

# 3D @ Home

- Die plastische Dimension

Thomas Lustig

Sie ist Hauptbestandteil jedes guten Si-Fi Filmes - die künstliche Realität - *Virtual Reality*.

Um nun den Betrachter diesen künstlichen Raum wahrnehmen zu lassen, bedient man sich allerlei Tricks. Jede Umwelt - egal ob nun echt oder falsch - reizt die anatomischen Rezeptoren (Augen, Ohren, Haut). Zur Erzeugung dieser Reizung im *Virtual Space* bedient man sich einer entsprechenden Schnittstelle. Diese reicht vom einfachen Monitor, bis zu ausgeklügelten 3D-Systemen.

## 3D Beschleuniger - Der Rausch der Geschwindigkeit

Da sich der Benutzer auch bewegt und in seiner VR auch spazieren will, ist es nötig, die dem Auge zugeführten Bilder immer wieder neu zu berechnen. Dazu muss jedes hochauflösende Bild mit seinem



**Schritt 1:** Zuerst werden die sogenannten Vertices gebildet. Dabei handelt es sich um die Eckpunkte der Polygone, deren Position durch ihre X-, Y- und Z-Koordinaten bestimmt werden.



**Schritt 2:** Durch die Verbindung der Eckpunkte entstehen die Polygonkanten. Diese beiden Schritte, auch Geometrie-Setup genannt, werden zumeist noch vom Hauptprozessor berechnet.



**Schritt 3:** Jeweils drei Kanten werden fest einem Polygon zugeordnet und das Dreieck anschließend ausgefüllt (Flat Shading). Hier greift nun erstmals die 3D-Karte ins Geschehen ein.



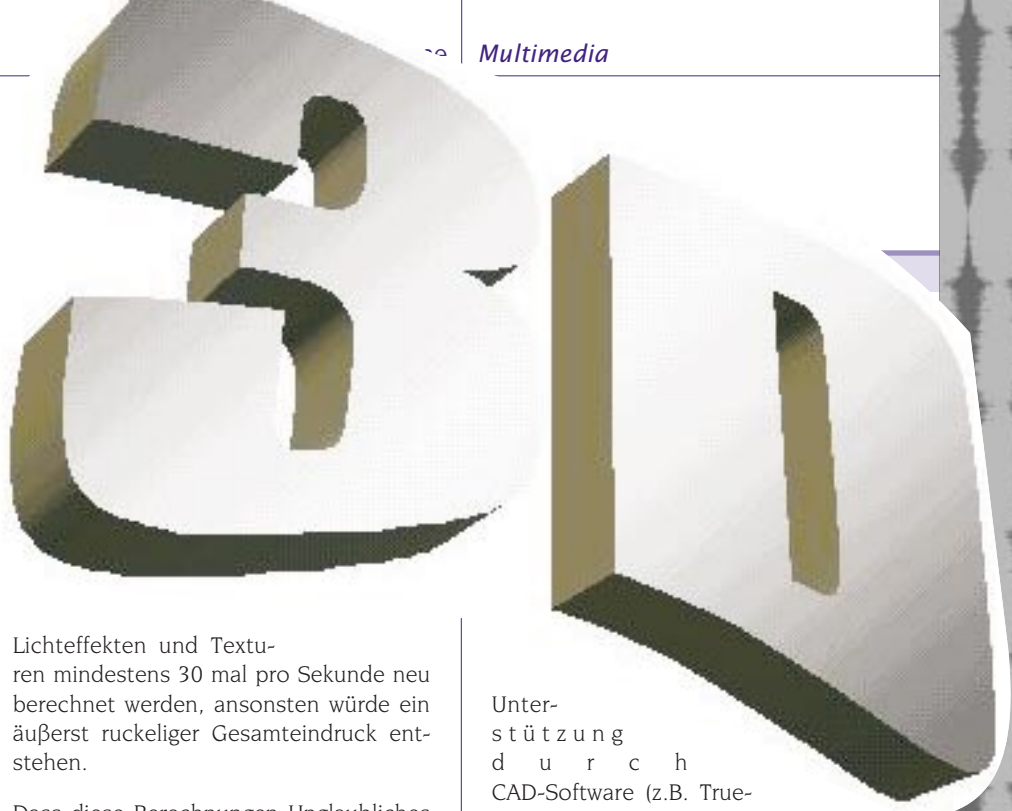
**Schritt 4:** Es folgt das Gouraud Shading, bei dem jeder Eckpunkt (Vertex) eines Polygons eine eigene Farbe zugeteilt bekommt und anschließend zu den Farbwerten der anderen Vertices überblendet wird.



