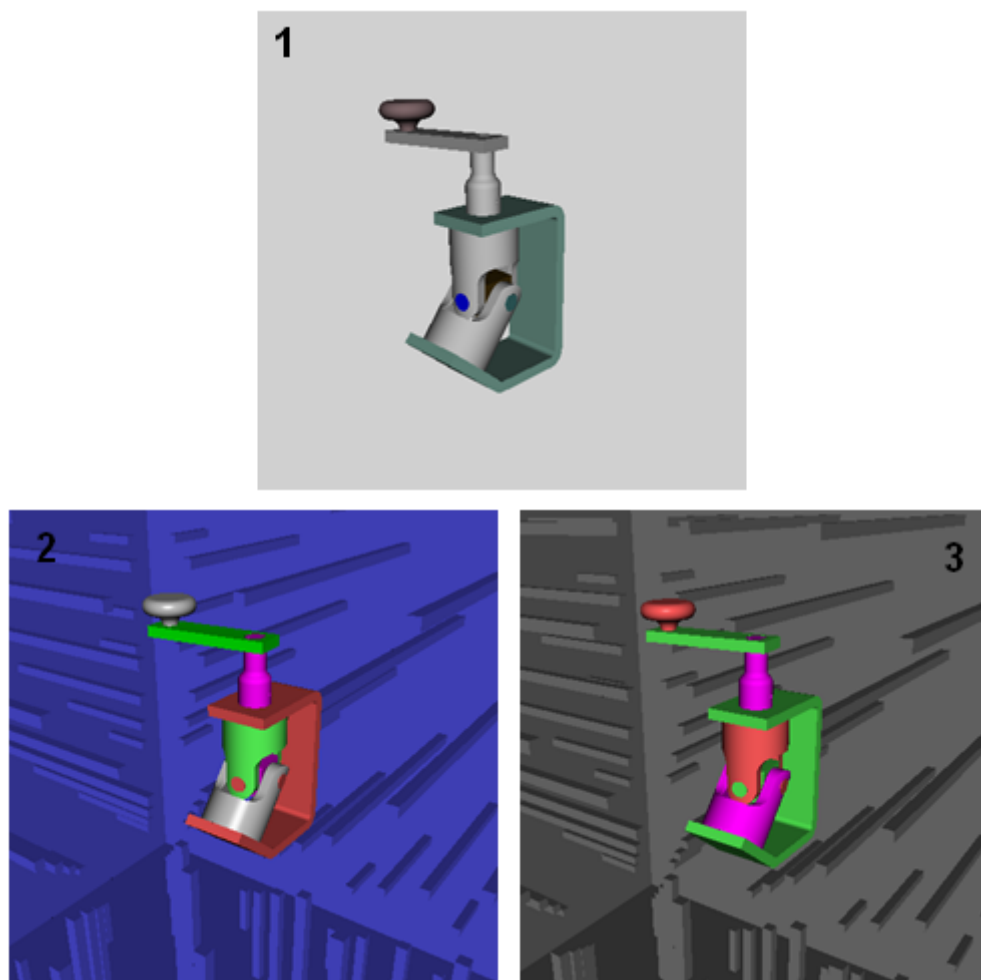


Abbildung 6-5: Darstellungsvarianten des Universalgelenkes

In der Abbildung 6-5 ist ein erheblicher Unterschied in der Tiefenwirkung der drei Bilder zu erkennen. In Bild 1 ist eine geringe Tiefenwirkung zu sehen. Für die Bilder 2 und 3 kann gesagt werden, dass ein deutlicher Anstieg der Tiefenwirkung gegenüber dem Bild 1 festzustellen ist. Bei der Betrachtung der stereoskopischen Darstellung rückt dieser Eindruck noch stärker in den Vordergrund.

Bei der stereoskopischen Betrachtung der Szenen wird die in Abbildung 6-5 verwendete Blickrichtung gewählt. Während in der Szene 2 keine Ghosting-Effekte zu bemerken sind, ist hier die Tiefenwirkung im Vergleich zu Bild 3 geringfügig reduziert. Demgegenüber sind in der Szene 3 kleine Ghosting-Effekte zu bemerken, hierbei scheint die Tiefenwirkung jedoch im Vergleich zu Szene 2 höher zu sein.

Bei der Rotation des Objektes macht sich die Materialeinstellung des Rahmens negativ bemerkbar. Wird senkrecht auf die Rückwand des Rahmens geblickt, kommt es aufgrund der definierten Materialparameter zu einer starken Reflexion und damit zu einer kompletten Weißverfärbung des Rahmens. Da die Farbe Weiß im starken Kontrast zu der Raumfarbe Blau bzw. Grau steht, kommt es zu einer deutlichen Ghosting-Erscheinung.

Die Materialparameter des Rahmens sind entsprechend optimiert worden, damit haben die Spiegelung und der Glanz eine Reduzierung erfahren und somit verhalten sich die potentiell vorhandenen Ghosting-Erscheinungen wie bereits beschrieben.

