

3D-Grafik Programmierung

von
Marius Apetri

Neuausgabe

[3D-Grafik Programmierung – Apetri](#)

schnell und portofrei erhältlich bei beck-shop.de DIE FACHBUCHHANDLUNG

mitp/bhv 2007

Verlag C.H. Beck im Internet:
www.beck.de
ISBN 978 3 8266 1767 6

Aus dem Inhalt:

- Mathematische Grundlagen
- Grafikprogrammierung in C++
- 3D-Programmierung
- 3D-Polygonfiguren
- Verwendung von Matrizen
- Projektion und Rasterization
- Schattieralgorithmen
- Texture Mapping
- Semitransparenzeffekte
- Echtzeitdarstellung dreidimensionaler Landschaften

Neu in der 2. Auflage:

- Linsen- und Plasmaeffekte
- Fraktale Grafiken
- Texturparkettierung
- Polygonübergreifende Texturprojektion
- Einführung in OpenGL und Direct3D

Auf der CD:

- CYGWIN Compiler (32 Bit)
- sämtliche Quellcodes
- ca. 150 zusätzliche pdf-Seiten mit den Lösungen und Besprechungen der Übungsaufgaben und Projekte sowie 2 zusätzliche Anhänge

Leserstimmen zur Vorausgabe:

„Es ist eines der besten deutschsprachigen Bücher über 3D-Programmierung.“

„Wer sich ernsthaft mit Computergrafik beschäftigen will, kommt an diesem Buch nicht vorbei.“

Wertung bei amazon.de: ★★★★★

Probekapitel und Infos erhalten Sie unter:
info@mitp.de
www.mitp.de

Regalsystematik:
Programmierung

(D) € 49,95

ISBN 978-3-8266-1767-6

 9 783826 617676

Komplexe dreidimensionale Grafikentwicklung ist die Grundlage vieler neuer Softwareprodukte. Spiele, Animationen für Film- und Videoprojekte, wissenschaftliche Darstellungen oder „virtuelle Realität“ in der Technik basieren auf den mathematischen und softwaretechnischen Methoden der 3D-Grafikprogrammierung.

Dieses Buch liefert die ideale Grundlage für den ambitionierten Software-Entwickler in der Grafikprogrammierung, es deckt alle wesentlichen Aspekte des Fachgebietes ab und kann als Lese- und Nachschlagewerk verwendet werden. Alle Themen werden ausführlich und allgemein verständlich behandelt. Die zweite Auflage des Buches wurde neben den vollständig überarbeiteten Texten um zahlreiche neue Themen erweitert wie beispielsweise die Visualisierung einer virtuellen Welt aus der Sicht verschiedener Kameras. Die Erarbeitung der Kenntnisse wird durch zahlreiche Übungsaufgaben und Beispielprogramme unterstützt.

Zusätzlich zur Beschreibung der Themen werden die Techniken und Vorgehensweisen vermittelt, die dem Leser die Formulierung eigener, unabhängiger Algorithmen ermöglichen.

Dank des bewährten didaktischen und thematischen Aufbaus werden komplexe Zusammenhänge einem breiten Publikum nachvollziehbar erklärt. Dieses Buch ist selbst für Nicht-Mathematiker und Leser mit nur wenigen Grundkenntnissen verständlich.

Zum gleichen Thema:

