Teil 4: Texturing

Farbe, Struktur, Umgebung

Wozu?

Mit Textur:

- Farbe
- Oberflächenstruktur
- Reflexion, Transparenz
- Highlights

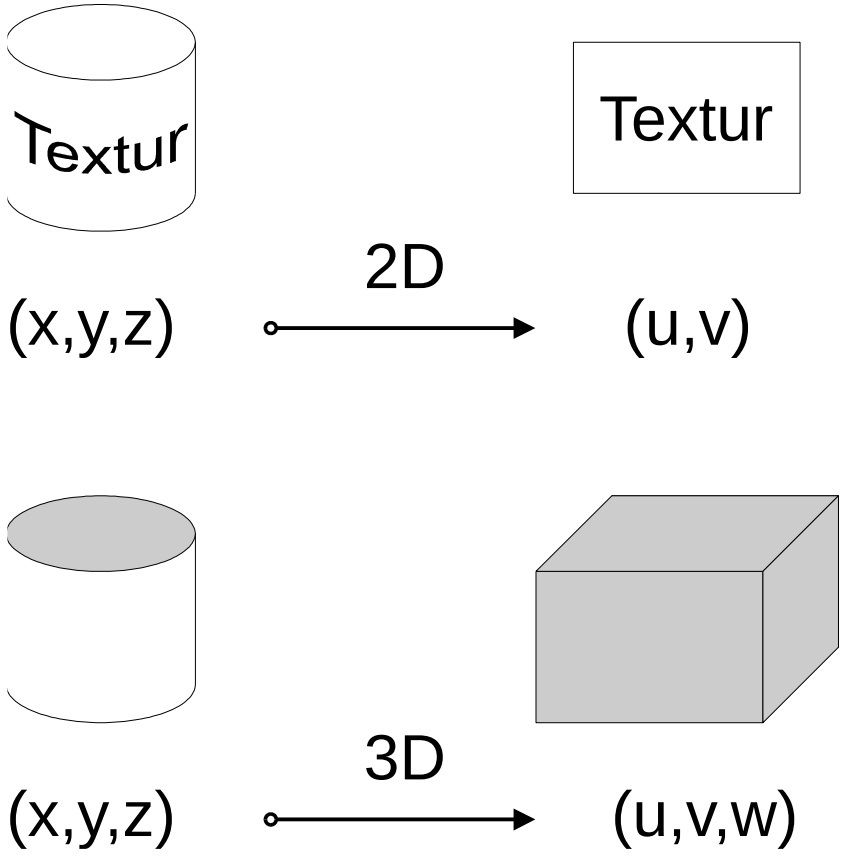
Textur – was ist das?

Textur = Eigenschaft, separat definiert

- 2D Textur: wie Aufdruck (Tapete)
- 3D Textur: innere Struktur (Holz)

