

Was Sie über 3D-Effekte wissen sollten

3D: Die Effekte

Alpha Blending, bilineares Filtern, Mip-Mapping – wer eine 3D-Grafikkarte kauft, muß sich mit vielen Fachbegriffen herumschlagen. Was es damit auf sich hat, erfahren Sie hier

Die Zeiten, da eine schnelle CPU für gängige Spiele ausreichte, sind vorüber. 3D-Grafik ist „in“. Heute brauchen Sie eine leistungsstarke CPU und einen guten 3D-Beschleuniger, um Spiele in hoher Qualität ruckelfrei zu genießen.

Vermeiden Sie einen Fehlkauf. Nicht jede 3D-Karte beherrscht alle wichtigen Effekte. Beispiele: die Millennium II und Mystique 220 von Matrox. Sie können kein Fogging, kein Mip-Mapping, kein bilineares Filtern. Nutzt ein Spiel diese Funktionen, ist entweder die Darstellung schlecht, oder – falls die Funktio-



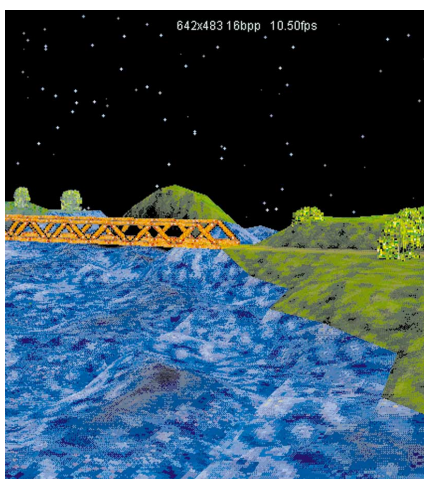
Eine gut gemachte 3D-Grafik ist aufwendig: Nur mit geübtem Blick ist erkennbar, wieviel Know-how dahintersteckt, daß diese Szene so realistisch wirkt

nen ersatzweise per Software emuliert werden – es gibt Tempo-Einbußen.

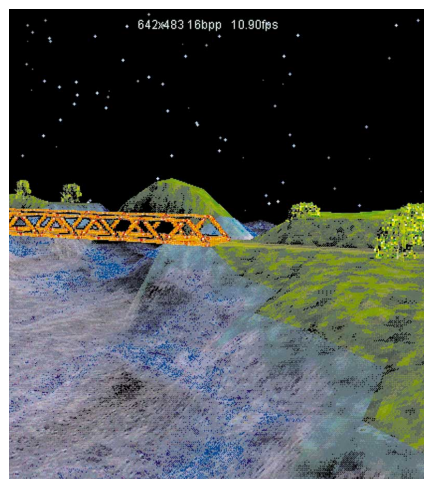
Ein Bild sagt mehr. Wir erklären die 12 wichtigsten 3D-Funktionen. Jeden Effekt veranschaulichen wir mit zwei Bildern. Zur besseren Übersicht haben wir die Punkte alphabetisch geordnet. Da das Prinzip „Textur“ aber so wichtig ist, möchten wir hier ein wenig vorgreifen (siehe auch Punkt „Textur-Mapping“, Seite 184).

Heutige 3D-Karten behandeln Form und Aussehen eines Gegenstandes getrennt. Beispiel: Die Erde könnte man durch Millionen von Polygonen darstellen, von denen jedes eine bestimmte Farbe hat. Der Rechenaufwand wäre enorm. Oder man stellt die Erde durch eine Kugel mit ein paar Hundert Polygonen dar, auf die nur ein Bild der Erde projiziert wird. Dieses Bild heißt Textur.