ISBN 978-3-8266-1469-0



ISBN 978-3-8266-5512-8



ISBN 978-3-8266-5945-4

3D-Grafik Programmierung



inklusive



Marius Apetri

3D-Grafik Programmierung

Alle mathematischen Grundlagen

Von einfachen Rasteralgorithmen bis hin zu Landscape Generation

3D-Grafik in C++, optimaler Einstieg in OpenGL und Direct3D

Marius Apetri

1767

48,99 mm



Inhaltsverzeichnis

Vorwort	17
I Mathematische Grundlagen	25
I.1 Zuordnungen	25
I.1.1 Proportionale Zuordnungen	26
I.1.2 Antiproportionale Zuordnungen	28
I.1.3 Beliebige lineare Zuordnungen	30
I.2 Der Mengenbegriff nach Cantor	33
I.3 Punkte in der Ebene und im Raum	36
I.4 Vektoren	40
I.4.1 Die Länge eines Vektors	42
I.4.2 Operationen mit Vektoren	44
I.5 Trigonometrische Funktionen	48
I.5.1 Sinus- und Kosinuswerte am beliebigen rechtwinkligen Dreieck	50
I.5.2 Projekt: Die Koordinaten der Punkte regelmäßiger Polygone	52
I.5.3 Rotation im dreidimensionalen Raum	53
I.5.4 Die Einteilung in Bogenmaß	56
I.5.5 Der Winkel zwischen zwei Vektoren	58
I.5.6 Inverse trigonometrische Funktionen	61
I.6 Geraden	62
I.7 Ebenen	63
I.7.1 Das Skalarprodukt	66
I.7.2 Die Normalenform der Ebene	69
I.7.3 Der dreidimensionale Raum als Menge von Vektoren	70
I.7.4 Die Parameterform der Ebene	73
I.8 Projekt: Die Abstände von Punkten, Geraden und Ebenen	75
2 Einführung in die Grafikprogrammierung	77
2.1 Grundarchitektur der Grafikkarte	78
2.1.1 Auflösung und Farbtiefe	78
2.1.2 Der Videospeicher	79



Inhaltsverzeichnis

2.1.3	Die Position eines Pixels	80
2.1.4	Praktische Beschriftung des Videospeichers	82
2.1.5	Zufallszahlen	87
2.1.6	Passive Wiederholungsanweisungen	89
2.2	Der Funktionsbegriff	91
2.3	Darstellung von Linien	95
2.3.1	Bresenhams Algorithmus zur Darstellung beliebig verlaufender Linien	98
2.4	Projektion von R^2 nach R^2	107
2.4.1	Projekt a2_7: Praktische Darstellung von Funktionen von R nach R	111
2.5	Funktionen von R nach R^2	114
2.5.1	Direkte und rekursive Algorithmen	117
2.5.2	Bresenhams Algorithmus zur Darstellung von Kreisen	118
2.6	Hardwarebeschleunigte Darstellung von Punkten	123
2.6.1	Projekt a2_12: Darstellung des Schneeffekts in einem Rahmen	127
2.7	Hardwarebeschleunigte Darstellung von Linien	129
2.7.1	Praktische Darstellung regelmäßiger n-Ecke	130
2.8	Projekt a2_14: Darstellung von Zykloiden	134
2.9	Darstellung von Kurven	139
2.9.1	Bézierkurven	139
2.9.2	Splines	144
3	Einführung in die 3D-Programmierung	169
3.1	Der Polygonbegriff	169
3.1.1	Praktischer Umgang mit Polygone	174
3.1.2	Verwaltung von Polygone mit zweidimensionalen Punkten	176
3.1.3	Praktische Darstellung von Polygone mit zweidimensionalen Punkten	179
3.2	Die Projektion von R^3 nach R^2	180
3.2.1	Erster Schritt der Projektion	182
3.2.2	Zweiter Schritt der Projektion	183
3.2.3	Der Viewport	185
3.2.4	Verwaltung dreidimensionaler Polygone	187
3.2.5	Praktische Darstellung von Polygone mit dreidimensionalen Koordinaten	191