Textur wird in Texturraum definiert

Aufbringung per Parametrisierung



Textur – Abbildungen

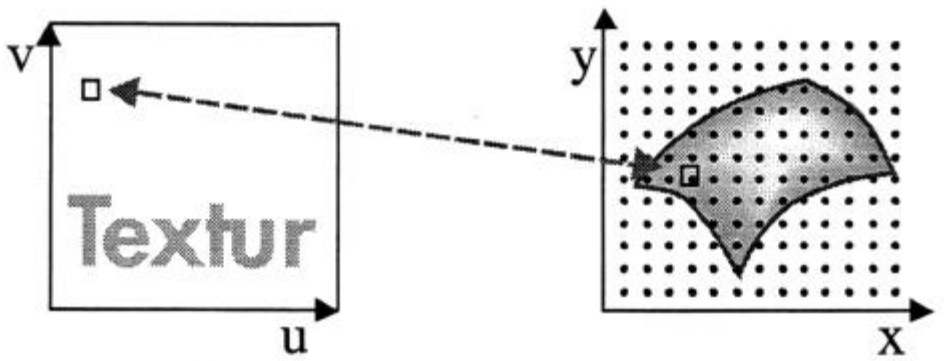
Texturraum $(u,v)^T$

Parametrisierung

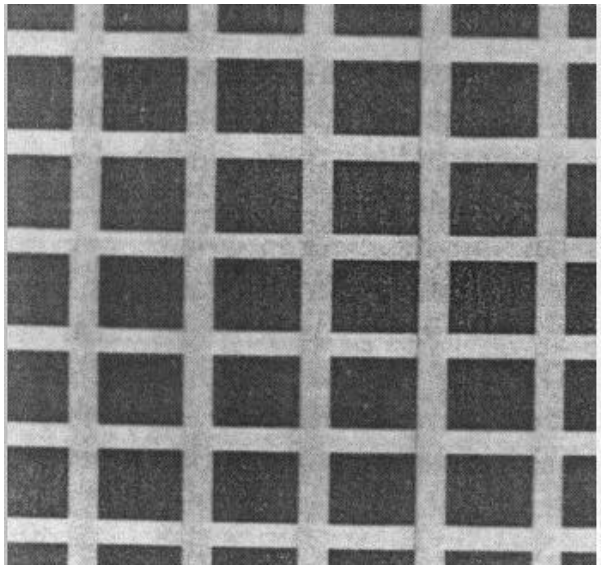
Objektraum $(x,y,z)^T$

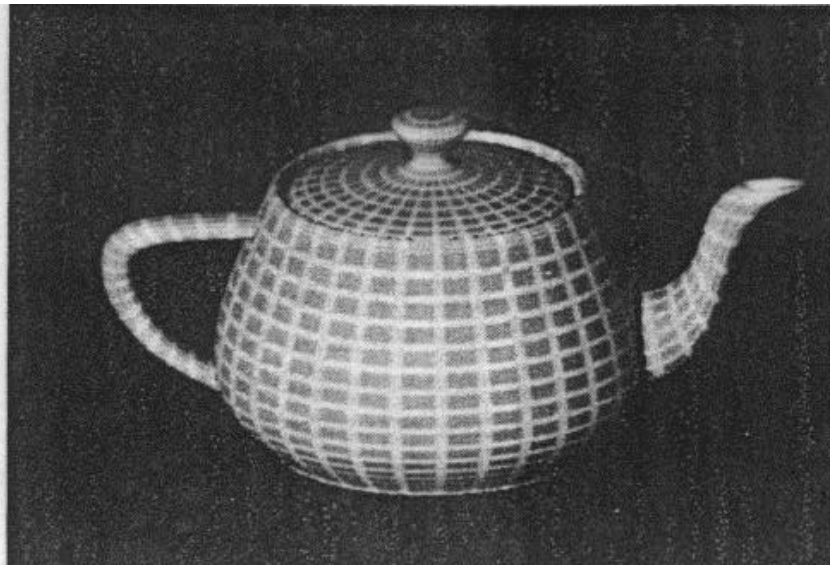
Rendering

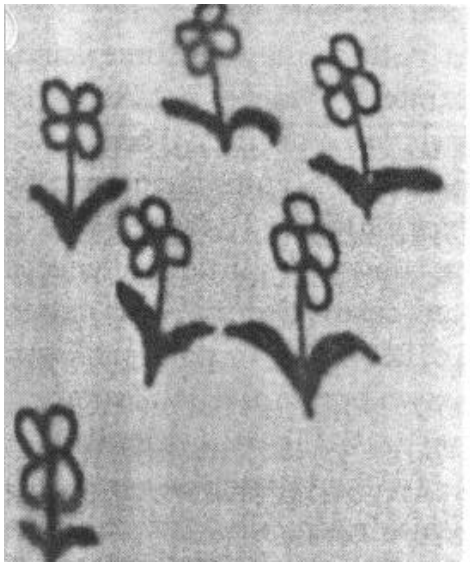
Bildraum $(x,y)^T$