RICHARD COPPOLA / RER



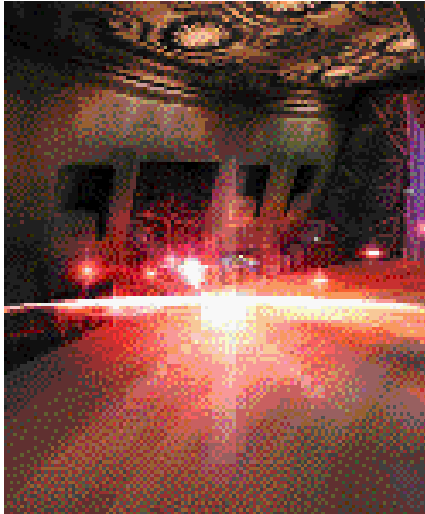
Ohne Alpha Blending: Der Grund des Sees ist nicht zu sehen



Mit Alpha Blending: Die Oberfläche des Wassers erscheint lichtdurchlässig

Alpha Blending

Dieser Effekt gestattet es, auch teilweise durchsichtige Objekte – etwa Wasser oder Fensterglas – darzustellen. Damit ein Texturpunkt transparent erscheint, wird der üblichen Bildinformation pro Pixel ein weiterer Wert hinzugefügt – der „Alpha-Kanal“. Damit hat der Programmierer nicht nur die Farben Rot, Grün und Blau (RGB), sondern zusätzlich den Alpha-Wert (RGBA) zur Verfügung. Häufig wird die Darstellung RGBA 4444 verwendet, das bedeutet, daß jeder Wert mit einer Genauigkeit von 4 Bit verarbeitet wird. Die Transparenz – und natürlich auch die einzelne Farbnuance – läßt sich damit in 16 Stufen fein dosieren. ►



Ohne bilineare Filterung: Texturen im Vordergrund wirken blockartig



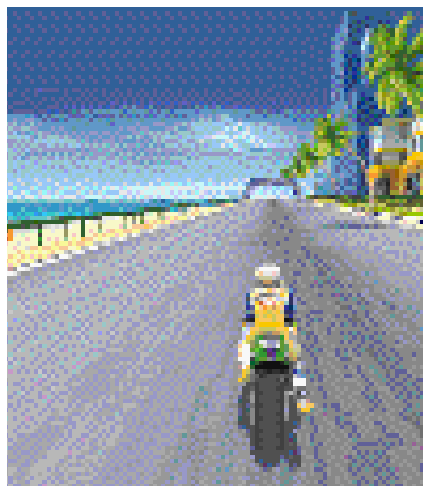
Mit bilinearer Filterung: Das Bild ist unschärfer, aber die Blöcke sind weg

Bilineares Filtern

Hierbei handelt es sich um einen Weichzeichner, der auf Kosten der Bildschärfe geht. Er ist dennoch sinnvoll, wenn Texturen stark vergrößert werden. Denn dann müssen neue Bildpunkte „erfunden“ werden. Dazu werden gewöhnlich vorhandene Punkte kopiert, was zu Blöcken mit gleicher Bildinformation führt. Die bilineare Filterung ist ein Tiefpaß-Filter, der scharfe Übergänge verwischt – so verschwinden die Blöcke. Dazu werden die Daten aus den vier nächsten Texturpunkten interpoliert – jeweils eine lineare Interpolation für zwei Punkte. Das führt zu zwei Ergebnissen, die wiederum linear interpoliert werden – daher der Name „bilineares Filtern“.



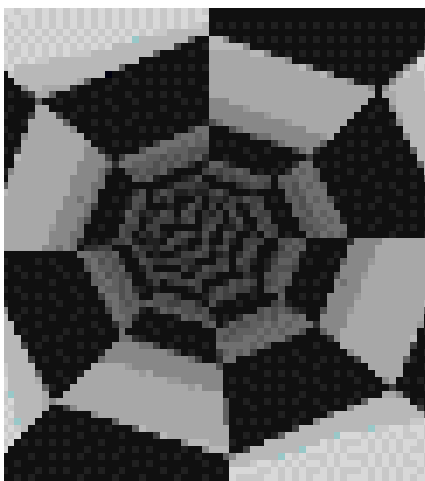
Ohne Color-Key: Zaun und Palmen haben schwarze Zwischenräume