Inhaltsverzeichnis

3.2.6	Die Projektionskonstante	193
3.3	Lineare Funktionen und Matrizen	198
3.3.1	Lineare Funktionen von R^n nach R	199
3.3.2	Aufbau von Matrizen	201
3.3.3	Elementare Bewegungen im dreidimensionalen Raum	205
3.3.4	Multiplikation von Matrizen	212
3.3.5	Verwaltung von Matrizen	220
3.3.6	Praktischer Einsatz von Matrizen	226
3.3.7	Fehlerbehandlung	229
3.4	Double Buffering	230
3.4.1	Das Problem der direkten Beschriftung des Videospeichers	234
3.4.2	Flimmerfreie Animation	237
3.5	Die Rotation	241
3.5.1	Die räumliche Position eines komplexen Gegenstands	241
3.5.2	Rotation um die Koordinatenachsen der dreidimensionalen Welt	243
3.5.3	Die eigenen Achsen eines Gegenstands	247
3.5.4	Rotation eines Gegenstands um seine eigenen Achsen	248
3.5.5	Rotation um einen beliebigen Punkt	254
3.6	Der Einsatz variabler Matrizen	258
3.6.1	Aufbau variabler Matrizen	263
3.6.2	Praktischer Einsatz variabler Matrizen	264
3.6.3	Vor- und Nachteile beider Typen von Matrizen	267
3.6.4	Projekt a3_i3: Gleichzeitige Rotation um die eigenen Achsen und um einen beliebigen Punkt	269
3.7	Skalierung	271
3.7.1	Die Versetzung in Richtung des Ursprungs	275
3.7.2	Praktische Unterbindung der skalierungsbedingten Translation	276
3.7.3	Ungleichmäßige Skalierung	277
3.7.4	Iteration Counting	278
3.7.5	Praktische Umsetzung der ungleichmäßigen Skalierung	279
3.7.6	Fehlerbehandlung	282
3.7.7	Die Illusion der Tiefenverschiebung	284
3.7.8	Praktische Erzeugung der Illusion der Tiefenverschiebung	285

Inhaltsverzeichnis

3.8	Hardwarebeschleunigte Darstellung von Linien mit OpenGL	288
3.8.1	Der Viewport in OpenGL	288
3.8.2	Praktische Darstellung von Drahtgittermodellen	290
3.9	Hardwarebeschleunigte Darstellung von Linien mit DirectX	296
3.9.1	Initialisierung der dreidimensionalen Welt	296
3.9.2	Praktische Darstellung von Linien	297
4	Polyeder aus gefüllten Polygonen	305
4.1	Der Begriff des Polygons	306
4.2	Rasterization	311
4.2.1	Die Kategorie einer Polygonseite	312
4.2.2	Anfang und Ende der Rasterzeilen	315
4.2.3	Die lineare Interpolation	316
4.2.4	Praktische Ermittlung der Endpunkte der Rasterzeilen	319
4.2.5	Bestimmung der Kategorie einer Polygonseite	320
4.2.6	Der Rasterisationsalgorithmus	322
4.2.7	Ausnahmebehandlung	326
4.2.8	Darstellung konkaver Polygone	327
4.2.9	Praktische Darstellung gefüllter Polygone	330
4.2.10	Projekt: Offsetinterpolation	334
4.2.11	Das Problem der Nummerierung entgegen dem Uhrzeigersinn	335
4.2.12	Darstellung entgegen dem Uhrzeigersinn nummerierter Polygone	337
4.3	Polyeder	341
4.3.1	Der Begriff des Polyeders	341
4.3.2	Benutzerdefinierter Aufbau von Polygonen	344
4.3.3	Der Mittelpunkt eines Polyeders	345
4.3.4	Speicherung von Polyedern	345
4.3.5	Verwaltung von Polyedern	347
4.3.6	Der Einsatz von Polygonen als Bestandteil von Polyedern ..	353
4.3.7	Praktische Darstellung von Polyedern	356
4.4	Visual Surface Determination	358
4.4.1	Die Vorder- und Rückseite von Polygonen	359
4.4.2	Automatische Erkennung nicht sichtbarer Polygone	362
4.4.3	Projekt: Platonische Körper	363
4.4.4	Hardwarebeschleunigte Erkennung abgewandelter Polygone	365