**Schritt 5:** In diesem Schritt bricht für alle 3D-Beschleuniger Schwerstarbeit an: Es werden die Texturen über die Polygone gelegt und mit allerlei Effekten wie Filtering, Perspektivenkorrektur oder Antialiasing bearbeitet.



**Schritt 6:** Zusätzlich kann nun noch eine zweite Textur über das Objekt gelegt werden. Am gebräuchlichsten sind neben Lightmaps die Environment Maps, in denen sich scheinbar die Umgebung reflektiert oder spiegelt.



Lichteffekten und Texturen mindestens 30 mal pro Sekunde neu berechnet werden, ansonsten würde ein äußerst ruckeliger Gesamteindruck entstehen.

Dass diese Berechnungen Unglaubliches an Rechenpower fordern, ist klar. Deshalb sind im High-End-Bereich der VR-Anwendungen (z.B. für Architektur) immer noch spezielle Grafikkarten im Einsatz.

Aber zum Glück schläft die Industrie nicht und hat es in den vergangenen Jahren geschafft, VR in die Wohnungen des normalen Bürgers zu bringen - zu einem erschwinglichen Preis, versteht sich. 3D ist das Zauberwort der heutigen Zeit, und alles muss in 3D erhältlich sein - vom einfachen Gartenplaner bis zur Autorensimulation. Der Kunde wird mit dieser Ziffer-Buchstaben-Kombination beworfen, wo es nur geht, egal ob Grafik-, Soundkarte, etc..

## Deshalb wird es Zeit, einmal sachlich zu informieren.

Seit der Entwicklung des Computers stand bis jetzt immer die Steigerung der Prozessorleistung im Mittelpunkt. Waren zu Beginn Drahtgittermodelle das Höchste der Gefühle, so sind heute Schattierung, Texturen, Spiegelungen, komplexe Lichteffekte nur mehr Standard. Dabei wurde aber klar, dass es wesentlich intelligenter ist, Grafikberechnungen dorthin zu verlagern, wo sie auch hingehören - in die Grafikkarte. So hat heute ein moderner Grafikprozessor die Komplexität eines Pentium.

Den Verlockungen der neuen Grafikboards ist in der Tat schwer zu widerstehen. In Verbindung mit der hohen Rechenleistung eines K6 oder Pentium-II zaubern sie gefährliche Schlachten, packende Rennen u.v.m. in einer Grafikqualität auf den heimischen PC, die jede Playstation oder N64 verblasen läßt. Aber auch in ernsthafteren Bereichen sind diese 3D-Karten von hohem Nutzen. Durch die

Unterstützung durch CAD-Software (z.B. True-



space) ist es erstmals möglich, aufwendig texturierte Objekte verzögerungsfrei über den Bildschirm zu bewegen und auf Knopfdruck eine bildschirmfüllende Animation anzuzeigen, ohne diese vorher minutenlang vorberechnen zu lassen.

Die ersten Gehversuche der Industrie den Grafikkarten 3D-Funktionalität zu verleihen, waren leider erfolglos. Das lag nicht zuletzt daran, dass die Funktionen nur alibihafter hinzugefügt waren - um vor anderen Herstellern nicht schlecht dazustehen. Um den Kunden zu zeigen, was die Karte alles konnte, war meist ein kleines Demoprogramm oder Spiel beigefügt. Trotz Versicherung auf der Verpackung, weitere Software würde folgen, wurde der Kunde enttäuscht, denn die 3D Funktionen wurden von der Softwareindustrie nicht unterstützt, und jeder Grafikkartenhersteller kochte sein eigenes Süppchen.