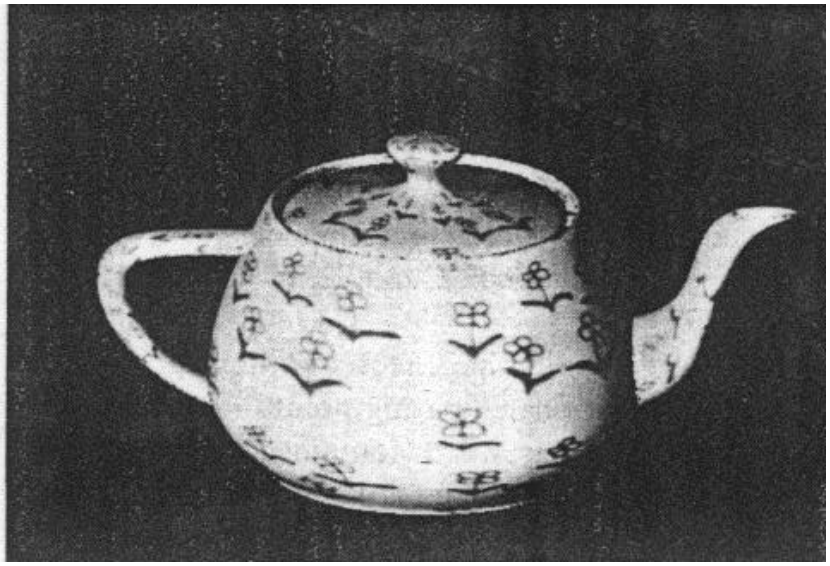


Textur – Beispiel



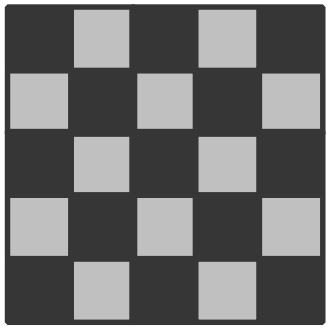


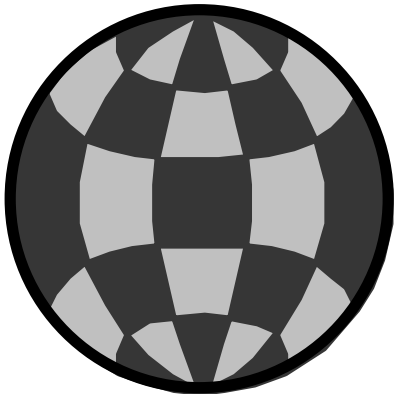




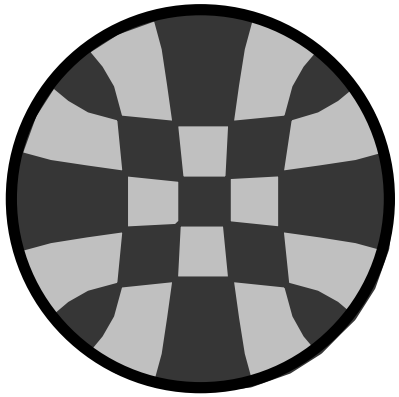
Parametrisierung

Meist verschiedene Parametrisierungen



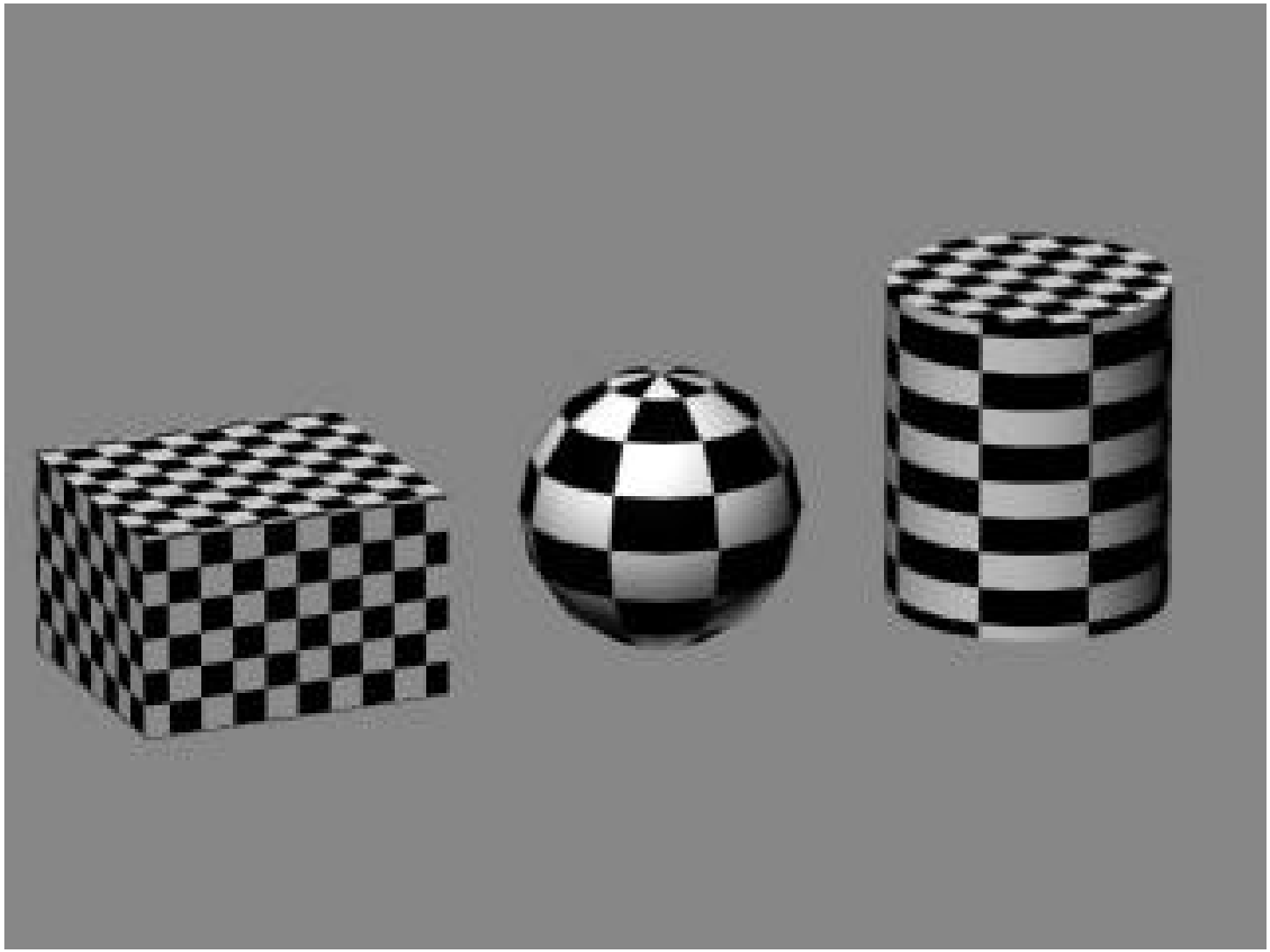


or



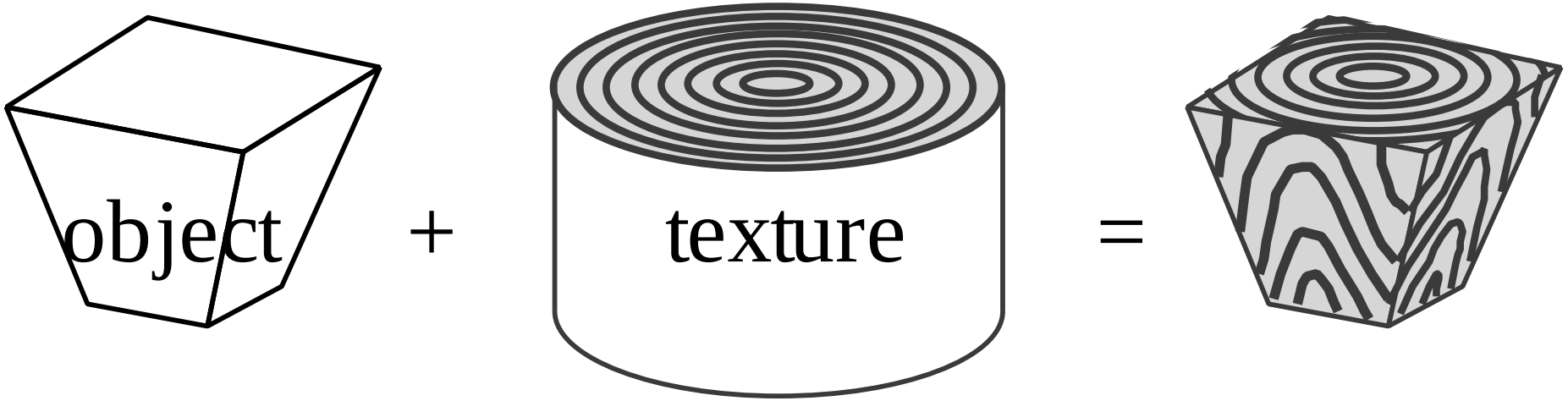
or …?