Inhaltsverzeichnis

4.5	Der Z-Buffer Algorithmus	367
4.5.1	Grundidee des Z-Buffer-Algorithmus	367
4.5.2	Speicherdarstellung der dreidimensionalen z-Koordinaten	368
4.5.3	Funktionsweise des Z-Buffer-Algorithmus	370
4.5.4	Mathematische Grundlage des linearen Z-Buffer-Algorithmus	373
4.5.5	Der Einsatz des Z-Buffers während der Rasterization	378
4.5.6	Bestimmung der z-Koordinaten entlang der Rasterzeilen	381
4.5.7	Das Löschen des Z-Buffers	385
4.5.8	Praktische Implementierung des Z-Buffer-Algorithmus	386
4.5.9	Z-Buffer vs. Visual-Surface-Determination-Algorithmus ...	387
4.6	Der Clear-Reduction-Algorithmus	389
4.6.1	Grundidee des Clear-Reduction-Algorithmus	389
4.6.2	Funktionsweise des Clear-Reduction-Algorithmus	390
4.6.3	Einordnung des Clear-Reduction-Algorithmus innerhalb des Visualisierungsprozesses	392
4.6.4	Clear-Reduction-Algorithmus und das Zurücksetzen des Z-Buffers	395
4.6.5	Praktischer Einsatz des Clear-Reduction-Algorithmus	397
4.7	Der Z-Buffer-Algorithmus in hardwarebeschleunigten Programmen	400
4.8	Polygon-Clipping	403
4.8.1	Grundlagen des Polygon-Clippings	404
4.8.2	Mathematische Grundlagen des Pre-Projection-Clippings	405
4.8.3	Mathematische Grundlagen des Post-Projection-Clippings	409
4.8.4	Der Sutherland-and-Hodgman-Polygon-Clipping-Algorithmus	411
4.8.5	Praktische Darstellung beliebiger, mathematisch definierter Polygone	426
4.8.6	Projekt: Drehung um einen beliebigen Punkt mit benutzerdefinierter Rotationsebene	428
5	Unterstützung von Eingabegeräten	431
5.1	Verwaltung der Tastatur	431

Inhaltsverzeichnis

5.1.1	Steuerung von Anwendungen in Multitasking-Betriebssystemen	431
5.1.2	Indirekte Verarbeitung von Benutzereingaben	435
5.1.3	Vor- und Nachteile der Verwendung eventbasierender Benutzereingaben	438
5.1.4	Direkte Verarbeitung von Benutzereingaben	440
5.1.5	Das Problem der flachen Objektkopien	445
5.2	Verwaltung der Maus	448
5.2.1	Die Position des Mauszeigers	449
5.2.2	Abfrage der Cursorposition	450
5.2.3	Abfrage des Zustands der Maustasten	452
5.2.4	Praktische Verwaltung der Maus	453
5.2.5	Praktische Verarbeitung von Mauseingaben	456
6	Einfache Polygonschattierung	463
6.1	Mathematische Ansätze zur Simulation natürlicher Farbwahrnehmung	465
6.1.1	Das RGB-Farbmodell	465
6.2	Flat Shading	472
6.2.1	Intensität der Polygonfarbe	473
6.2.2	Mathematische Grundlagen des Flat-Shading-Algorithmus	475
6.2.3	Das Umgebungslicht	479
6.2.4	Implementierung des Flat-Shading-Algorithmus	481
6.3	Gouraud Shading	495
6.3.1	Der Normalenvektor eines Punktes	497
6.3.2	Farbinterpolation	501
6.3.3	Rotation der Vektornormalen dreidimensionaler Vertices	509
6.3.4	Praktischer Einsatz des Gouraud-Shading-Algorithmus	510
6.4	Unterstützung mehrerer Primärfarben	511
6.4.1	Möglichkeiten des Einsatzes mehrerer Farbpaletten	519
6.5	Hardwarebeschleunigte Polygonschattierung	520
6.6	Projekt: Rotationssymmetrische Polyeder	527
6.7	Einheitssphären verschiedener Längenfunktionen	531
6.8	Polygon Morphing	532
6.9	Phong Shading	537