Um aber die Kunden nicht vollends zu vergrämen, gab es später zu jeder Software - meist Spiele - sogenannte "Patches". Ihre Aufgabe war es, den Programmcode der Software so zu verändern, dass die Grafikfunktionen auf die Hardwarebeschleunigung umgeleitet werden konnten.

### Bald war es aber klar: Standards für Schnittstellen müssen her!

Die erste wirklich schnelle und leistungsfähige Schnittstelle ist heutzutage **OpenGL** - entwickelt von Silicon Graphics (SGI). Sie ist betriebssystem-unabhängig und findet häufig bei Unix-Betriebssystemen seine Anwendung, aber auch Windows NT verwendet sie. Sie bietet durch ihre Einfachheit den Programmierern einiges an Erleichterung. Bei High-End Applikationen wie CAD-Systemen (3DS Max) ist **Heidi** ein sehr beliebter Standard. Da für Windows95-User auch eine Schnittstelle her mußte, wurde **Direct3D** aus dem Hut gezaubert. Sie bietet in ihrer letzten Version ein gutes Maß an Leistung - das liegt daran, dass bei der neuesten Version eine Softwareemulation von OpenGL implementiert wurde. Direct3D belegt aber in puncto Geschwindigkeit einen niederen Rang und reicht leider nicht an OpenGL heran. Da sie aber standardmäßig in Windows95/98 implementiert ist, wird sie - zum Schaden der Windows-Anwender - die wahrscheinlichste Zukunft sein.

Die beliebteste Schnittstelle für den Spielerbereich ist **Glide**. Sie ist die Sprache des schon fast legendären 3D-Beschleunigers Voodoo von 3dfx. Durch die Geschwindigkeit von Glide ist ihre Unterstützung schon fast "Ehrensache" geworden.

### 3D-Begriffswelt

Gerade im 3D-Bereich schwirren nur so die Fremdwörter umher, und nur die wenigsten wissen, was sie eigentlich bedeuten. Die nun folgenden Erklärungen sollen Licht in den Wörterdschungel bringen. Zur besseren Ordnung sind die Begriffe von A bis Z geordnet.

#### AGP - Accelerated Graphics Port

Der von Intel entwickelte AGP stellt einen Steckplatz für eine einzelne Grafikkarte und per Definition keinen Bus dar. Es wird aus diesem Grund wohl auch nie ein Mainboard mit 2 AGPs geben. Der AGP wird mit 66 MHz, demnächst 133 MHz, getaktet, und ist somit gegenüber dem PCI Slot 2 - 4 mal schneller, da dieser heute üblicherweise mit 33 MHz läuft. Der Datenbus des AGP ist 32 Bit breit, so sind auf diesem ein Datendurchsatz bis zu 266 Mbyte/s möglich. Grafikkarten, die den X2 Mode beherrschen, können sogar bis auf maximal 533Mbyte/s kommen. Theoretisch, wäre der AGP also den PCI Slots haushoch überlegen.

AGP-Grafikkarten können über AGP 3D-Daten und Texturen im Arbeitsspeicher plazieren und das mit einer hohen Geschwindigkeit.

Allerdings ist hier zu erwähnen, dass AGP außer von Intel Grafikkarten von keiner 3D-Karte genutzt wird, weil die Zugriffe trotz neuer Technik noch um einiges langsamer sind, als auf den Onboard-Bildspeicher. Für die heutige Software reichen 16MB Texturspeicher, da die Texturdaten sowieso noch komprimiert werden und sollten einmal mehr Daten anfallen, so sind längst die 32Mb-Karten da.

#### Alpha-Blending

Verfahren für teilweise transparente Texturen zur Darstellung von Staub, Rauch, Wasser, u.s.w. Dazu bildet der Grafikchip die Mittelwerte aus den Farbwerten von Bildteilen, die bereits gezeichnet wurden und der durchscheinenden Textur.