2D Texturen – Beispiel

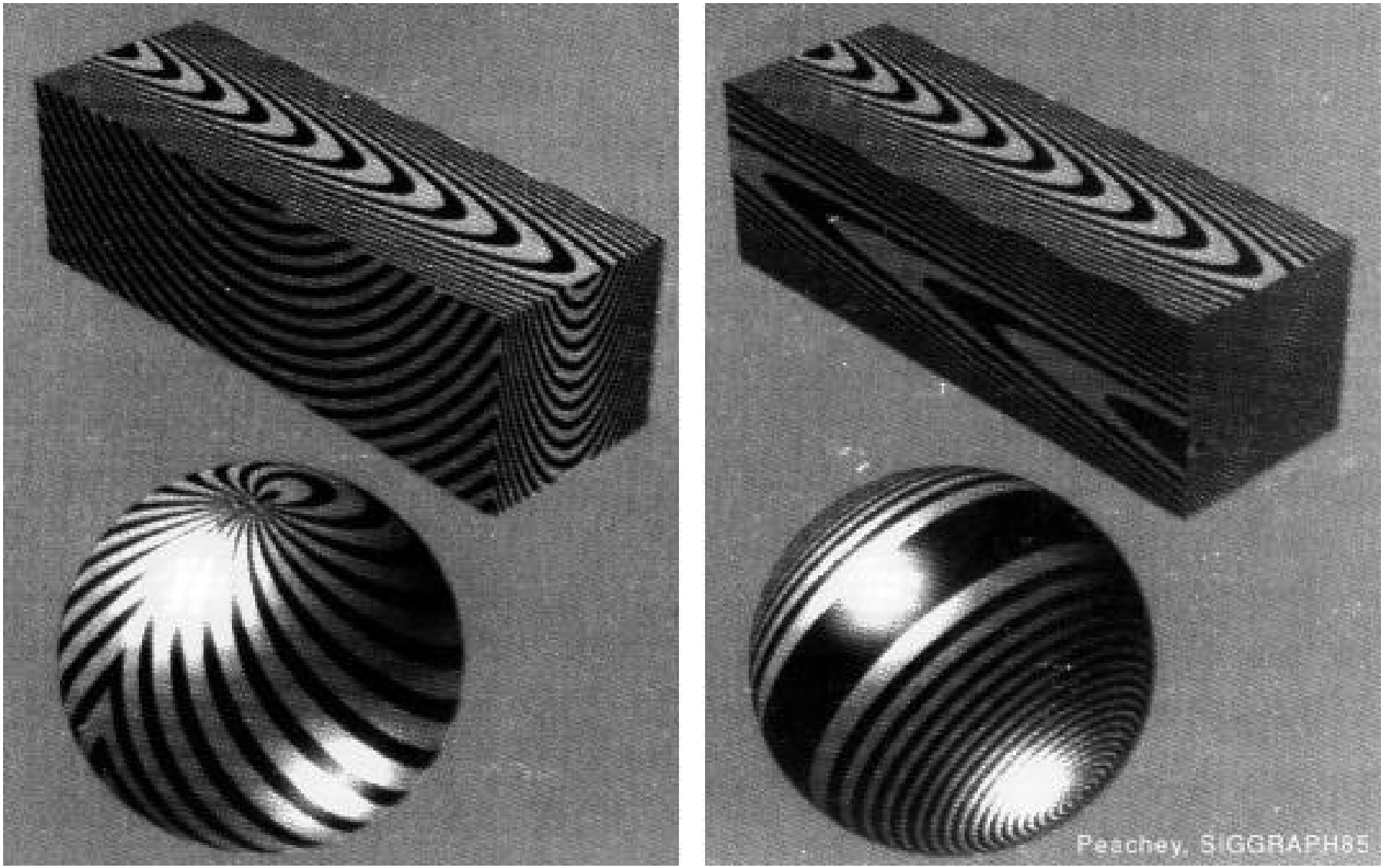


Sold Texturing

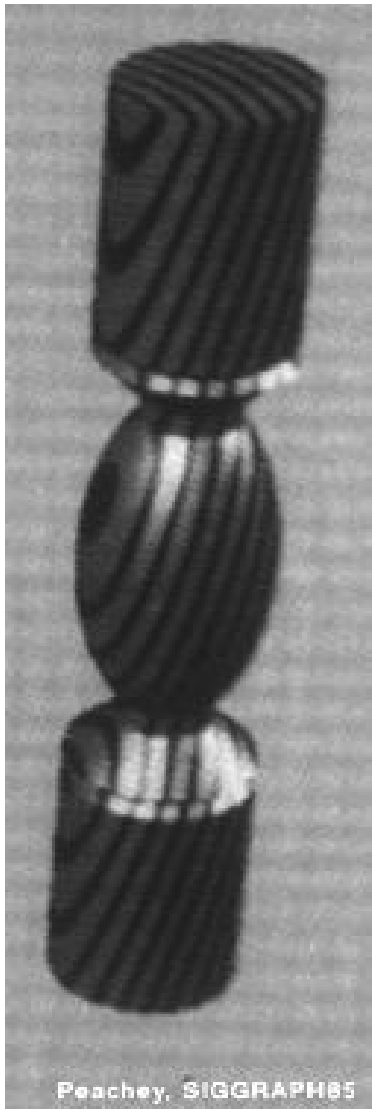
3D Textur: innere Struktur
Texturing: wie Ausschneiden