Inhaltsverzeichnis

6.9.1	Die Vektornormale eines Pixels	539
6.9.2	Anpassung der Vektornormalen im Laufe des Polygon Clipping	544
6.9.3	Praktischer Einsatz des Phong-Shading-Algorithmus	546
6.10	Metal Shading	547
6.10.1	Exponentialinterpolation	548
6.10.2	Praktischer Einsatz des Metal-Shading-Algorithmus	555
6.10.3	Implementierung in die Hardware	556
6.10.4	Metal Shading unter Verwendung einer alternativen Farbgebung	560
6.11	Farbverläufe n-ten Grades	561
6.11.1	Anwendungsgebiete für Farbverläufe höheren Grades	562
6.11.2	Praktischer Einsatz von Farbverläufen n-ten Grades	563
7	Bitmaps	567
7.1	Verwaltung von Bitmaps	568
7.1.1	Der Umgang mit 8-Bit-Bitmaps	569
7.2	Das 8-Bit-BMP-Dateiformat	572
7.2.1	Praktische Darstellung von 8-Bit-Bitmaps	574
7.3	Beschriftung von Bitmaps	576
7.3.1	Grundlagen der Beschriftung von Bitmaps	576
7.3.2	Praktische Darstellung von Linien innerhalb von Bitmaps	579
7.3.3	Background Buffering	582
7.3.4	Projekt: Elementare Bitmap-Rotation	584
7.4	Verwaltung von 32-Bit-Bitmaps	585
7.4.1	Die Struktur von 24-Bit-BMP-Dateien	586
7.5	Partielle Darstellung von Bitmaps	589
7.5.1	Visualisierung rechteckiger Teilbereiche von Bitmaps	590
7.5.2	Einfache Transparenzeffekte	596
7.5.3	Zweidimensionale Umgebungen	598
7.5.4	Bitmap Clipping	601
7.5.5	Verwaltung zweidimensionaler Gegenstände	610
7.5.6	Praktische Visualisierung zweidimensionaler Gegenstände	613
7.6	Semitransparenzeffekte	622
7.6.1	Grundidee der Generierung semitransparenter Darstellungen	622

Inhaltsverzeichnis

7.6.2	Mathematische Grundlagen der Generierung semitransparenter Darstellungen	624
7.6.3	Praktische Generierung semitransparenter Darstellungen	625
7.7	Feuereffekte	630
7.7.1	Praktische Darstellung von Feuereffekten	632
7.8	Plasmaeffekte	636
7.8.1	Praktische Erzeugung von Plasmaeffekten	637
7.8.2	Konstruktion von Plasmafunktionen	638
7.9	Plasmafraktale	642
7.9.1	Grundaufbau	642
7.9.2	Praktischer Aufbau von Plasmafraktalen	644
7.9.3	Praktische Ausgabe von Plasmafraktalen	650
7.9.4	Der Einsatz ringförmiger Farbpaletten	652
7.10	Bitmap-Verzerrung	654
7.10.1	Modellierung einfacher Linsen	657
7.10.2	Einsatz komplexer Linsenfunktionen	660
7.10.3	Erzeugung von Wirbeln	665
7.11	Die Mandelbrotmenge	669
7.11.1	Farbgebung komplexer Zahlen	670
7.11.2	Praktische Darstellung der Mandelbrotmenge	672
7.11.3	Projekt a7_11: Benutzerdefinierte Vergrößerung von Bereichen der Mandelbrotmenge	675
8	Texturprojektion	679
8.1	Linear Texture Mapping	680
8.1.1	Mathematische Grundlage des Linear-Texture-Mapping-Algorithmus	683
8.1.2	Implementierung des Linear-Texture-Mapping-Algorithmus	686
8.1.3	Praktischer Einsatz des Texture-Mapping-Algorithmus	700
8.1.4	Fehlerbehebung nach dem High-Level-, Low-Level-Prinzip	701
8.2	Perspective Texture Mapping	704
8.2.1	Grundlagen des Perspective-Texture-Mapping-Algorithmus	706
8.2.2	Interpolation der inversen z-Koordinaten	707