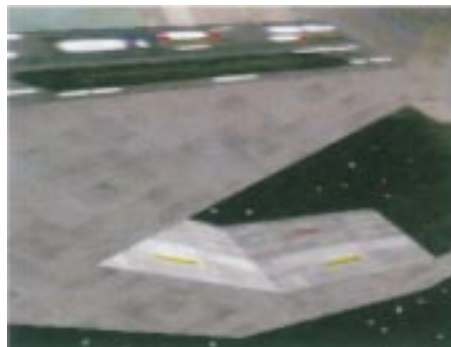
Dieses Verfahren ist aufwendig, weil zu den Zugriffen auf den Z-Buffer und Texturspeicher zusätzliche Zugriffe auf den Bildspeicher erfolgen (das bezeichnet man auch als *multi pass texture mapping*). Einige Grafikchips (Matrox, Permedia) vermeiden die Lesezugriffe auf den Bildspeicher dadurch, dass sie jeden zweiten Punkt des Hintergrundes mit der transparenten Textur überschreiben (-> *stippled alpha blending*). Dabei entstehen die Farbmittelwerte durch das Verschmelzen des Punktrasters erst beim Betrachten. Die Bildqualität läßt hierbei allerdings zu wünschen übrig.

#### Anisotrope Texturfilterung

Ist eine neue Technik, um das Aussehen von Texturen bei der Betrachtung unter einem flachen Winkel zu verbessern. Damit bleibt auch beispielsweise ein Schriftzug auch an extrem schrägen Flächen noch immer lesbar.

#### Anti-Aliasing

unterdrückt die so genannten Treppeneffekte an schräg verlaufenden Kanten durch Bildung einer Mischfarbe aus Vorder- und Hintergrund. Durch den extremen Rechenaufwand wird es nur selten verwendet (nächste Grafikkartengeneration - z.B. Permedia 3 schaffen es allerdings



in einem Rechenschritt Fullscreen). Der Trend geht eher zu höheren Auflösungen.

#### Back Face Culling

Dieser Prozeß berechnet verdeckte Flächen, die sich aus der Betrachtungsperspektive ergeben. Jedes zu zeichnende Objekt, dessen Vorderseite nicht zu sehen ist, wird nicht dargestellt.

#### Bilineare Filterung

Wenn ein Polygon samt Textur nah an den Betrachter heranzoomt, werden die einzelnen Pixel enorm groß, die Szenerie wirkt dadurch sehr grob und unrealistisch. *Bilinear Filtering* verhindert die Bildung großer, einfarbiger Pixelblöcke, indem es die Farbwerte angrenzender Bildpunkte vermischt und so weiche, fließende Übergänge schafft.

Beim Aufbringen einer Textur auf entfernte Objekte überdeckt ein Pixel meist mehrere Textur-Bildpunkte (*Texel*). Um durch Aliasing-Effekte verursachtes Flimmern zu verhindern, bildet man den Mittelwert aus vier oder mehr benachbarten Texeln.

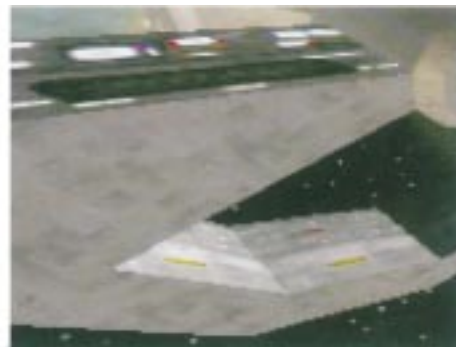
Je nach Filterimplementierung des Chips, Konfiguration des Treibers und Entfernung des Objekts werden zwischen einem Texel (*filter to nearest*, also nicht interpoliert), 2-4 Texel (linear interpoliert), 4-12 Texel (bilinear interpoliert) und 6-24 Texel (trilinear interpoliert) verwendet. Üblich sind 4 Texel ohne *Mip-Mapping* oder 8 Texel mit *Mip-Mapping* (trilinear). Über mehr Texel zu interpolieren, macht Übergänge weicher, benötigt aber auch mehr Speicherzugriffe.

