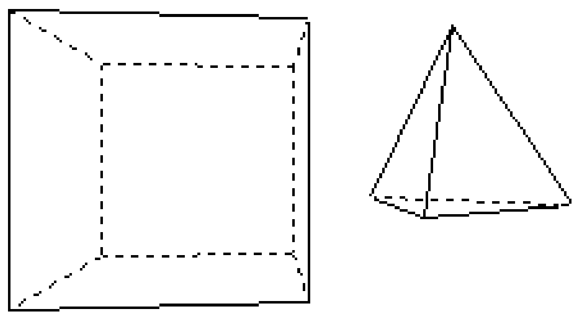
Kapitel 16

3D-Objekte

Im Flächenmodell werden Objekte durch ihre Begrenzungsflächen beschrieben und diese wiederum durch ihre Kanten mit den Eckpunkten im Modellkoordinatensystem. Außerdem wird noch ein Normalenvektor im MC benötigt. Auf diese Weise kann ein Programm Polyeder exakt darstellen und gekrümmte Oberflächen durch konvexe Polygone beliebig gut approximieren.

16.1 Polyeder

Am Beispiel des Würfels wird die Repräsentation von Polyedern im Halbkantenmodell erläutert.



Vom Projektionsalgorithmus erzeugte Szene mit Würfel und Tetraeder in der Drahtmodell-Darstellung mit gestrichelten Rückkanten

16.1.1 Würfel

Der *Würfel* mit der Kantenlänge 1 hat im MC die homogenen Eckpunktkoordinaten

$$\begin{array}{llllllll}
(+0.5, & +0.5, & +0.5, & 1), & (+0.5, & +0.5, & -0.5, & 1),\\
(+0.5, & -0.5, & +0.5, & 1), & (+0.5, & -0.5, & -0.5, & 1),\\
(-0.5, & +0.5, & +0.5, & 1), & (-0.5, & +0.5, & -0.5, & 1),\\
(-0.5, & -0.5, & +0.5, & 1), & (-0.5, & -0.5, & -0.5, & 1).
\end{array}$$

Da jede der sechs Flächen des Würfels eben ist, gibt es pro Fläche nur einen Normalenvektor, der durch das Kreuzprodukt zweier benachbarter Kanten berechnet werden kann. Die Reihenfolge der Punkte ist dabei signifikant, da jeweils aufeinanderfolgende Punkte eine Kante bilden. Ein *Quader* entsteht aus dem Würfel durch ungleichmäßige Skalierung beim Modeling.

Auch alle anderen Polyeder werden im MC in der Halbkantendarstellung definiert, ein Tetraeder beispielsweise mit Kantenlänge 1 und Schwerpunkt im Ursprung.

16.2 Gekrümmte Flächen

Eine gekrümmte Oberfläche, die analytisch beschrieben werden kann, läßt sich beliebig gut durch Polygone approximieren. Die Gruppe der Objekte mit analytisch gekrümmter Oberfläche besteht aus zwei Typen: Während z.B. die Kugel nur aus einer solchen Fläche besteht, sind Zylinder und Kegel aus mehreren Flächen zusammengesetzt.

Die Repräsentation der gekrümmten Körper erfolgt im Flächenmodell mit einer Flächen- und Eckpunkteliste sowie einer Normalenliste, die pro Eckpunkt die zugehörige Normale enthält.

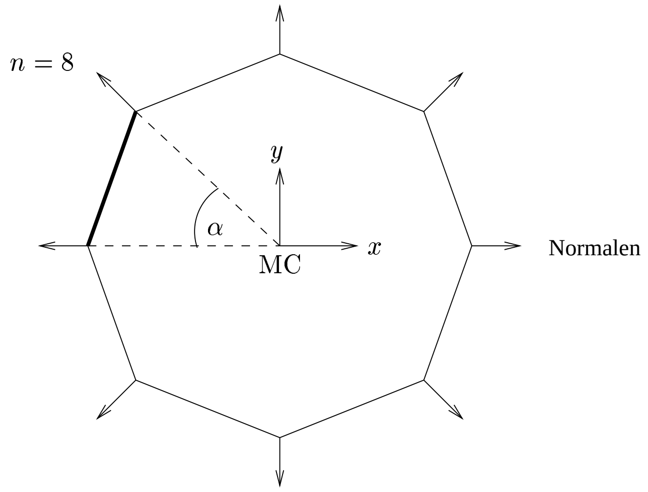
16.2.1 Zylinder

Der zur z-Achse symmetrische *Zylinder* mit Höhe 2 und Radius 1 besteht aus einer Mantelfläche und zwei Deckflächen. Die beiden Kreisscheiben bei $z = 1$ und $z = -1$ lassen sich durch regelmäßige n-Ecke darstellen. Die Mantelfläche wird durch n Rechtecke approximiert. Dabei wachsen mit n sowohl die Genauigkeit der Approximation als auch der Rechenaufwand. Sei $\alpha = (2\pi)/n$, so lauten für ein solches Rechteck die vier Eckpunkte und die zugehörigen Normalenvektoren:

Eckpunkt	Normalenvektor
$(\cos(\phi), \sin(\phi), +1, 1)$	$(\cos(\phi, \sin(\phi), 0, 0)$
$(\cos(\phi+\alpha), \sin(\phi+\alpha), +1, 1)$	$(\cos(\phi+\alpha), \sin(\phi+\alpha), 0, 0)$
$(\cos(\phi+\alpha), \sin(\phi+\alpha), -1, 1)$	$(\cos(\phi+\alpha), \sin(\phi+\alpha), 0, 0)$
$(\cos(\phi), \sin(\phi), -1, 1)$	$(\cos(\phi), \sin(\phi), 0, 0)$

mit $\phi = k \cdot \alpha$, $k \in \{0, ..., n-1\}$.