## 7 ZUSAMMENFASSUNG UND AUSBLICK

Der verschärfte Wettbewerb und Kostendruck sind die Rahmenbedingungen, denen die Unternehmen zunehmend ausgesetzt sind. Deshalb sind die Entwicklungsbereiche ständig gefordert, den wachsenden Anforderungen in Hinsicht auf Innovation und Variantenvielfalt einerseits, 'Time to Market' und Kosteneffizienz andererseits, zu begegnen. Die Entwicklung neuer Produkte in möglichst kurzen Zeiträumen verlangt nach Werkzeugen, die den Entwicklungsprozess beschleunigen. In der Praxis etablieren sich dabei das Rapid und das Virtual Prototyping als zwei Methoden, die grundsätzlich eine Verkürzung der Produktentwicklungsprozesse ermöglichen. Während sich das Rapid Prototyping mit der schnellen Erstellung von physischen Prototypen auseinandersetzt, steht beim Virtual Prototyping die Entwicklung von digitalen Prototypen im Vordergrund. Die Darstellung virtueller Prototypen geschieht häufig mittels Virtual Reality. Diese Technologie wird aufgrund der kostenintensiven Hochleistungsrechner für Photorealismus und der aufwendigen Prozesskette zur Erstellung von VR-Modellen, die Fachpersonal mit Expertenwissen notwendig macht, häufig nur von Großunternehmen eingesetzt.

Um eine Durchdringung von VR-Technologien in den KMU zu erreichen, befinden sich erste Softwarepakete auf dem Markt, die VR auf PC-Basis grundsätzlich ermöglichen. Als Visualisierungstechnologie kommt oftmals das passive Projektionsverfahren mittels Polarisation zur Anwendung. Mit diesem Verfahren ist es möglich, vielen Nutzern zeitgleich eine Produktvisualisierung zu geringen Kosten zu gewährleisten. Ein Defizit stellt hier der geringe Immersionsgrad auf Desktop-Basis dar. Als ein weiterer Nachteil stellt sich heraus, dass die Generierung von VR-Daten aus CAD-Daten eine erhebliche Nacharbeit für eine Optimierung erfordert.

Ziel dieser Arbeit war es, den Produktentwicklungsprozess mittels VR zu beschleunigen und so die 'time-to-market' zu reduzieren. Dazu waren Untersuchungen notwendig, inwieweit der Immersionsgrad auf Single-PC-Basis gesteigert und der Visualisierungsprozess zur Darstellung von CAD-Daten mittels Polarisation automatisiert werden kann.

Die Untersuchungen zur Steigerung des Immersionsgrades lassen Hinweise erkennen, dass durch eine geschickte Farbmodulation der virtuellen Prototypen und der Kulisse zum einen die Tiefenwirkung erhöht und zum anderen die Ghosting-Effekte erheblich reduziert werden könnten. Nach der Transformation des virtuellen Prototyps in das Zentrum eines speziell konzipierten Raumes, ergeben sich auf der Grundlage der Untersuchungsergebnisse Anzeichen, dass mit diesem Vorgehen der Tiefeneindruck möglicherweise zusätzlich zu steigern ist. Die Summe der Ergebnisse lässt vermuten, dass dies zu einem signifikanten Anstieg des Immersionsgrades führen kann.

Im Anschluss an diese Untersuchungen werden die gewonnenen Ergebnisse als Eingangsgrößen genutzt, um am Beispiel des CAD-Systems *SolidWorks* aus CAD-Daten zeitnah virtuelle Prototypen mittels VR darzustellen. Der Focus dieser Arbeit richtete sich

nach einer manuellen Parameteroptimierung der CAD-Daten auf die direkte Ableitung und Visualisierung des VP-Datensatzes aus einem 3D-CAD-System. Hierzu war neben den bereits angesprochenen Untersuchungen auch die Konzeption und Entwicklung eines Softwarealgorithmus erforderlich, der in nur einem Prozessschritt:

- die Ableitung des VP-Modells aus dem CAD-System und
- die Visualisierung der fusionierten Modelle

automatisiert.

Die vorliegende Arbeit zeigt, dass die Automatisierung des Visualisierungsprozesses im Hinblick auf die 3D-Visualisierung von technischen Prototypen aus CAD-Systemen noch ein erhebliches Potential besitzt.

Unter Berücksichtigung der in dieser Arbeit entwickelten Ansätze ist zu erkennen, dass die Ableitung und Visualisierung von virtuellen Prototypen deutlich beschleunigt werden kann. Ebenfalls konnte gezeigt werden, dass zur Steigerung des Immersionsgrades beim Polarisationsverfahren noch erhebliche Optimierungsreserven vorhanden sind.

Die Nutzung dieses Verfahrens ermöglicht eine Verkürzung der „time-to-market“. Vor diesem Hintergrund kann sich ein Wettbewerbsvorteil, insbesondere für die KMU ergeben, da diese in der Regel bezüglich einer eigenen Entwicklung für diese Thematik nur über sehr geringe finanzielle Mittel verfügen.