#### Bump - Mapping

Durch Modulation der Beleuchtung läßt sich der Eindruck von zusätzlichen räumlichen Details erwecken. Die Modulationswerte entnimmt der Pixel-Prozessor der zweiten Textur.

#### Clipping

Bei diesem Prozess wird jedes Polygon überprüft, ob es teilweise sichtbar ist, oder ganz außerhalb des Zeichenbereichs liegt. Die nicht sichtbaren Flächen oder Teilbereiche des Objekts werden entfernt.



#### Color-Key-Transparenz

Um komplexe Objekte darzustellen, definiert man eine beliebige nicht verwendete



Texturfarbe als transparent (Color Key). Dadurch läßt sich beispielsweise die Transparenz eines Gitters erzeugen u.s.w. Allerdings kann es hier durch fehlerhafte Grafikkartentreiber zur Falschdarstellung kommen.

#### Depth Cueing

Die Intensität eines Objektes läßt nach, sobald es sich vom Betrachter entfernt. Entspricht im Prinzip -> *Fogging*, bei dem nach Schwarz überblendet wird.

#### Dithering

Bei Farbtiefen unter 24 Bit pro Pixel gehen beim Rendern Farbinformationen verloren. Beim Dithern wird versucht diesen Farbfehler über mehrere Pixel zu verteilen, um so mit den Farben benachbarter Pixel Zwischenfarben zu simulieren. Die beiden gängigen Verfahren addieren entweder die fehlenden Farbanteile auf einen benachbarten Pixel (relativ genau), oder sie runden die Farbwerte je nach Position des Pixels unterschiedlich (schnell).

#### Double Buffer

Um den Aufbau einer 3D-Darstellung zu verdecken, läßt man den Grafikchip in einem nicht sichtbaren Bereich des Bildspeichers (*Back Buffer*) zeichnen. Ist der Bildaufbau abgeschlossen, überträgt man die Daten in den sichtbaren Teil des Bildspeichers (*Front Buffer*) oder schaltet zwischen den beiden Buffern um (*Page Flipping*). Durch Synchronisieren mit dem Bildwechsel beseitigt man letzte Störungen (*Tearing*), verliert wegen der entstehenden Wartezeiten aber etwas Performance. Einige Treiber verwenden deshalb drei Puffer in Rotation, einen zum Zeichnen, einen zum Darstellen und einen für das *Page Flipping*. Da man noch Speicher für den Z-Buffer und Texturen braucht, geht das natürlich nur bei kleinen Auflösungen oder mit viel Speicher.

#### Flat Shading / Gouraud Shading

Während beim *Flat Shading* ein Dreieck einheitlich gefärbt ist, besitzt beim *Gouraud Shading* jede Ecke eine eigene Farbinformation. Aus der Zwischenberechnung dieser Eck-Farbwerte ergibt sich ein sehr weicher Verlauf, der sogar mit weniger Dreiecken für deutlich bessere Ergebnisse sorgt als das *Flat Shading*.

#### Fogging

Die Pixel eines Objektes werden dabei mit einer festen Farbe vermischt, deren Intensität mit wachsender Entfernung zunimmt. Sorgt bei Mischung mit Weiß zum Beispiel für atmosphärische Nebel-effekte. *Fogging* spart außerdem Rechenzeit, da Körper erst ab einer bestimmten Distanz gezeichnet werden müssen.

#### fps

Frame Rate. Diese Einheit gibt eigentlich nur an, wie viele "Bilder pro Sekunde" (*frames per second*) berechnet werden. Für das menschliche Auge erscheinen Bewegungsabläufe ab etwa 15 fps flüssig. Kinofilme laufen mit 24 fps, das Fernsehen arbeitet mit 25 fps. Hier gilt:

#### Je mehr fps desto besser.

#### Hidden Surface Removal

Es wird überprüft, welche Oberflächen der Polygone für den Benutzer im Endeffekt sichtbar sein werden. Nur diese werden dann gerendert, um Rechenzeit zu sparen.

