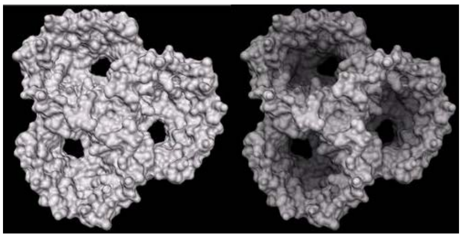
3D-Viewing

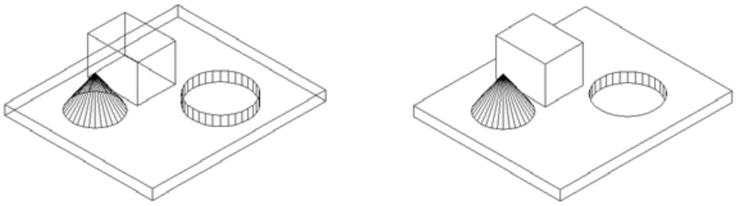
Konzepte des 3D-Viewing

Um 3D-Objekte erkennbar oder schön auf 2D-Geräten wiederzugeben, werden viele verschiedene Techniken kombiniert eingesetzt. Zuerst einmal muss man eine Projektion definieren, das wird im nächsten Absatz behandelt. Nach dieser Projektion können unter Anderem folgende Qualitäten erzeugt werden:

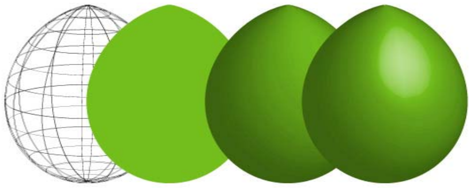
Drahtgitterdarstellung = Es werden nur die Kanten aller Polygone gezeichnet, wobei verdeckte Kanten entweder sichtbar oder unsichtbar sein können.



Depth-Cueing = Kanten oder Teile, die näher zum Betrachter liegen, werden intensiver dargestellt (heller, breiter, gesättigter), Kanten, die weiter weg liegen, weniger intensiv (dunkler, dünner, grauer).

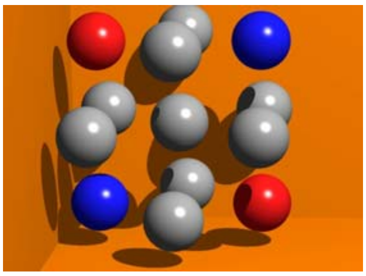


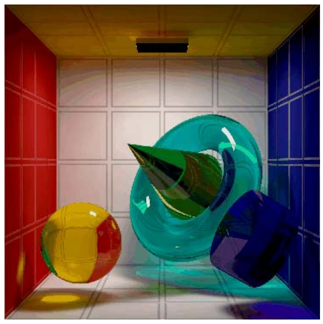
Korrekte Sichtbarkeit = Oberflächenelemente (Kanten, Polygone), die von anderen Oberflächen verdeckt werden, werden nicht gezeichnet, sodass man nur sichtbare Bereiche sieht.



Schattierung = in Abhängigkeit des Blickwinkels oder Lichteinfallswinkels werden Oberflächen heller oder dunkler gefärbt.

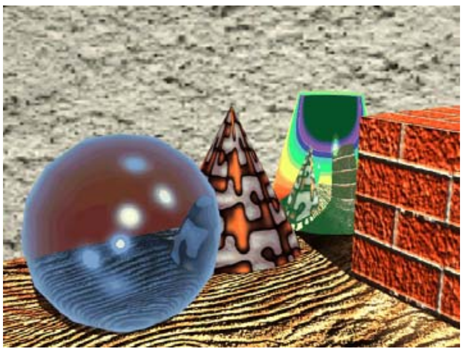
Beleuchtungsmodelle = physikalische Simulation der Lichtverhältnisse und –ausbreitung und deren Auswirkung auf das Erscheinen von Oberflächen.





Schatten = Bereiche, die keinen Blickkontakt mit Lichtquellen haben, werden dunkler dargestellt.

Reflexionen, Transparenz = reflektierende Objekte zeigen ein Spiegelbild, und durch transparente Objekte sieht man den Hintergrund.



Texturen = Muster auf Objektoberflächen werden „aufgemalt", um den Objekten ein komplexeres Aussehen zu geben (wirkt wesentlich echter).