Aufgrund der eingeschränkten Hardwareperformance auf Single-PC-Basis ist keine photorealistische Visualisierung umgesetzt worden. Durch die offene Gesamtarchitektur des Konzeptes ist es jedoch möglich, den Immersionsgrad der Prototypen durch Erweiterungen zusätzlich zu steigern.

Die bei den Untersuchungen zur Steigerung des Immersionsgrades gewählte geringe Probandenzahl lässt zwar durch die neuen Ansätze und Ideen eine potentielle Steigerung des Immersionsgrades vermuten, die Untersuchungsergebnisse sind jedoch nicht als statistische Verifikation zu interpretieren.

Hier bleibt noch Raum für weitere Untersuchungen, um mit einer höheren Probandenzahl die Untersuchungsergebnisse wissenschaftlich zu untermauern.

## 8 LITERATUR

- [Ame97] AMES, A. L.; NADEAU, D. R.; MORELAND, J. L.: *VRML 2.0 Sourcebook*  
New York, John Wiley & Sons, Inc., 1997
- [Ass03] ASSMANN, B. O.: *Herstellung hochgenauer Prototypen mittels Fräsen als quasi-generativem Rapid-Prototyping-Verfahren*  
Universität Duisburg-Essen, Diss., 2003
- [Bar03] BARCO: *Stereoscopic Projection*  
Barco Simulation Products, USA, 2003
- [Bed03] BENDER, M; BRILL, M.: *Computergrafik*  
München, Wien, Carl Hanser Verlag, 2003
- [Ber07a] BERGERS, D.: *Produktentwicklung*  
Skriptum, Universität Duisburg-Essen 2007
- [Ber07b] BERGERS, D.: *Produkt Engineering*  
Skriptum, Universität Duisburg-Essen 2007
- [Ber07c] BERGERS, D.: *Rapid und Virtual Prototyping*  
Skriptum, Universität Duisburg-Essen 2007
- [Bla90] BLAKE, A.; BÜLTHOFF, H. H.: *Does the brain know the physics of specular reflection?*  
Nature, 343: 165 – 168, 1990
- [Bre07] BREYMANN, U.: *C++, Einführung und professionelle Programmierung*  
München, Wien; Carl Hanser Verlag, 2007
- [Bre97] BREYMANN, U.: *C++, Eine Einführung*  
München, Wien; Carl Hanser Verlag, 1997
- [Bul97] BULLINGER, H.-J.; WARSCHAT, J.: *Forschungs- und Entwicklungsmanagement*  
Stuttgart, 1997
- [Cut95] CUTTUNG, J. E.; VISHTON, P. M.: *Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth*  
In W. Epstein & S. Rogers (eds.) Handbook of perception and cognition, Vol 5; Perception of space and motion. (pp. 69-117). San Diego, CA: Academic Press, 1995
- [Dub00] BEITZ, W.; GROTE, K.-H.: *Dubbel – Taschenbuch für den Maschinenbau*  
Berlin, Heidelberg, New York, Springer-Verlag, 2000
- [Ehr03] EHRENSPIEL, K.: *Integrierte Produktentwicklung*  
München, Wien: Carl Hanser Verlag, 2003
- [Fol94] FOLEY, J. D.; VAN DAM, A.; FEINER, S. K.; HUGES, J. F.; PHILLIPS, R. L.: *Grundlagen der Computergraphik*  
Bonn, Addison-Wesley, 1994
- [Fro07] FRÖHLICH, B.; BLACH, R.; VAN LIERE, R.: *Three Extentions to Subtractive Crosstalk Reduction*  
IPT-EGVE Symposium, pp. 1-9, 2007
- [Gau00] GAUSEMEIER, J.; LINDEMANN, U.; REINHART, G.; WIENDAHL, H. P.: *Kooperatives Produktengineering*  
Paderborn: HNI, Bd. 79, 2000
- [Geb96] GEBHARDT, A.: *Rapid Prototyping*