#### MIP-Mapping

MIP steht für "*multum in parvo*" (viele unter gleichen) und meint die Speicherung einer Textur in verschiedenen Größenabstufungen. Man nutzt hochaufgelöste Bitmaps für nahe Objekte und kleinere für entferntere. Beim trilinearen *MIP-Mapping* wird zunächst innerhalb der nächst größeren und der nächst kleineren Textur jeweils bilinear gefiltert und schließlich zwischen diesen beiden Texeln noch einmal gemittelt. Auf diese Weise lassen sich Klötzcheneffekte im Nahbereich und Aliasing bei entfernten Objekten sehr wirkungsvoll unterdrücken.

#### Perspektivenkorrektur

Bei der Verwendung von gleichen Texturen kann es vorkommen, dass Texturen vom Muster her falsch aneinander gelegt werden. Diese Funktion korrigiert den Fehler.

#### Shading

Damit Texturen einen realistischeren Eindruck auf das menschliche Auge machen, werden unterschiedliche Schattierungsverfahren angewandt, die einen Farbverlauf erzeugen. Bekannte Verfahren sind: *Flashshading*, was aber mittlerweile nicht oder nur noch sehr selten angewandt wird (hier wird pro Fläche eine Schattenfarbe berechnet) und das heute übliche *Gouraudshading* (hier wird ein natürlicher Schattenverlauf ermöglicht).

#### SLI - Scan Line Interface

Durch dieses System ist es möglich zwei 3D Beschleuniger miteinander zu "kopeln". Dabei wird das Bild auf die 2 Karten aufgeteilt. Eine Karte berechnet die Zeilen 1,3,5,.. und die Zweite die Zeilen 2,4,6.. Die Leistungssteigerung beträgt damit 100%.

#### Specular Highlights

Glanzlichter. Unter anderem Sonnenstrahlen die von Metall oder Glas reflektiert werden. Achtung! Rechenaufwendig!

#### Stencil Buffer

Ein *Stencil Buffer* dient zur unregelmäßig geformten Begrenzung beziehungsweise

Maskierung des Zeichenbereichs. Der Grafikchip erzeugt 3D-Objekte beispielsweise nur innerhalb einer Windschutzscheibe, ohne dass die Software die Lage der Objekte zum Fensterrand überprüfen muß.

#### Tessellation

Ein Objekt wird in eine Vielzahl von Polygonen bzw. Dreiecken zerlegt. Die Eckpunkte der Dreiecke werden mittels Koordinaten (x, y und z) beschrieben wobei der Wert "z" die Tiefeninformation enthält. Diese Punkte enthalten je nach Darstellung zusätzlich noch Informationen über Material und Textur. Durch diese Umrechnung der Bildinformation erhöht sich die zu verarbeitende Datenmenge immens.

#### Texel

Grafikkarten speichern Texturen in Form einer rechteckigen Anordnung von Bildpunkten. Allerdings wird nicht unbedingt jeder gespeicherte Bildpunkt auch angezeigt. Deshalb unterscheidet man zwischen den angezeigten Bildpunkten und den Textur-elementen (= Texel).

#### Texture-Cache

Der Zugriff auf den Texturspeicher erfolgt entweder in 64 Bit breiten Worten und mehr oder in so genannten *Bursts*. Dadurch werden mehr Daten geliefert als im Moment verarbeitet werden können. Der Texturen Cache (in die 3D CPU integriert) erhöht die Zugriffsgeschwindigkeit, ähnlich wie der *Level 1 Cache* des Computers.

#### Texture - Mapping

Überlagerung eines Polygons mit einem perspektivisch korrekt dargestellten Textur.

#### Texturing

Muster wie auch Tapeten, Ziegelwänden oder Zebras, die 3D - Beschleuniger auf beliebig geformte 3D - Körper projizieren können. Ohne geeignete Wand-, Boden-, Deckenverzerrungen wäre ein Spiel äußerst kahl. Um dies zu ändern setzt man Texturen ein, die dann auf Polygone "geklebt" werden.