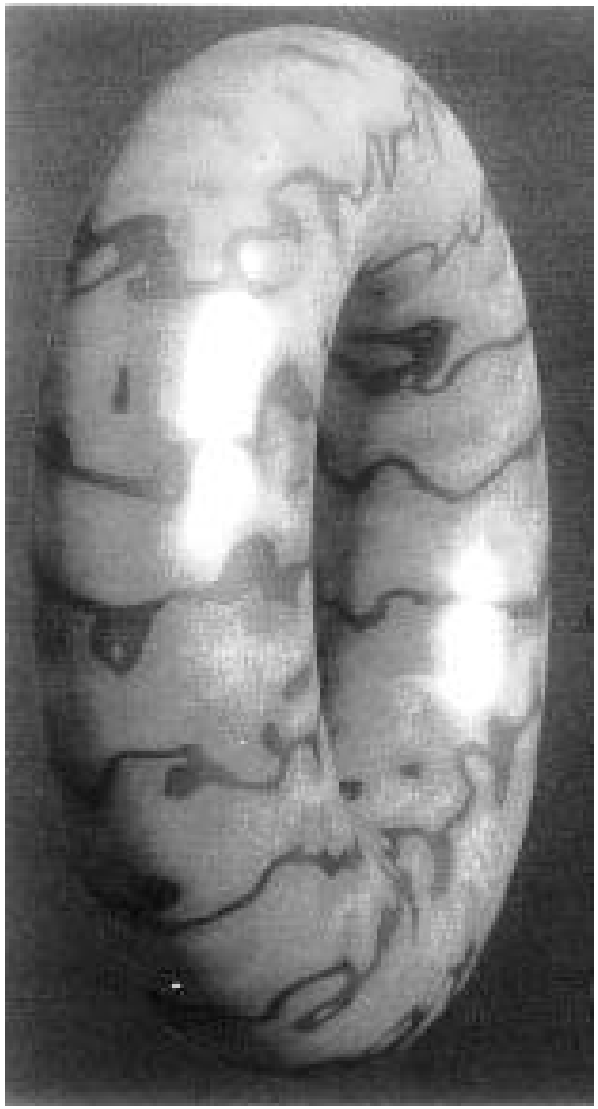


2D Textur vs. 3D Textur



Solid Texturing – mehr Beispiele

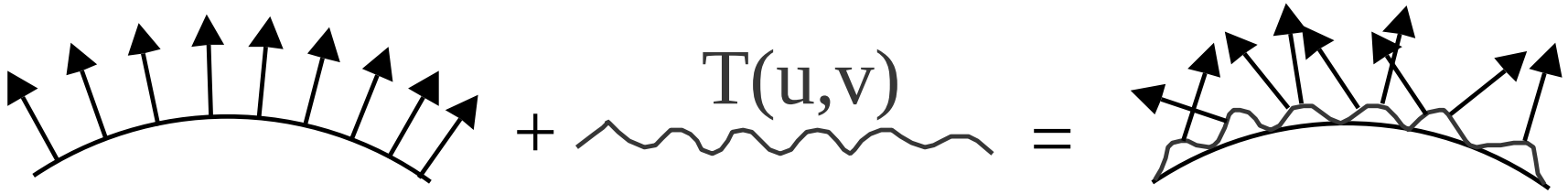


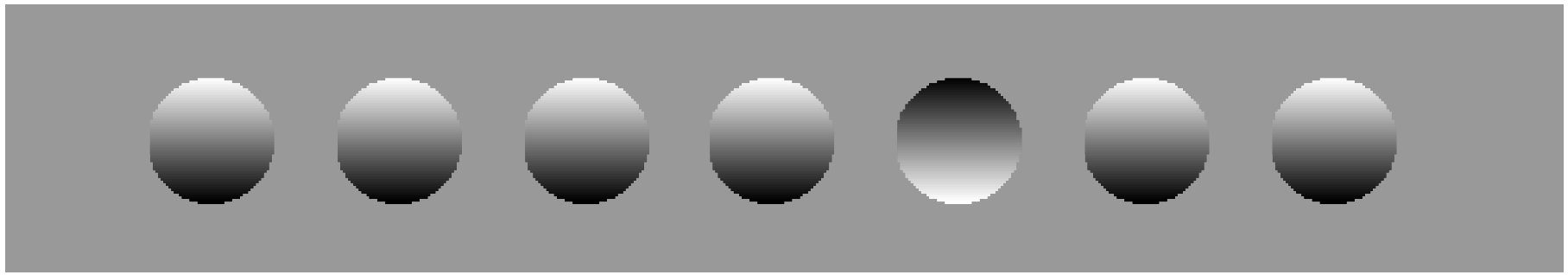


Bump Mapping

Bump Mapping =

- Vortäuschen von geometrischen Details
- Normalvektorvariation per Textur

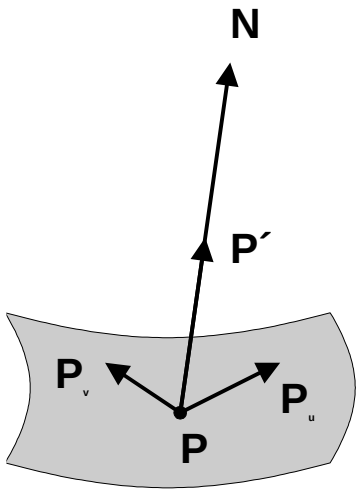




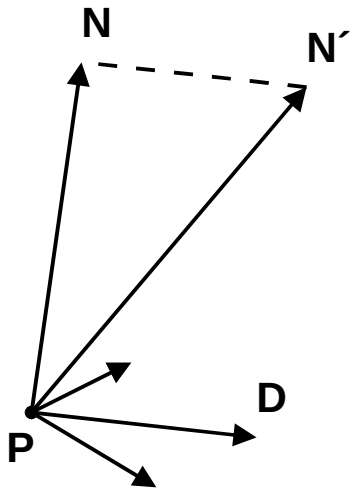
Bump Mapping (2)