- München, Wien, Carl Hanser Verlag 1996
- [Gra86] GRANDIS, L. D.: *Theory and Use of Color*  
Engelwood Cliffs, New Jersey, Pentice Hall Inc., 1986
- [Gut83] GUTENBERG, E.: *Grundlagen der Betriebswirtschaftslehre. Band 1: Die Produktion*  
Berlin, Heidelberg, Springer Verlag, 1983
- [Har07] HARTMANN, K.: *Computergraphik I*  
Skriptum, Otto-von-Guericke-Universität Magdeburg, 2007
- [Hei61] HEIMENDAHL, E.: *Licht und Farbe*  
Berlin, Walter de Gruyter & Co., 1961
- [Hen97] HENNING, A. : *Die andere Wirklichkeit*  
Bonn: Addison Wesley Longman Verlag, 1997
- [Her04] HERMANN, G. M.: *Physik für Biologen, Agrarwissenschaftler und Haus- und Ernährungswissenschaftler*  
Skriptum, Justus-Liebig-Universität Gießen WS 2004 / 2005
- [Hof02] HOFMANN, J.: *Raumwahrnehmung in virtuellen Umgebungen*  
Diss. Technische Universität München  
Deutscher Universitätsverlag, Wiesbaden, 2002
- [Hoy06] HOYER, J., UHL, C., BEYER, C.: *Virtual & Rapid Prototyping*  
Düsseldorf, VDI-Verlag 2006
- [Jam95] JAMES S. LIPSCOMB, WAYNE L. WOOTEN: *Reducing crosstalk between stereoscopic views*  
in Proc. of SPIE (Stereoscopic Displays and Virtual Reality Systems II), Vol. 2409, pp.31-40, Feb. 1995
- [Jos01] JOSUTTIS, N.: *Objektorientiertes Programmieren in C++*  
München, Addison-Wesley Verlag, 2001
- [Kli03] KLIMENKO, S; FROLOV, P.; NIKITINA, L; NIKITIN, I.: *Crosstalk reduction in passive stereo-projection systems*  
Eurographics Association, Schweiz, 2003
- [Lie97] LIEDL, R.; AMERSTORFER, S. N.: *Die Pracht der Farben*  
Heidelberg, Spektrum Akademischer Verlag, 1997
- [Lob04] LOBECK, F.: *Konzept zur Optimierung von Produktentwicklungsprozessen einschließlich Simulation und Rapid Prototyping unter Verwendung eines neuen PLM-CAD-Integrationsmoduls*  
Universität Duisburg-Essen, Habil., 2004
- [Mal00] MALLOT, H. A.: *Sehen und die Verarbeitung visueller Information*  
Braunschweig, Wiesbaden, Vieweg Verlag 2000
- [Mey95] MEYR, H.: *Aufbau von High-Tech Unternehmungen, Anforderungen hinsichtlich Technik, Unternehmensführung und Finanzen*  
Tagungsband LBBZ, Aachen, 1995
- [Raa05] RAAB, D.: *Optimierung von virtuellen Prototypen für das stereoskopische Polarisationsverfahren zur Visualisierung von Kommunikationsmodellen in der Produktentwicklung*  
Universität Duisburg-Essen, Diplomarbeit, 2005
- [Rei02] REINERS, DIRK: *OpenSG: A Scene Graph System for Flexible and Efficient Realtime Rendering for Virtual and Augmented Reality Applications*  
Technische Universität Darmstadt, Diss., 2002



- [Rod91] RODENACKER, W.: *Methodisches Konstruieren*  
Berlin, Heidelberg, 4. überarb. Auflage, Springer Verlag, 1991
- [Sch00] SCHUMANN, H.; MÜLLER, W.: *Visualisierung*  
Berlin, Heidelberg, Springer-Verlag, 2000
- [Sch02] SCHUH, G.; LOCKE, F.; STRAUBE, M.; RIPP, S.; HOLLREISER, J.: *Integration als Grundlage der digitalen Fabrikplanung*  
Springer VDI-Verlag, VDI-Z, Integrierte Produktion, Nr. 11/12, 2002
- [Sch89] SCHMELZER, H. J.: *Wettbewerbsvorteile durch kürzere Entwicklungszeiten*  
Siemens-Zeitschrift, Heft 5/1989
- [Sei98] SEIBERT, S.: *Technisches Management*  
Stuttgart, Leipzig, Teubner Verlag, 1998
- [Skr96] SCHULZ-KRÖHNERT, U.: *Wahrnehmungspsychologische Aspekte der Entwicklung verwechslungsfreier Farbcodierungen für CAD-Anwendungen*  
Universität Gesamthochschule Kassel, Diss., 1996
- [Smi07] SMITH, F.; VAN LIERE, R.; FRÖHLICH, B.: *Non-uniform Crosstalk Reduction for Dynamic Scenes*  
In Proc. IEEE Virtual Reality, pp. 139-146, 2007
- [Sol07a] SOLIDWORKS CORPOARTION: *SolidWorks Online Benutzerhandbuch 2007*  
SolidWorks Corporation, Massachusetts, USA, 2007
- [Sol07b] SOLIDWORKS CORPOARTION: *SolidWorks Online Lehrbuch 2007*  
SolidWorks Corporation, Massachusetts, USA, 2007
- [Som07] SOMMERVILLE, I.: *Software Engineering*  
München, Pearson Education Deutschland GmbH, 2007
- [Spu96] SPUR, G.: *Hütte – Die Grundlagen der Ingenieurwissenschaften*  
Berlin, Heidelberg, Springer-Verlag 1996
- [Spu97] SPUR, G; KRAUSE, L.: *Das virtuelle Produkt*  
München, Wien, Hanser-Verlag 1997
- [Sta08] STANDKE, J.: *Rechnerunterstützte Auslegung von reversiblen Füge-techniken für quasi-generativ hergestellte Funktionsprototypen*  
Universität Duisburg-Essen, Diss., 2008
- [Ste01] STENZEL, H.: *Schlußbericht, Visualisierung von Prozess-Informationen*  
Fachhochschule Köln, 2001
- [Ste07] STENZEL, H.: *Avango-Einführung*  
Fachhochschule Köln, Fakultät für Informatik und Ingenieurwissenschaften 2007
- [Ste95] STEGER, W.: *Neue Verfahren der Prototypenherstellung und Ihre Integration in der Unternehmensstrategie*  
Spektrum der Wissenschaft, S. 90, April 1995
- [Ste97] STEREOGRAPHICS: *Developers Handbook: Background on Creating Images for CrystalEyes and SimulEyes*  
StereoGraphics Corporation 1997
- [Str98] STROUSTRUP, B.: *Die C++ Programmiersprache*  
Bonn, Addison Wesley Longman Verlag, 1998
- [Tod83] TODD, J. T.; MINGOLLA: *Perception of surface curvature and direction of illumination from patterns of shading*