Oberflächendetails = kleine geometrische Strukturen auf Oberflächen (z.B. Orangenschale, Rinde, Pflastersteine, Reifenprofil) werden mit Tricks simuliert.

Stereobilder = Für jedes Auge wird ein eigenes Bild hergestellt und diesem präsentiert (verschiedene Techniken) um einen 3D-Eindruck der Objekte zu erzeugen.

3D-Viewing-Pipeline

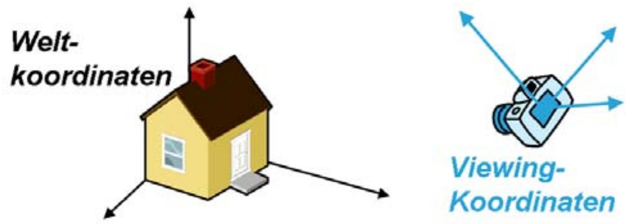
Die Viewing-Pipeline im 3-Dimensionalen ist mit der 2D-Viewing-Pipeline fast ident. Lediglich nach der Definition der Blickrichtung und Orientierung (also der Kamera) erfolgt noch ein zusätzlicher Projektions-Schritt, also die Reduktion der 3D-Daten auf eine Projektionsebene:

Dieser Projektionsschritt kann beliebig aufwändig sein, je nachdem welche der 3D-Viewing-Konzepte dabei einfließen sollen.

Viewing-Koordinaten

Ähnlich wie beim Fotografieren hat man beim Festlegen der Kamerawerte mehrere Freiheitsgrade:

1. Position der Kamera im Raum
2. Blickrichtung von dieser Position aus
3. Orientierung der Kamera (wo ist oben?)
4. Größe des Bildausschnittes (entspricht der Brennweite bei einem Fotoapparat)



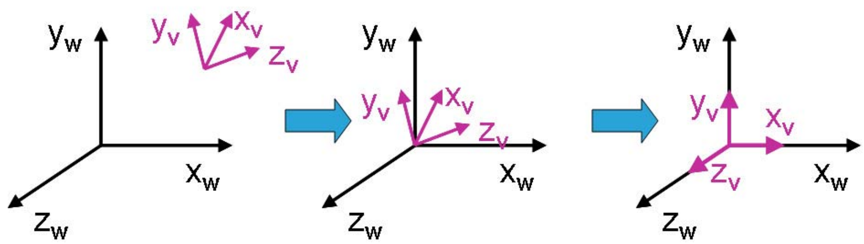
Mit diesen Parametern legt man das *Kamerakoordinatensystem* fest (Viewing-Koordinaten). Normalerweise ist die xy-Ebene dieses Viewing-Koordinatensystems normal auf die Hauptblickrichtung, und man *blickt in die Richtung der negativen z-Achse*.

Ausgehend von der Kameraposition geht man prinzipiell folgendermaßen vor, um das Viewing-Koordinatensystem festzulegen:

1. Wahl einer Kameraposition (auch Augpunkt oder Viewing-Punkt genannt)
2. Wahl einer Blickrichtung = Wahl der z^- Richtung des Viewing-Koordinaten.
3. Wahl einer Richtung „nach oben"; aus dieser lassen sich dann die x- und y-Achsen berechnen: Die Abbildungsebene liegt normal auf die Blickrichtung. Durch Parallelprojektion des Vektors „nach oben" auf diese Abbildungsebene erhält man die y-Achse der Viewing-Koordinaten.
4. Berechnung der x-Achse als Vektorprodukt der z- und y-Achsen.
5. Der Abstand der Abbildungsebene vom Augpunkt erzeugt den Blickwinkel, also die Größe des Ausschnittes der Szene, der abgebildet wird.

In Animationen wird die Kameradefinition meist aus bestimmten Bedingungen automatisch berechnet, z.B. wenn die Kamera rund um ein Objekt fährt oder bei einer Flugsimulation, so dass gewünschte Effekte unkompliziert erreicht werden.