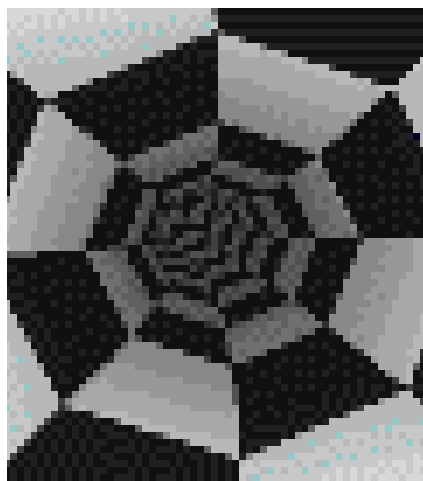
Mit Color-Key: Hier wird die Farbe Schwarz als transparent definiert

Color-Key

Diese Funktion kommt dann zum Zuge, wenn Zäune und Bäume darzustellen sind. Sie macht einzelne Texturpunkte durchsichtig, so daß sich komplizierte Objekte leichter darstellen lassen. Statt etwa einen Zaun und alle seine Zwischenräume mit sehr vielen Polygonen zu zeichnen, genügt eine relativ einfache Form wie ein schmaler Quader. Auf dessen Oberfläche malt man die sichtbaren Stellen. Die Flecken, die den Zwischenräumen des Zauns entsprechen, füllt man mit einer als durchsichtig definierten Farbe. Im Unterschied zum Alpha Blending gibt es hier aber keine stufenweise Transparenz: Entweder ist ein Punkt durchsichtig oder nicht.



Ohne Dithering: Die grauen Flächen haben einen stufenartigen Farbverlauf



Mit Dithering: Durch diese Technik wirken die Farbverläufe gleichmäßig

Dithering

Dithering sorgt dafür, daß die unterschiedlichen Stufen eines Farbverlaufs nicht zu sehen sind. Ein stufenloser Verlauf läßt sich eigentlich nur mit 16,7 Millionen Farben darstellen, da dies in etwa dem farblichen Auflösungsvermögen des menschlichen Auges entspricht. Je weniger Farben, desto auffälliger die Abstufungen. Aber viele Farben kosten Speicher. Dithering erhöht scheinbar die Anzahl der darstellbaren Farben, indem es das begrenzte örtliche Auflösungsvermögen des menschlichen Auges nutzt: Sind einzelne Bildpunkte so nah beieinander, daß sie sich nicht getrennt erfassen lassen, wird ihre Information vermischt. ►

Was Sie über 3D-Effekte wissen sollten



Ohne Environment-Mapping: Die Heckscheibe des Rennwagens ist schwarz



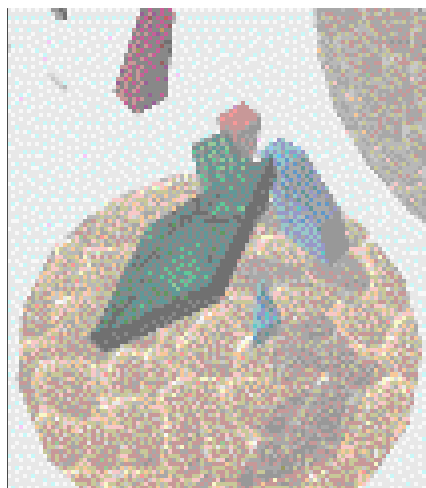
Mit Environment-Mapping: Der Himmel spiegelt sich in der Heckscheibe wider

Environment-Mapping

Dieser Effekt simuliert reflektierende Oberflächen. So spiegelt sich zum Beispiel der Himmel oder die Umgebung in einer Glasscheibe wider – ganz wie in der Realität. Anders als bei einer statischen Textur wird beim Environment-Mapping (englisch: Abbildung der Umwelt) das Aussehen der reflektierenden Oberfläche an die Umgebung angepasst. Führt das Auto beispielsweise langsamer, werden auch die vorbeiziehenden Wolken, die sich in der Heckscheibe spiegeln, langsamer abgebildet. Führt das Auto durch einen Tunnel, sind Wolken fehl am Platz. Dafür spiegelt sich die Tunnelbeleuchtung wider – natürlich ebenfalls an die Geschwindigkeit des Autos angepasst.