Journal of Experimental Psychology:  
Human Perception and Performance, 9:583 – 595, 1983

- [VRC06] REKTOR VOM VIRTUAL REALITY CENTER DER RWTH AACHEN: *Jahresbericht 2005/2006*  
Rheinisch-Westfälische Technische Hochschule Aachen, 2006
- [VRE06] FRAUNHOFER-INSTITUT FÜR WERKZEUGMASCHINEN UND UMFORMTECHNIK: *Virtual-Reality-Entwicklungsumgebung für Werkzeugmaschinen*  
Chemnitz, Fraunhofer-Institut für Werkzeugmaschinen und Umformtechnik, 2006
- [Wei07] WEIDLICH, D.: *Virtual Reality mindert Risiken in Entwicklung und Produktion*  
Würzburg, In MM-Maschinenmarkt, Vogel Industrie Medien GmbH & Co. KG, 2007

## 9 ANHANG

### 9.1 Abbildungsverzeichnis

Abbildung 2-1: Das Geschäftsfeld [Ber07a] .....	3
Abbildung 2-2: Wertschöpfung [Ber07b] .....	5
Abbildung 2-3: Wertschöpfungskette [Ber07b].....	5
Abbildung 2-4: Produktentwicklungsprozess [Ber07a] .....	6
Abbildung 2-5: Stellhebel zur Verkürzung des Produktentwicklungsprozesses [Ber07c] .....	6
Abbildung 2-6: Zeitverkürzung mittels Simultaneous Engineering [Ber07c].....	7
Abbildung 2-7: Einfluss unterschiedlicher Zielverfehlungen auf den erzielbaren Gewinn [Geb96].....	9
Abbildung 2-8: Produktdaten in einem 3D-CAD-System [Sol07b] .....	12
Abbildung 2-9: Zunehmende Genauigkeit bei Reduzierung des Sehnenfehlers [Ber07c] .....	13
Abbildung 2-10: Funktionsanalyse der Prototypenherstellung [Ass03] .....	14
Abbildung 2-11: Prozess der Prototypenherstellung [Lob04] .....	15
Abbildung 2-12: Stereoskopisches Sehen in Projektionssystemen [Ber07c] .....	16
Abbildung 2-13: Workbench [BARCO] .....	18
Abbildung 2-14: Holobench [RWTH Aachen] .....	19
Abbildung 2-15: Powerwall [BARCO] .....	20
Abbildung 2-16: CAVE [VRCA] .....	20
Abbildung 2-17: 6-Seiten-CAVE HyPi-6 [Fraunhofer IAO] .....	21
Abbildung 2-18: Das Projektionsverfahren mittels Polarisation.....	22
Abbildung 2-19: Crosstalk-Erscheinung .....	23
Abbildung 2-20: Entstehung des Geisterbildes [Kli03] .....	24
Abbildung 2-21: Eliminierung des Geisterbildes [Kli03].....	24
Abbildung 2-22: Die Visualisierungspipeline .....	26
Abbildung 4-1: Parameter der Raumwahrnehmung .....	38
Abbildung 4-2: Relative Helligkeit [Ste97] .....	39
Abbildung 4-3: Relative Größe .....	39
Abbildung 4-4: Atmosphärische Perspektive [Ste97] .....	40

---

Abbildung 4-5: Textur .....	40
Abbildung 4-6: Texturgradient .....	41
Abbildung 4-7: Höhe im Gesichtsfeld .....	41
Abbildung 4-8: Interposition.....	42
Abbildung 4-9: Licht und Schatten [Fol94] .....	42
Abbildung 4-10: Lineare Perspektive [Ste97].....	43
Abbildung 4-11: Sphärische Perspektive [Mal00].....	43
Abbildung 4-12: Farbiger Tiefeneffekt.....	43
Abbildung 4-13: Verwendete Softwaresysteme .....	44
Abbildung 4-14: Materialeigenschaften [Fol94].....	46
Abbildung 4-15: Baugruppe mit unterschiedlichen Materialeigenschaften [Sol07b] .....	47
Abbildung 4-16: Eingabemaske für die Farbeinstellung in SW2007 .....	48
Abbildung 4-17: Eingabemaske für die optischen Eigenschaften in SW2007.....	48
Abbildung 4-18: Eingabemaske für Texturen in SW 2007 .....	50
Abbildung 4-19: Eingabemaske für den Tessellierungsgrad in SW 2007 .....	51
Abbildung 4-20: Würfel.....	52
Abbildung 4-21: VRML-Code - Würfel.....	52
Abbildung 4-22: Spektrum des sichtbaren Lichtes [Sch00].....	58
Abbildung 4-23: Spektralreaktion der drei Arten von Zäpfchen auf der Retina [Fol94] .....	58
Abbildung 4-24: Additives RGB-Farbmodell [Her04].....	60
Abbildung 4-25: RGB-Würfel [Bed03] .....	61
Abbildung 4-26: Links warme Farben – rechts kalte Farben.....	62
Abbildung 4-27: Komplementärfarben.....	63
Abbildung 4-28: Simultankontrast .....	63
Abbildung 4-29: Hell-Dunkel-Kontrast [Lie97] .....	64
Abbildung 4-30: Bunt-Unbunt-Kontrast .....	65
Abbildung 4-31: Raumwahrnehmung durch Schattierung.....	67
Abbildung 4-32: Flat-Shading [Har07].....	68
Abbildung 4-33: Gouraud-Shading [Har07] .....	68
Abbildung 4-34: Phong-Shading [Har07].....	68