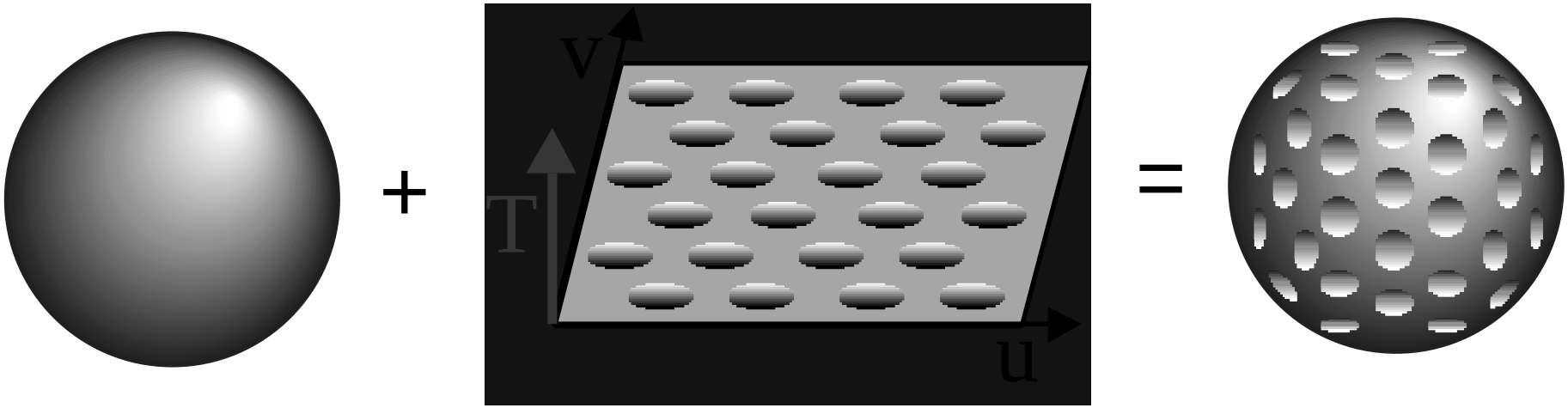
Bump Mapping =

- Vermeidung von viel Geometrie
- Normale verwackeln



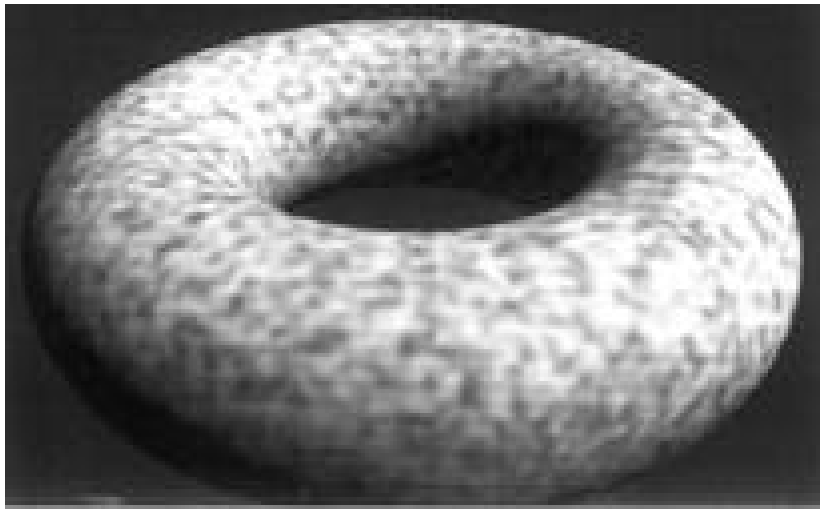
Surface



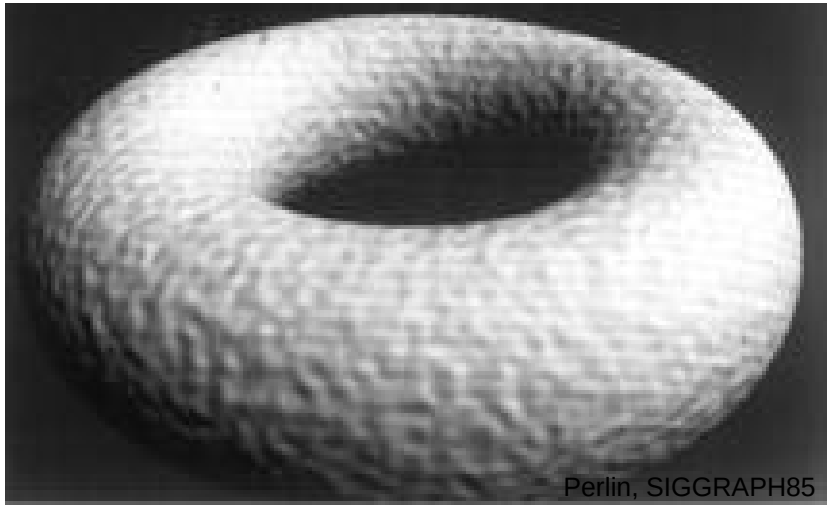


Bump Mapping – Beispiel

Normale Textur

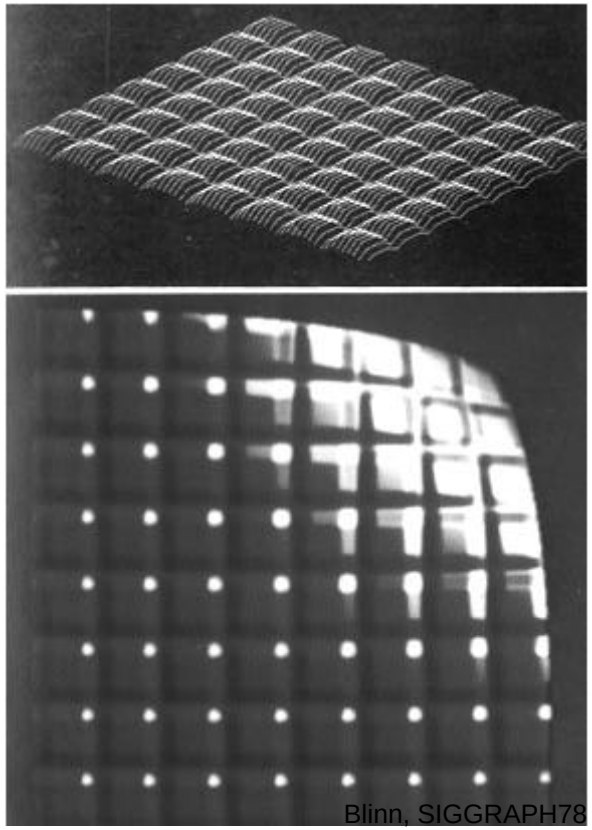


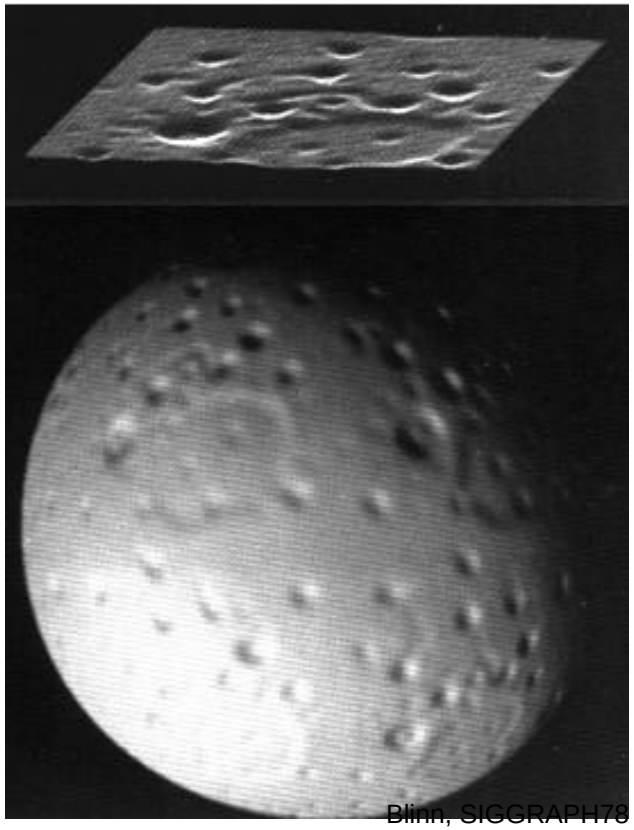
Bump Mapping

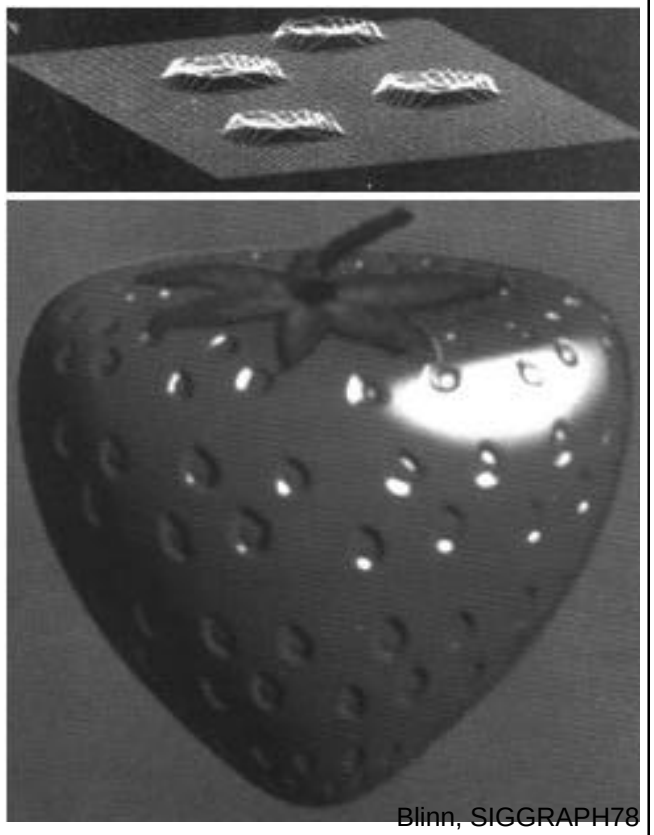