Ohne Fogging: Die Luft ist klar, sogar der Hintergrund ist gut zu sehen



Mit Fogging: Diese trübe Suppe läßt den Hintergrund völlig verschwinden

Fogging

Fogging gehört zu den atmosphärischen Effekten und verstärkt den Tiefeindruck einer 3D-Szene – schließlich ist die Sicht, etwa von einem Berg aus, oft durch Dunst oder Smog eingeschränkt. Geschickt eingesetzt, kann Nebel auch den Spielablauf beschleunigen – ein im Nebel verschwundenes Objekt muß nicht dargestellt werden. Grundsätzlich gibt es zwei Techniken: Fog Table und Fog Vertex. Fog Vertex ist Standard und vielseitiger, da hier der Programmierer jedem Polygon einen Fog-Wert zuordnet. So läßt sich etwa der Dampf über kochendem Wasser zeigen. Fog Table wirkt realistischer, eignet sich aber nur für Nebel-Effekte.



Flat Shading: Wegen der vielen Ecken wirkt der Kopf unrealistisch



Gouraud Shading: Die unerwünschten Ecken sind weg – der Kopf wirkt rund

Gouraud Shading

Dieses Verfahren verhindert den Effekt, daß Objekte eckig wirken – ein Nachteil einfacher Computergrafiken. Der Grund für die häßliche Blockbildung ist die Berechnungsweise der Lichtwerte (englisch: Shading). Wenn pro Polygon nur ein Lichtwert gilt, spricht man von Flat Shading. Diese Art der Beleuchtung kostet wenig Rechenzeit, allerdings ist dabei der Umriß des Polygons zu sehen. Anders beim Gouraud Shading: Die Beleuchtung wird für jeden Eckpunkt des Polygons berechnet. Die Werte für die übrige Polygonfläche ergeben sich durch Interpolation. So entstehen gleichmäßige Verläufe, die Polygonränder sind unsichtbar. ►

Was Sie über 3D-Effekte wissen sollten



Ohne Lens Flaring: Die Sonne scheint, aber es entsteht kein optischer Effekt



Mit Lens Flaring: Die Sonne erzeugt Lichtreflexe in der Auto-Frontscheibe

Lens Flaring

Dieser Effekt vermittelt dem Spieler den Eindruck, als würde er die Szene durch eine Glasscheibe, ein Objektiv oder eine Brille beobachten. Das steigert die Wirklichkeitstreue eines 3D-Spiels – schließlich sitzt man hier ja meist in einem Auto oder Raumschiff mit Frontscheibe. Beispiel: Trifft eine helle Lichtquelle wie die Sonne in einem bestimmten Winkel auf ein Objektiv, entstehen an der Blende deutlich sichtbare Reflexionen. Die Lichtreflexe können rund sein oder der Blendenform entsprechen (etwa fünfeckig), das ist vom Objektiv abhängig. Auch in der Frontscheibe eines Autos oder in einer Brille entstehen bei entsprechender Lichteinstrahlung Reflexe.



Ohne Mip-Mapping: Die Struktur des Fußbodens hat irritierende Schlieren



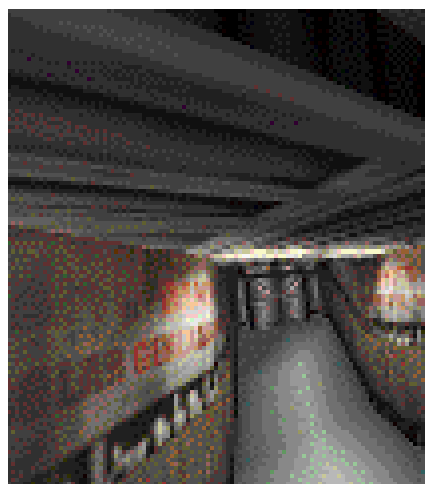
Mit Mip-Mapping: Der Fußboden zeigt jetzt deutlich horizontale Streifen

Mip-Mapping

Damit lassen sich nahe und ferne Gegenstände einigermaßen korrekt darstellen. Mip steht für „multum in parvo“ (viele im Kleinen) und bedeutet, daß Texturen in verschiedenen Auflösungen verwendet werden. Diese sind von einem hochwertigen Grafikprogramm vorskaliert und werden vom schlechteren Skalierer im Grafikchip nur noch marginal verändert. Folge: weniger Fehler, wenn die Texturgröße geändert wird. Ansonsten können regelmäßige Muster (Aliase) erscheinen, die im ursprünglichen Bild nicht vorhanden sind. Mip-Mapping verhindert das Auftauchen dieser Muster, ist also eine Form des Anti-Aliasing.