---

Abbildung 4-35: Einfacher Volumenkörper.....	70
Abbildung 4-36: Untersuchungsmatrix - Farbe.....	71
Abbildung 4-37: Gruppe A – Positive Tiefenwirkung.....	73
Abbildung 4-38: Gruppe B – Positive Tiefenwirkung.....	74
Abbildung 4-39: Objekt mit und ohne metallische Materialeigenschaften.....	77
Abbildung 4-40: Untersuchungsmatrix - Ghosting-Effekt .....	80
Abbildung 4-41: Gruppe A – Sehr geringe Ghosting-Effekte .....	81
Abbildung 4-42: Gruppe B – Geringe Ghosting-Effekte .....	81
Abbildung 4-43: Verschachtelung der Elemente .....	85
Abbildung 4-44: Objekt und verschachtelte Elemente .....	85
Abbildung 4-45: Elementdichte .....	86
Abbildung 4-46: Elementquerschnittsgeometrien [Raa05].....	86
Abbildung 4-47: Raumgestaltung.....	87
Abbildung 4-48: Objekt mit neutralem Hintergrund und mit neuem Raum.....	87
Abbildung 5-1: Architektur des Gesamtkonzeptes .....	89
Abbildung 5-2: VBA-Editor .....	96
Abbildung 5-3: Ermittlung der Objektausdehnungen.....	98
Abbildung 5-4: Bestimmung von Max. bzw. Min. dreier Werte.....	99
Abbildung 5-5: Bestimmung der Mittelpunktkoordinaten und der Längenausdehnung.....	99
Abbildung 5-6: Datentypumwandlung .....	100
Abbildung 5-7: VRML-Exportbefehl.....	100
Abbildung 5-8: Start des VR-Moduls .....	101
Abbildung 5-9: Menübutton zum Starten des VR-Prozesses .....	102
Abbildung 5-10: Grafik-Abstraktionsebenen.....	103
Abbildung 5-11: Abstrahiertes Sonnensystem .....	104
Abbildung 5-12: Bausteine des VP-Moduls.....	107
Abbildung 5-13: Ablaufdiagramm der VP-Funktion .....	109
Abbildung 5-14: Struktur der VRML-Gesamtdatei.....	110
Abbildung 5-15: Ablaufdiagramm des Hauptprogramms .....	111
Abbildung 5-16: Verbindungsherstellung zu GLUT .....	112

---

Abbildung 6-1: Einstellung der Farb- und Materialparameter im CAD-System .....	114
Abbildung 6-2: Export-Optionen .....	115
Abbildung 6-3: Absenden der Kommandozeile .....	116
Abbildung 6-4: Universalgelenk [Sol07b] .....	117
Abbildung 6-5: Darstellungsvarianten des Universalgelenkes .....	118
Abbildung 9-1: Teil des Bewertungsbogens – Farbiger Tiefeneffekt.....	133
Abbildung 9-2: Teil des Bewertungsbogens – Ghosting-Effekt.....	134

## 9.2 Abkürzungen - Indizes

<b>Abkürzungen - Indizes</b>	<b>Bedeutung</b>
2D	Zweidimensional
3D	Dreidimensional
ANSI	American National Standard Institute
API	Application Programming Interface
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CAVE	Cave Automatic Virtual Environment
CNC	Computer Numeric Control
COM	Component Object Model
CPU	Central Processing Unit
CRT	Cathode Ray Tube
DLL	Dynamic Link Library
DLP	Digital Light Processing
EgT	Egozentrische Tiefe
ExT	Exozentrische Tiefe
FMEA	Fehlermöglichkeiten- und Einflußanalyse
IMK	Fraunhofer Institut Medienkommunikation
IRIX	Kommerzielles Betriebssystem von Silicon Graphics
ISO	International Organization for Standardization
KMU	Klein- und mittelständische Unternehmen
L	Bild für das linke Auge
LCD	Liquid Cristal Display
LGPL	Lesser General Public License
LoD	Level of Detail
PC	Personal Computer
PDM	Product Data Management
R	Bild für das rechte Auge