#### Trilineares Filtering

Um zwischen zwei MIP-Maps weichere Übergänge zu schaffen, fließen in die Berechnungen eines Bildpunktes noch die Farbwerte der beiden Maps mit ein, d.h. es wird zwischen den beiden Maps interpoliert, was die nötigen Speicherzugriffe verdoppelt. Das ergibt die besten Resultate hinsichtlich der Bildqualität, ist aber auch sehr rechenaufwendig.

#### Video Mapping

Variante des *Texture Mappings*, bei der die Polygone mit einer animierten Textur (z.B.

einer AVI- oder MPEG-Datei) beklebt werden.

### Z-Buffer

Der sogenannte Z-Buffer speichert die Positionierung eines Pixel auf der Z-Achse (also in der Tiefe des Raumes). Bei gleicher Bildschirmposition mehrerer Pixel wird aufgrund der Tiefeninformation entschieden, ob der jeweilige Bildpunkt für den Betrachter sichtbar wäre und deshalb zu zeichnen ist oder nicht. Auf Hochleistungs-3D-Grafikhardware ist der Z-Buffer meist als separates EDO-DRAM ausgeführt (*Local Buffer*). Bei komplexen Objekten sollte der Grafikchip Z-Werte mit einer Genauigkeit von 24 oder 32 Bit verarbeiten, um Darstellungsfehler bei feinen Details zu vermeiden. Die für Spiele entwickelte Windows 95-Schnittstelle DirectX arbeitet nur mit 16 Bit.

## 3D-Beschleuniger statt neues Motherboard

3D Beschleuniger sind eine äußerst preiswerte Alternative zum Kauf eines neuen Prozessors im Grafikbereich geworden. Sie sind als Beschleuniger + Grafikkarte - Kombination, aber auch als sogenanntes *Add-On* verfügbar. Bei einer *Add On*-Karte wird das Signal der vorhandenen Grafikkarte durch den Beschleuniger durchgeschleift. Werden die 3D Funktionen des Beschleunigers benötigt, so schaltet die Treibersoftware um. Für eine gute 3D-Karte werden heute zwischen 1400 - 2300 ATS dem Käufer abverlangt, während ein neuer Prozessor mit Motherboard zu einer beachtlichen Belastung der Brieftasche werden kann und dann doch nicht in der Lage ist, den Benutzer mit allerlei Effekten und Geschwindigkeitsvorteil zu erfreuen.

### 3Dnow!

Im Gegensatz zu der hochgejubelten MMX-Befehlserweiterung von Intel-Prozessoren existiert eine viel nützlichere Erweiterung von AMD. Sie nennt sich **3Dnow!** und ist nicht als Konkurrenz einer 3D-Karte, sondern viel mehr als Verbündeter anzusehen. Die 3Dnow!-Schnittstelle ist in allen neuen AMD K6 II Prozessoren integriert und dient zur Entlastung der eingebauten Beschleunigerkarte. Die zum Rendern benötigten Instruktionen für die 3D-Karte müssen unbedingt vorher vom Systemprozessor aufbereitet werden, um auch "verstanden" zu werden. Der 3Dnow!-Teil übernimmt diese Aufgabe und erreicht damit eine Leistungssteigerung von bis zu (realistischen!) 50%. Im Gegensatz zu MMX ist hier keine Totgeburt zu erwarten, da die Softwareentwickler äußerst positiv eingestellt sind und schon einige sehr gute Pro-

gramme mit der nötigen Unterstützung verfügbar sind.

Beim Kauf eines 3D-Beschleunigers sollte man sehr genau darauf achten welche Schnittstellen unterstützt werden (Glide, OpenGL,...). Gerade auf dem Spielsektor ist es wichtig, dass die Investition in solch eine Karte nicht in einem Inselsystem endet.