Inhaltsverzeichnis

8.2.3	Praktischer Einsatz des Perspective-Texture-Mapping-Algorithmus	710
8.3	Der perspektivische Z-Buffer-Algorithmus	711
8.3.1	Simple Depth Shading	711
8.3.2	Praktischer Einsatz des Depth-Shading-Algorithmus	713
8.3.3	Z-Buffer-Algorithmus unter Verwendung inverser z-Koordinaten	715
8.3.4	Mathematische Grundlagen des perspektivischen Z-Buffer-Algorithmus	724
8.4	Multitextureffekte	729
8.4.1	Statische Multitextureffekte	730
8.4.2	Dynamische Multitextureffekte	734
8.4.3	MIP Mapping	737
8.4.4	Der Einsatz verallgemeinerter Texturkoordinaten	740
8.4.5	Implementierung des Pixel-Based-MIP-Mapping-Algorithmus	743
	8.4.6 Effiziente Verwaltung von Texturen	747
8.5	Hardwarebeschleunigte Texturprojektion	753
8.6	Projekt: Texturprojektion im zweidimensionalen Raum	760
8.7	Texturparkettierung	783
8.7.1	Optimierte Texturparkettierung	789
8.8	Projekt: Texturierung rotationssymmetrischer Polyeder	793
8.9	Erweiterte Polygonschattierung	795
8.9.1	Textured Flat Shading	796
8.9.2	Projekt: Darstellung einer wehenden Fahne	808
8.10	Navigation im dreidimensionalen Raum	811
8.10.1	Praktische Simulation der Bewegung innerhalb einer dreidimensionalen Welt	812
8.10.2	Lokale Koordinatensysteme	813
8.10.3	Der Einsatz lokaler Darstellungskameras	833
8.11	Benutzerdefinierte Sichtfenster	836
8.11.1	Praktischer Einsatz benutzerdefinierter Sichtfenster	837
8.11.2	Die Projektionskonstante eines benutzerdefinierten Sichtfensters	844
8.11.3	Lokale Darstellungskameras und benutzerdefinierte Sichtfenster	846
8.11.4	Rotation um beliebig verlaufende Geraden	852
8.11.5	Der Einsatz hardwarebeschleunigter Matrizen	857

Inhaltsverzeichnis

9 Aufbau und effiziente Darstellung dreidimensionaler Landschaften	865
9.1 Projekt: Generierung einer einfachen Landschaft	866
9.2 Der Algorithmus Visible Terrain Visualization	873
9.2.1 VTV-Algorithmus, einfache Version	874
9.2.2 VTV-Algorithmus, optimierte Version	883
9.2.3 Weiterführende Möglichkeiten des VTV-Algorithmus	897
9.3 Uneingeschränkte Bewegung innerhalb dreidimensionaler Umgebungen	898
9.3.1 Erweiterung des Höhenfeldes	902
9.3.2 Texturierung des erweiterten Höhenfeldes	907
9.4 Der Einsatz unterschiedlicher Landschaftstexturen	910
9.4.1 Bitmapverläufe n-ten Grades	911
9.4.2 Aufbau einer erweiterten Landschaftstextur	914
9.4.3 Der Einsatz von Detailtexturen	918
9.5 Berechnung von Schatten	919
A Erweiterte mathematische Grundlagen	925
A.1 Das Zahlengebäude	925
A.1.1 Die Menge der natürlichen Zahlen	926
A.1.2 Die Menge der ganzen Zahlen	928
A.1.3 Die Menge der rationalen Zahlen	929
A.1.4 Die Menge der reellen Zahlen	932
A.1.5 Die Menge der komplexen Zahlen	934
A.1.6 Weiterentwicklung des Zahlengebäudes	938
A.2 Folgen reeller Zahlen	939
A.3 Die erweiterte binomische Formel	941
B Literaturempfehlungen	945
B.1 Weiterführende Literatur, Computergrafik	945
B.2 Weiterführende Literatur, Mathematik	946
Stichwortverzeichnis	947