Um die Weltkoordinaten in Viewingkoordinaten umzurechnen braucht man eine Kette von einfachen Transformationen: im Wesentlichen eine Translation der Koordinatenursprünge aufeinander und anschließend 3 Rotationen, so dass die Koordinatenachsen ebenfalls übereinander liegen (zwei Drehungen für die erste Achse, eine für die zweite Achse, und die dritte ist dann automatisch korrekt). Diese Transformationen kann man natürlich wieder durch Multiplikation zu einer Matrix zusammenfassen, das sieht etwa so aus:



$$M_{WC,VC} = R_z \cdot R_y \cdot R_x \cdot T$$

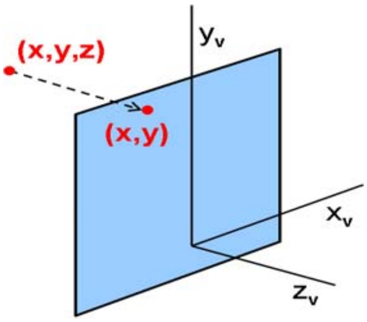
Projektionen

In der Geometrie sind mehrere verschiedene Projektionen bekannt, von denen in der Computergraphik aber nur die Parallelprojektion und die Perspektive eine größere Bedeutung haben.

Parallelprojektion

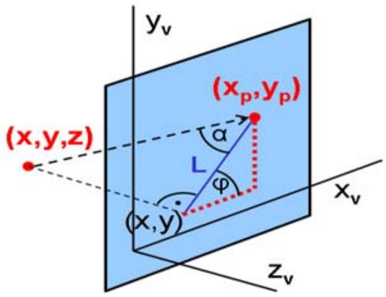
Eine Parallelprojektion kann entweder normal auf eine Abbildungsebene erfolgen, oder schräg (zum Beispiel auch beim Schattenwurf).

Die Parallelprojektion im rechten Winkel lässt sich unter der Annahme, dass die z-Achse die Projektsrichtung ist, sehr einfach bewerkstelligen, denn man braucht nur den z-Wert eines Punktes weglassen (bzw. auf 0 setzen). Die Matrix dazu ist daher auch äußerst simpel, weil einfach die z-Koordinate durch 0 ersetzt wird.



$$\mathbf{M}_{\text{parallel}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Eine schräge Parallelprojektion, die durch zwei Winkel α und φ definiert ist, erhält man so: der Abstand L zwischen dem im rechten Winkel auf die z-Ebene projizierten Punkt (x,y,(0)) und dem Ergebnis der schrägen Projektion $(x_p,y_p,(0))$ ist eine Kathete des rechtwinkeligen Dreiecks zwischen (x,y,z), (x,y,0) und $(x_p,y_p,0)$, aber auch die Hypotenuse des rechtwinkeligen Dreiecks in der Bildebene mit den Katheten $(x_p–x)$ und $(y_p–y)$. Daraus ergibt sich einerseits L = z/tan α, andererseits $x_p = x + L\cdot\cos\varphi$ und $y_p = y + L\cdot\sin\varphi$. Setzt man L in die weiteren Gleichungen ein so erhält man die angegebene Matrix.

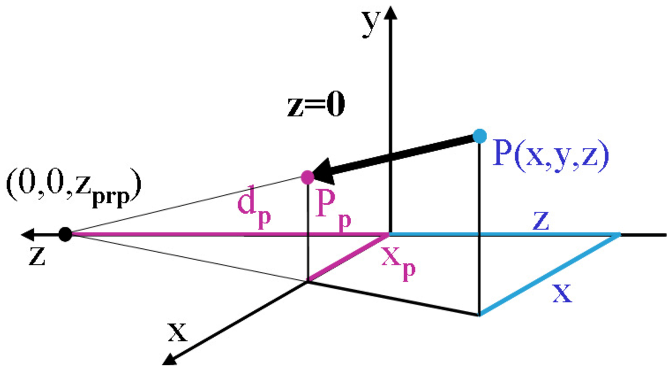


$$\mathbf{M}_{\text{parallel}} = \begin{bmatrix} 1 & 0 & \dfrac{\cos\varphi}{\tan\alpha} & 0 \\ 0 & 1 & \dfrac{\sin\varphi}{\tan\alpha} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Perspektivische Projektion