Ohne perspektivische Korrektur: Die Wandschrift links ist stark verzerrt



Mit perspektivischer Korrektur: So ist die Schrift gerade und lesbar

Perspektivische Korrektur

Damit wird verhindert, daß Gegenstände, die sich vom Vorder- in den Hintergrund erstrecken, verzerrt wirken. Das unrealistische Auf und Ab entsteht, wenn die Texturen, die ein Polygon bedecken, mit nur einem Faktor skaliert werden. Das führt zu einem unrealistischen Größensprung der Texturen von einem Polygon zum nächsten. Die Lösung ist einfach, aber rechenaufwendig: Die Texturen müssen hinten kleiner sein als vorne. Und genau das bewirkt die perspektivische Korrektur: Dabei werden einzelne Texturen, perspektivisch richtig, abhängig von ihrer Entfernung zum Betrachter skaliert.

Was Sie über 3D-Effekte wissen sollten



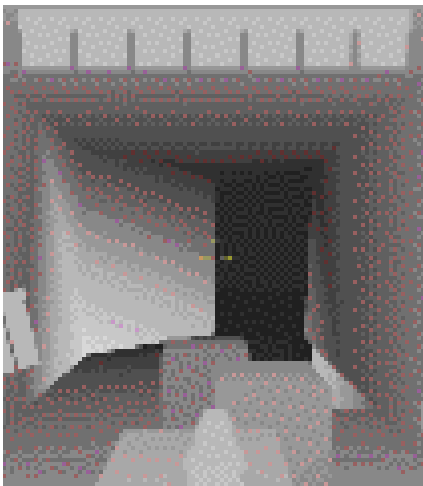
Ohne Schatten: Das Bild sieht ja ganz gut aus, aber irgend etwas fehlt



Mit Schatten: So ist's realistisch. Das Auto hat nun einen Schatten

Schattenbildung

Diese Funktion gehört zur Gattung der Beleuchtungseffekte, denn damit werden Schatten berechnet. Ein gut gemachter Schatten darf in einer realistischen Szenerie nicht fehlen. Aber der Effekt ist nicht rein kosmetischer Natur. Geschickt eingesetzt dient er dem Spieler als eine zusätzliche Informationsquelle. Ein Schatten hilft uns beispielsweise zu erkennen, wie weit ein Gegenstand vom Untergrund entfernt ist – etwa bei einem Flugzeug, das im Tiefflug über eine Gebirgskette gleitet. Oder er warnt uns, wenn sich ein Feind von hinten nähert. Nur bei völlig diffusem Licht gibt es keine Schatten – und bei einem schlecht gemachten Computerspiel.



Ohne Textur-Mapping: Die Objekte sind zwar alle da, aber nur schemenhaft



Mit Textur-Mapping: Erst jetzt werden aus Formen erkennbare Gegenstände

Textur-Mapping

Dieser Effekt macht aus farblosen Flächen realistische Gegenstände. Statt komplizierte Objekte in allen Einzelheiten zu verarbeiten – das kostet wertvollen Speicherplatz und Rechenzeit – werden einfache und farblose dreidimensionale geometrische Gebilde mit einem zweidimensionalen Äußeren – der Textur – überzogen. Beispiel: Um ein Hochhaus von außen zu zeigen, genügen ein einfacher Quader und eine Textur pro Seite des Hochhauses. Da das Haus auf der Erde steht, gibt es nur fünf sichtbare Flächen. Soll jede anders aussehen, brauchen wir folglich fünf Texturen. Diese werden auf die jeweilige Seite des Quaders projiziert. ■