Als Normalenvektor in einem Eckpunkt dieser Fläche wird also der tatsächliche Normalenvektor der Mantelfläche genommen. Dadurch erhalten aneinandergrenzende Flächen in ihren gemeinsamen Eckpunkten auch denselben Normalenvektor. Auf diese Weise erzeugt das Programm bei der späteren Beleuchtung den Eindruck eines stetigen Übergangs zwischen den Flächen. Der Betrachter nimmt statt einer n-eckigen Säule den Zylinder wahr.



Zylinder im Grundriß mit Normalenvektoren der Mantelfläche

16.2.2 Kugel

Die Oberfläche einer *Kugel* mit Radius 1 kann beschrieben werden durch

$$(\sin(\theta)\cos(\phi), \sin(\theta)\sin(\phi), \cos(\theta)), 0 \le \phi < 2\pi, 0 \le \theta < \pi \ .$$

Zur Approximation durch Flächen wird der Vollwinkel in n Teile zerlegt:

$$\alpha = (2\pi)/n, \quad n \in \mathbb{N} \text{ gerade } .$$

Dadurch entstehen auf der Kugel n Längenkreise und $(n/2)$ Breitenkreise. Diese schneiden n Dreiecke an jedem Pol und $n(n/2-2)$ Vierecke aus der Kugeloberfläche. Die Ortsvektoren eines Dreiecks am Nordpol ($\theta = 0$) lauten

$$(0, 0, +1, 1),$$

$$(\sin(\alpha)\cos(\phi), \sin(\alpha)\sin(\phi), \cos(\alpha), 1),$$

$$(\sin(\alpha)\cos(\phi+\alpha), \sin(\alpha)\sin(\phi+\alpha), \cos(\alpha), 1),$$

mit $\phi = k \cdot \alpha$, $k \in \{0, ..., n-1\}$.

Eines der Vierecke hat die Ortsvektoren

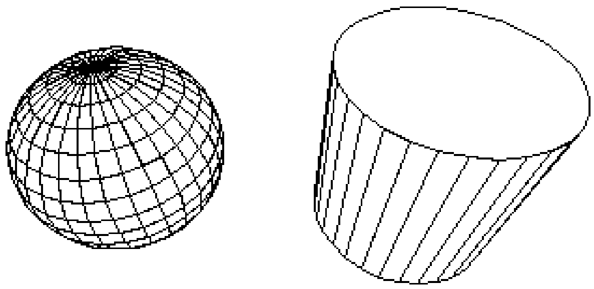
$$(\sin(\theta)\cos(\phi), \sin(\theta)\sin(\phi), \cos(\theta), 1),$$

$$(\sin(\theta)\cos(\phi+\alpha), \sin(\theta)\sin(\phi+\alpha), \cos(\theta), 1),$$

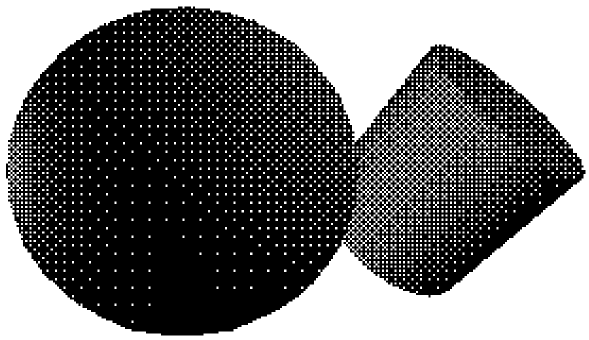
$$(\sin(\theta+\alpha)\cos(\phi+\alpha)\sin(\theta+\alpha)\sin(\phi+\alpha), \cos(\theta+\alpha), 1),$$

$$(\sin(\theta+\alpha)\cos(\phi), \sin(\theta+\alpha)\sin(\phi), \cos(\theta+\alpha), 1),$$

mit $\phi = k \cdot \alpha, k \in \mathbb{N}, k < n$ und $\theta = l \cdot \alpha, l \in \mathbb{N}, 0 < l < (n/2-1)$.
Als Normalenvektor wird in jedem Eckpunkt der Ortsvektor als Richtungsvektor ($w = 0$) eingetragen, denn der Radiusvektor steht senkrecht auf der Kugeloberfläche. Einen *Ellipsoid* erzeugt der Rendering-Algorithmus aus der Kugel durch ungleichmäßige Skalierung beim Modeling.



Vom Projektionsalgorithmus erzeugte Szene mit Kugel und Zylinder in der Drahtmodell-Darstellung ohne Rückkanten ($n = 32$).



Vom Renderingalgorithmus erzeugte Szene mit Kugel und Zylinder von zwei Strahlern beleuchtet, Gouraud-schattiert.