---

RP	Rapid Prototyping
SE	Simultaneous Engineering
SIG	Silicon Graphics Incorporated
STL	Stereolithographic Language
VP	Virtual Prototyping
VR	Virtual Reality
VRCA	Virtual Reality Center Aachen
VRML	Virtual Reality Modeling Language
$\alpha$	Subtrahiertes Teilbild
$\beta$	Addiertes Teilbild



9.3 Tabelle und Abbildungen

	Objekt	Ret [1 x x]	Grün [1 x x]	Blaue [x x 1]	Grün [x x 1]	Blau [x x 1]	0,60	0,25	0,75	Gelb [x 0]	Magenta [x 0 x]	Cyan [0 x x]	Weiß [x x x]	Schwarz [x x x]
Hintergrund	x	0,00	0,80	0,00	0,35	0,60	0,25	0,50	1,00	1,00	1,00	1,00	1,00	0,00
	0,00	1	3	6	7	8	9	10	12	13	14	15	16	17
	0,25	18	20	23	24	25	26	27	28	29	31	<b>32</b>	33	34
	0,50	35	37	39	40	41	42	43	44	45	46	47	48	49
	0,75	52	54	55	57	58	59	60	61	62	63	64	65	66
	1,00	69	71	72	73	74	75	76	77	78	80	81	82	83
	0,00	<b>86</b>	88	89	<b>90</b>	91	92	94	95	96	<b>99</b>	100	<b>84</b>	101
	0,25	<b>103</b>	<b>104</b>	<b>105</b>	<b>106</b>	<b>108</b>	109	111	112	113	<b>114</b>	<b>115</b>	<b>118</b>	119
	0,50	<b>120</b>	<b>121</b>	<b>122</b>	<b>123</b>	<b>124</b>	125	126	127	128	129	<b>132</b>	<b>135</b>	136
	0,75	137	138	139	140	141	142	143	144	145	146	147	148	149
	1,00	<b>154</b>	<b>155</b>	156	157	158	159	160	<b>161</b>	162	163	164	165	166
	0,25	171	172	173	174	175	176	177	178	179	180	181	182	183
	0,50	<b>188</b>	189	190	191	192	193	194	195	196	197	198	199	200
	0,75	205	206	207	<b>208</b>	209	210	211	212	213	214	215	216	217
	1,00	222	223	224	225	226	227	228	<b>229</b>	230	231	232	233	234
	0,25	239	240	241	242	243	244	245	246	247	248	249	250	251
	0,50	256	257	258	259	<b>260</b>	261	262	263	264	265	266	267	268
	0,75	<b>273</b>	<b>274</b>	<b>275</b>	<b>276</b>	<b>277</b>	278	279	280	<b>281</b>	282	283	284	285
	1,00													
	0,00													

Tabelle 7: Untersuchungsmatrix - 1

**Fragebogen zur subjektiven Bewertung einer Testreihe zum farbigen Tiefeneffekt**

Angaben zur Person:

Alter: _____	Sehhilfe? _____	Ja / Nein _____	Geschlecht: ♀ / ♂ _____
--------------	-----------------	-----------------	-------------------------

**Aufgabenstellung:**

Es wird eine Abfolge von unterschiedlich kolorierten Bildern visualisiert. Die visualisierten Abbildungen sollen danach bewertet werden, in wie weit der in der Abbildung dargestellte und kolorierte Volumenkörper gegenüber dem farbigen Hintergrund eine räumliche Differenz (Raumwahrnehmung) empfinden lässt. Dabei soll unterschieden werden in positive, negative oder keine Raumwahrnehmung. Die Raumwahrnehmung soll als positiv bewertet werden, wenn der Volumenkörper auf dem Hintergrund als schwebend empfunden wird bzw. sich vom Hintergrund in Richtung des Betrachters abhebt. Die Bewertung hat von 1 (geringer Effekt) bis 3 (starker Effekt) zu erfolgen. Die Raumwahrnehmung soll als negativ bewertet werden, wenn der Volumenkörper gegenüber der farbigen Kulisse in den Hintergrund rückt bzw. den Eindruck empfinden lässt, dass er sich in der Tiefe, also hinter der Kulisse befindet. Die Bewertung hat von -1 (geringer Effekt) bis -3 (starker Effekt) zu erfolgen. Die Raumwahrnehmung soll als nicht vorhanden bewertet werden, wenn keine räumliche Differenz zwischen Volumenkörper und Hintergrund festzustellen ist. In diesem Fall ist eine Bewertung mit 0 vorzunehmen. Die subjektiven Messwerte werden durch Ankreuzen in der entsprechenden Bewertungsskala dokumentiert.