Als sehr wichtiges Leistungsmerkmal einer Karte gilt hier die Füllrate (Pixel/s). Sie sollte bei 80 MegaPixel/s liegen. Zum Zweiten ist vor allem auf die unterstützten 3D-Funktionen zu achten (siehe ABC). Die Tests in diversen Zeitschriften und Magazinen sind äußerst informativ, aber trotzdem mit Vorsicht zu genießen. Auf dem Markt befinden sich Karten mit enormen Preisunterschieden, die einen genauen Vergleich unbedingt nötig machen. Meistens werden günstige Karten mit gleichem Beschleunigerchipsatz angeboten (z.B. Maxi Gamer mit Voodoo 2 Chipset).

Die Entwicklung der Beschleuniger geht sehr rasant weiter und fast täglich liest man Gerüchte von neuen Wunderchips, die aber meistens nur den Börsenkurs der Firma ankurbeln sollen.

In sehr naher Zukunft erwartet uns, b.z.w. sind schon auf dem Markt erhältlich:

- Permedia 3 : von Creative Labs entwickelt
- Riva TNT : SGS-Thomson/nVidia
- Rendition V3300
- PowerVR 2 - second generation : Produkt aus dem Hause NEC - werkelt übrigens in Segas Dreamcast (neue Spielkonsole)
- Ticket 2 Ride 4: Number 9
- Voodoo 3 : von 3Dfx und ist bereits in den Startlöchern.

Ferner wären hier:

- VelaTX: in den Laboren von Stellar Semiconductor
  - Glaze 3D: Projekt der finnischen Entwickler Bitboys Oy. (Füllrate bis 400 MegaPixel/s)
  - Im SLI Betrieb ( 2 Karten -> siehe ABC) bis 800 MegaPixel/s).
- zu erwähnen.

### 3D Sound

Die nächste Zeit wird von 3D gesegnet sein, und man ist schon heute in der Lage, sich die Spielhalle und die teuren Grafikkarte in einer Consumer-Variante ins Wohnzimmer zu holen. Gerade auf den Spielsektor läßt sich durch *Dolby Surround*-Unterstützung auch der Ton plastischer wahrnehmen. Vor Pseudo-3D-Soundsystemen wie sie von Billig-soundkarten-Herstellern angepriesen werden kann nur abgeraten werden. Sie arbeiten mit einem psychoakustischen Effekt und erzeugen einen nicht ortbaren

Klang (*Virtual Sound*) - ABER ein echter Lautsprecher ist durch keine Elektronik zu simulieren.

### 3D Brillen / Cyberhelme

Für die realistische Darstellung eines 3D Bildes ist ein Monitor leider unbrauchbar. Zur optimalen Darstellung wird dem linken und rechten Auge ein separat berechnetes Bild zugeführt - ähnlich wie in einem 3D Kino. Das geschieht mittels 2 kleinen LC -Displays und läßt dem Benutzer glauben, von der künstlichen Welt umgeben zu sein.



Diese Technologie findet leider nur im professionellen Bereich Verwendung. Aus dem privaten Sektor sind solche Systeme (i-glasses, VFX1,...) wegen ihrer schlechten Qualität und des hohen Preises wieder verschwunden. ABER Forte Inc. arbeitet an einem neuen Cyberhelmsystem - bleibt zu hoffen, dass hier etwas Ordentliches entsteht.

### Force Feedback

Um Objekte auch "greifbar" zu machen und einen fühlbaren Eindruck zu erzeugen, wird meistens durch Motoren in den Eingabegeräten eine "Gegenkraft" erzeugt (Force Feedback).



Damit ist es möglich z.B. das Ruckeln der Lenkung eines Rallye Wagens zu simulieren.

Die fatalen Auswirkungen durch diese komplette Simulationen zeigen uns diverse Sci-Fi-Filme. Aber vielleicht muss es ja auch nicht so enden und wir können die Systeme auch einsetzen, um uns zu entspannen und für ein paar Stunden in andere Welten zu tauchen. Auch hier gilt allerdings:

Nichts ist so gut, wie das Original ☺