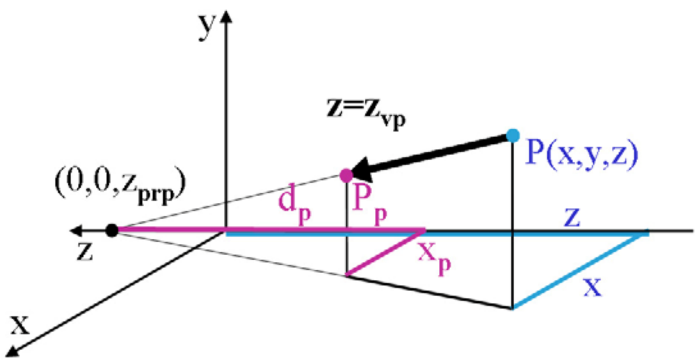
Bei der Perspektive gelten mehrere der Gesetze für affine Transformationen nicht mehr (z.B. verlaufen parallele Linien nach einer perspektiv. Projektion i.A. nicht mehr parallel), daher ist es keine affine Transformation und lässt sich auch nicht durch eine 3x3-Matrix erzeugen. Zum Glück helfen auch hier wieder die homogenen Koordinaten, allerdings ist dies der einzige Fall, bei dem die homogene Komponente h nicht den Wert 1 behält, und daher ein anschließender Divisionsschritt durch diesen Wert notwendig ist.

Betrachen wir zuerst den Fall, wo die Abbildung auf die <u>Ebene z=0</u> erfolgt. Ein Punkt P(x,y,z) wird auf den Punkt abgebildet, wo die Verbindung mit dem Projektionszentrum $(0,0,z_{prp})$ die Ebene z=0 schneidet. Von oben betrachtet erkennt man ähnliche rechtwinkelige Dreiecke mit den Katheten d_p und x_p einerseits, und $d_p–z$ (weil z auf der negativen z-Achse liegt) und x andererseits.



Daraus folgt $x_p : x = d_p : (d_p–z)$ oder $x_p = x\cdot d_p/(d_p–z)$
und analog $y_p : y = d_p : (d_p–z)$ oder $y_p = y\cdot d_p/(d_p–z)$
und selbstverständlich $z_p = 0$.

Wenn wir auf eine andere <u>Ebene $z = z_{vp} \neq 0$</u> abbilden, so ändert sich nicht viel: lediglich d_p wird einmal durch den z-Wert des Projektionszentrums z_{prp} ersetzt:



$x_p : x = d_p : (z_{prp} - z)$ oder $x_p = x \cdot d_p / (z_{prp} - z)$
$y_p : y = d_p : (z_{prp} - z)$ oder $y_p = y \cdot d_p / (z_{prp} - z)$
$z_p = z_{vp}$.

Dies lässt sich mit einer homogenen Matrix bewerkstelligen, die so aussieht:

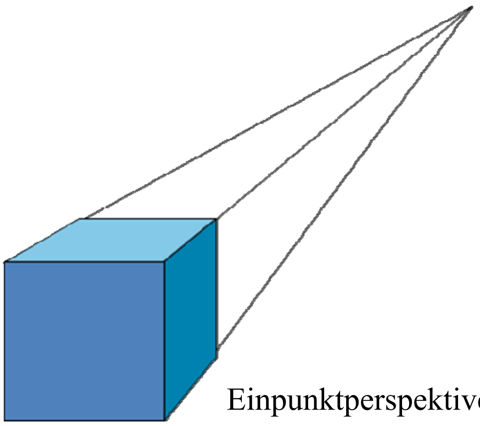
$$\begin{bmatrix} \mathbf{x_h} \\ \mathbf{y_h} \\ \mathbf{z_h} \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{z_{vp}/d_p} & \mathbf{z_{vp}(z_{prp}/d_p)} \\ \mathbf{0} & \mathbf{0} & -\mathbf{1/d_p} & \mathbf{z_{prp}/d_p} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \\ \mathbf{1} \end{bmatrix}$$