**Testbefragung:**

Bild-Nr.                      Bewertungsskala

Test A	-3	-2	-1	0	+1	+2	+3
Test B	-3	-2	-1	0	+1	+2	+3
Test C	-3	-2	-1	0	+1	+2	+3

**Bewertung der Messreihe:**

Nr. 001	-3	-2	-1	0	+1	+2	+3
Nr. 002	-3	-2	-1	0	+1	+2	+3
Nr. 003	-3	-2	-1	0	+1	+2	+3
Nr. 004	-3	-2	-1	0	+1	+2	+3
Nr. 005	-3	-2	-1	0	+1	+2	+3
Nr. 006	-3	-2	-1	0	+1	+2	+3
Nr. 007	-3	-2	-1	0	+1	+2	+3
Nr. 008	-3	-2	-1	0	+1	+2	+3
Nr. 009	-3	-2	-1	0	+1	+2	+3
Nr. 010	-3	-2	-1	0	+1	+2	+3
Nr. 011	-3	-2	-1	0	+1	+2	+3
Nr. 012	-3	-2	-1	0	+1	+2	+3
Nr. 013	-3	-2	-1	0	+1	+2	+3
Nr. 014	-3	-2	-1	0	+1	+2	+3
Nr. 015	-3	-2	-1	0	+1	+2	+3
Nr. 016	-3	-2	-1	0	+1	+2	+3
Nr. 017	-3	-2	-1	0	+1	+2	+3
Nr. 018	-3	-2	-1	0	+1	+2	+3
Nr. 019	-3	-2	-1	0	+1	+2	+3
Nr. 020	-3	-2	-1	0	+1	+2	+3
Nr. 021	-3	-2	-1	0	+1	+2	+3
Nr. 022	-3	-2	-1	0	+1	+2	+3
Nr. 023	-3	-2	-1	0	+1	+2	+3
Nr. 024	-3	-2	-1	0	+1	+2	+3
Nr. 025	-3	-2	-1	0	+1	+2	+3
Nr. 026	-3	-2	-1	0	+1	+2	+3
Nr. 027	-3	-2	-1	0	+1	+2	+3
Nr. 028	-3	-2	-1	0	+1	+2	+3
Nr. 029	-3	-2	-1	0	+1	+2	+3
Nr. 030	-3	-2	-1	0	+1	+2	+3
Nr. 031	-3	-2	-1	0	+1	+2	+3
Nr. 032	-3	-2	-1	0	+1	+2	+3

Abbildung 9-1: Teil des Bewertungsbogens – Farbiger Tiefeneffekt

**Fragebogen zur subjektiven Bewertung von Ghosting-Effekten**

Angaben zur Person:

Alter:	<input style="width: 80%;" type="text"/>	Sehhilfe?	<input type="checkbox"/> Ja / <input type="checkbox"/> Nein	Geschlecht:	<input type="checkbox"/> ♀ / <input type="checkbox"/> ♂
--------	--	-----------	---	-------------	---

**Aufgabenstellung:**

Es wird eine Abfolge von unterschiedlich kolorierten Szenen stereoskopisch visualisiert. Die visualisierten Szenen sollen danach bewertet werden, wie stark ein Ghosting-Effekt zwischen Objekt und Hintergrund auftritt. Die Bewertung soll von 0 (kein Ghosting-Effekt) bis 3 (starker Ghosting-Effekt) erfolgen. Die subjektiven Messwerte werden durch Ankreuzen der entsprechenden Bewertungsskala auf dem Bewertungsbogen dokumentiert.

**Testbefragung:**

Bild-Nr.	Bewertungsskala			
Test A	0	1	2	3
Test B	0	1	2	3
Test C	0	1	2	3

**Bewertung der Messreihe:**

Nr. 069	0	1	2	3
Nr. 070	0	1	2	3
Nr. 072	0	1	2	3
Nr. 073	0	1	2	3
Nr. 080	0	1	2	3
Nr. 081	0	1	2	3
Nr. 082	0	1	2	3
Nr. 084	0	1	2	3
Nr. 103	0	1	2	3
Nr. 104	0	1	2	3
Nr. 106	0	1	2	3
Nr. 107	0	1	2	3
Nr. 114	0	1	2	3
Nr. 115	0	1	2	3
Nr. 116	0	1	2	3
Nr. 118	0	1	2	3
Nr. 120	0	1	2	3
Nr. 121	0	1	2	3
Nr. 123	0	1	2	3
Nr. 124	0	1	2	3
Nr. 131	0	1	2	3
Nr. 132	0	1	2	3
Nr. 133	0	1	2	3
Nr. 135	0	1	2	3
Nr. 171	0	1	2	3
Nr. 172	0	1	2	3
Nr. 174	0	1	2	3
Nr. 175	0	1	2	3
Nr. 182	0	1	2	3
Nr. 183	0	1	2	3
Nr. 184	0	1	2	3
Nr. 186	0	1	2	3
Nr. 188	0	1	2	3
Nr. 189	0	1	2	3
Nr. 191	0	1	2	3
Nr. 192	0	1	2	3
Nr. 199	0	1	2	3
Nr. 200	0	1	2	3
Nr. 201	0	1	2	3
Nr. 203	0	1	2	3
Nr. 273	0	1	2	3
Nr. 274	0	1	2	3
Nr. 276	0	1	2	3
Nr. 277	0	1	2	3
Nr. 284	0	1	2	3
Nr. 285	0	1	2	3
Nr. 286	0	1	2	3
Nr. 288	0	1	2	3

Abbildung 9-2: Teil des Bewertungsbogens – Ghosting-Effekt