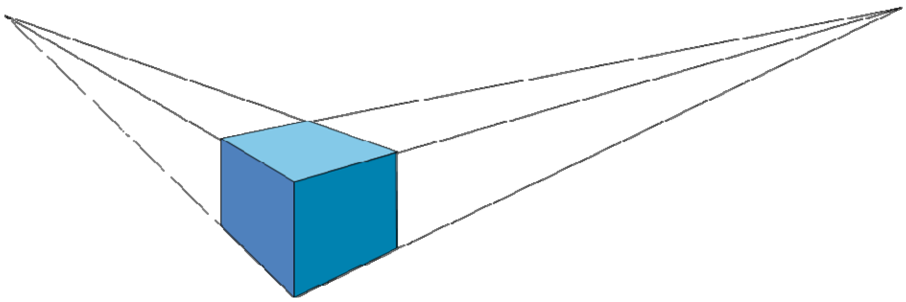
Im Allgemeinen wird der Wert h des Ergebnisvektors ungleich 1 sein (h berechnet sich zu $(z_{prp} - z)/d_p$), daher muss der Ergebnisvektor noch durch h dividiert werden (also mit 1/h skaliert werden) um den korrekten Ergebnispunkt zu erhalten:

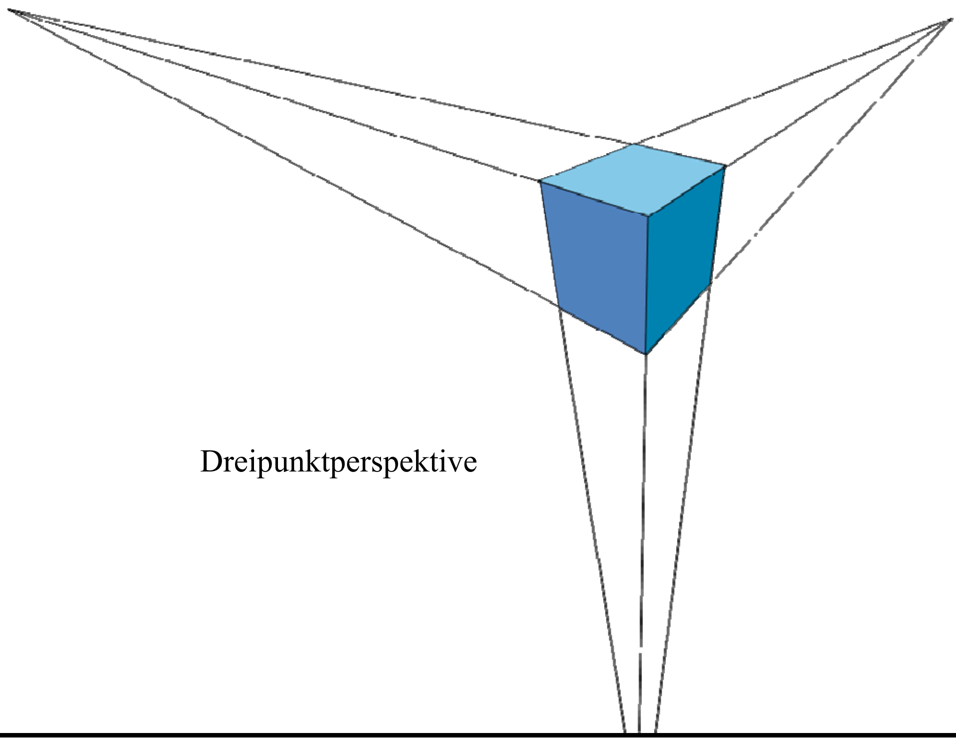
$x_p = x_h/h$, $y_p = y_h/h$ und für z_p erhält man z_{vp} .

Durch die Repräsentation der Projektionen in Matrixform ist es nun möglich, die gesamte Umwandlung von Modellkoordinaten bis zu Gerätekoordinaten durch Multiplikation der beteiligten Matrizen in einer einzigen Matrix zu formulieren. Wenn eine perspektivische Abbildung beteiligt ist darf man nicht vergessen, das Ergebnis durch die homogene Komponente h zu dividieren.

Abschließend sei noch erwähnt, dass die Anzahl der *Hauptfluchtpunkte* von der Lage der Bildebene zum Koordinatensystem abhängt. Wenn zwei Achsen parallel zur Bildebene liegen, dann spricht man von *Einpunktperspektive*, ist nur eine Achse parallel zur Bildebene, dann spricht man von *Zweipunktperspektive*, und sind alle drei Achsen nicht parallel zur Bildebene, dann spricht man von *Dreipunktperspektive* (weil es dann 3 Hauptfluchtpunkte gibt).


Einpunktperspektive


Zweipunktperspektive


Dreipunktperspektive