Unterschied: Shading $\rightarrow$ 3D Eindruck

Bump Mapping – mehr Beispiele





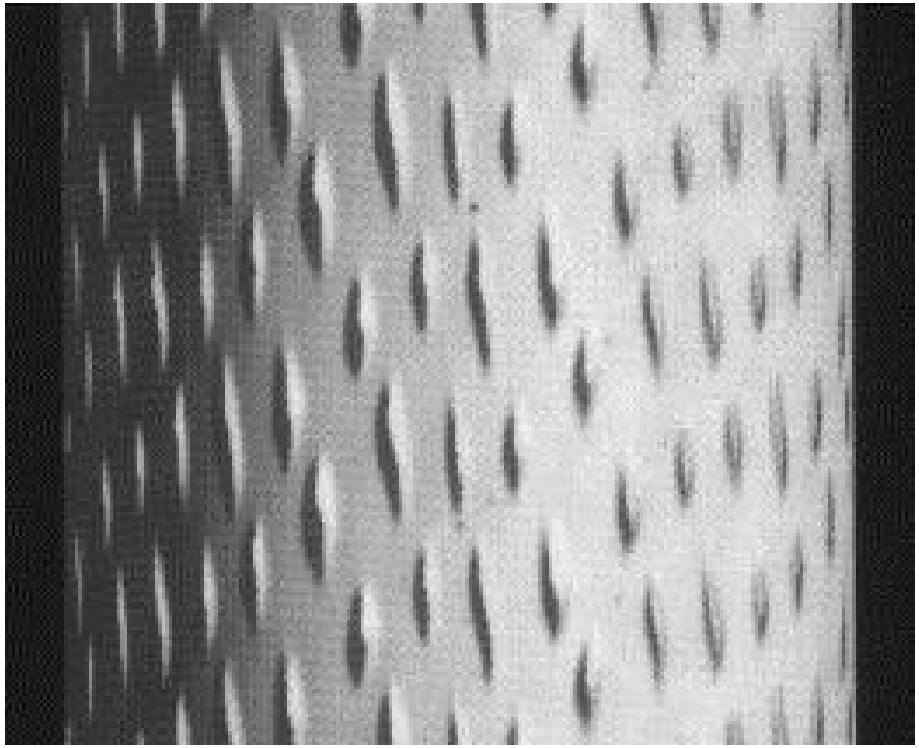


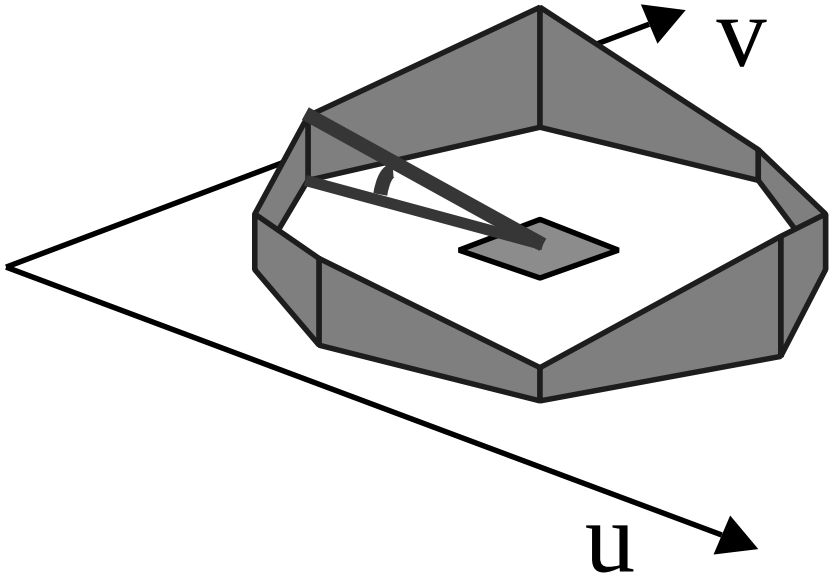
Problem: Trick sichtbar am Rand!

Problem: Bumps haben keine Schatten!

Abhilfe: Horizon Mapping

Schatten von bumps vortäuschen!



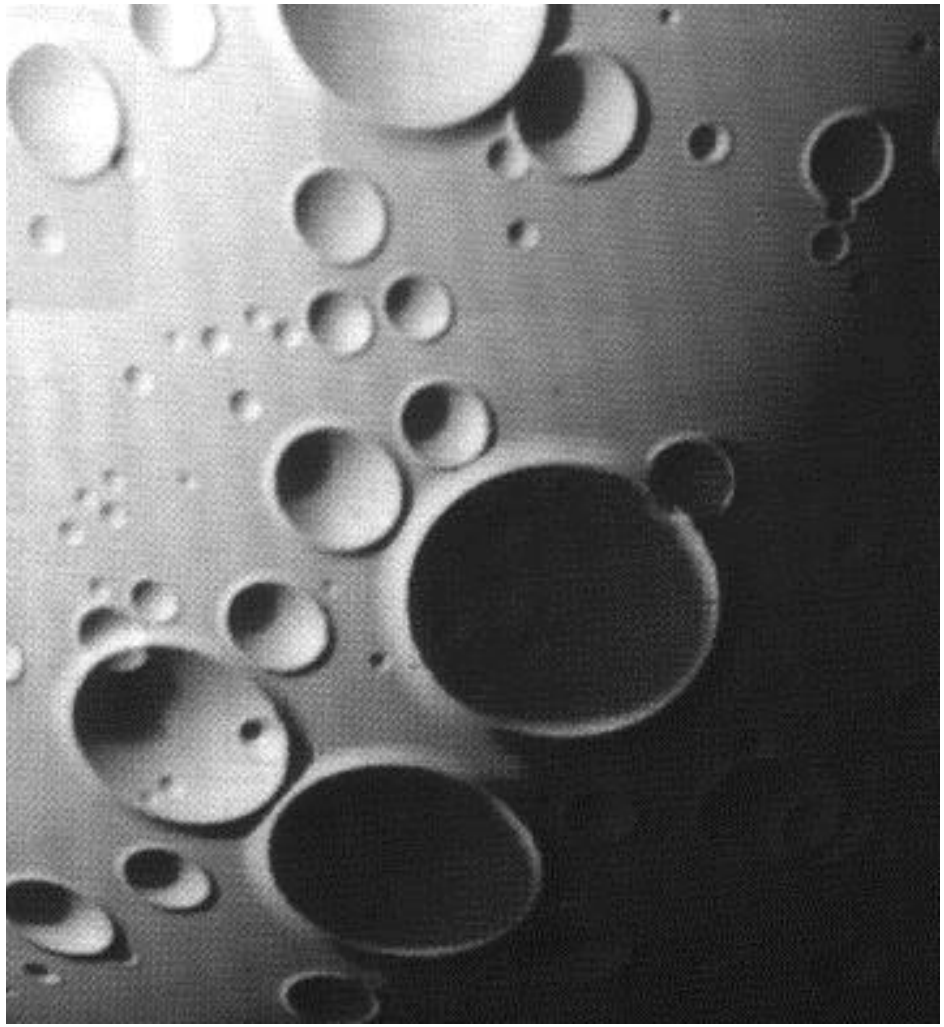


Horizont-Kontur vorberechnen

Horizon Mapping – Beispiel

Unterschied zu Bump Mapping:

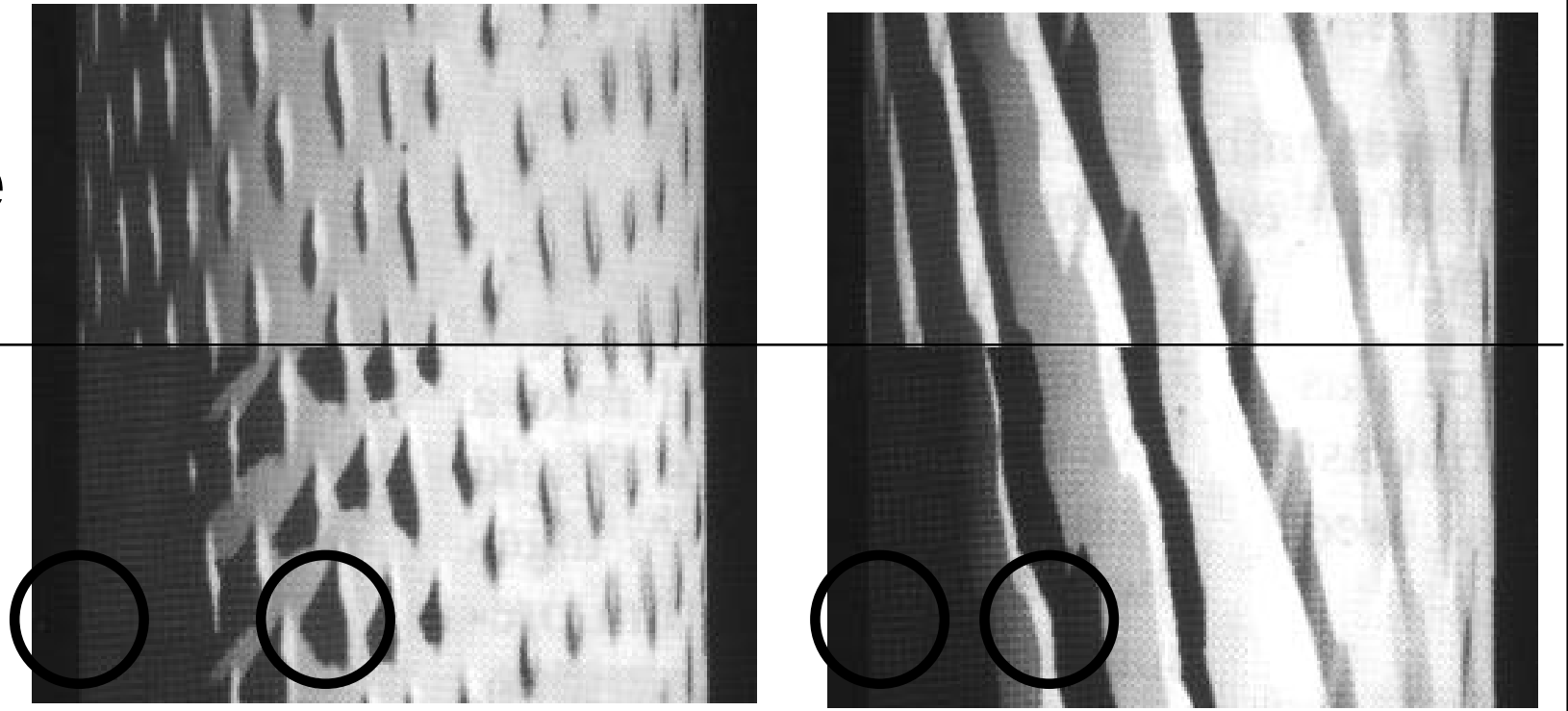
- Bumps haben Schatten
- Bumps liegen im Schatten



Horizon Mapping – Vergleich

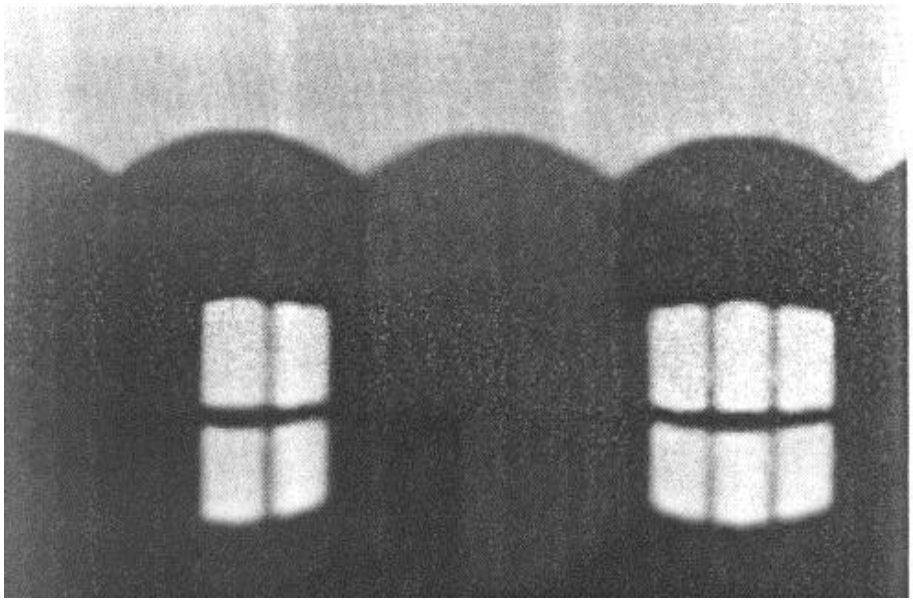
ohne

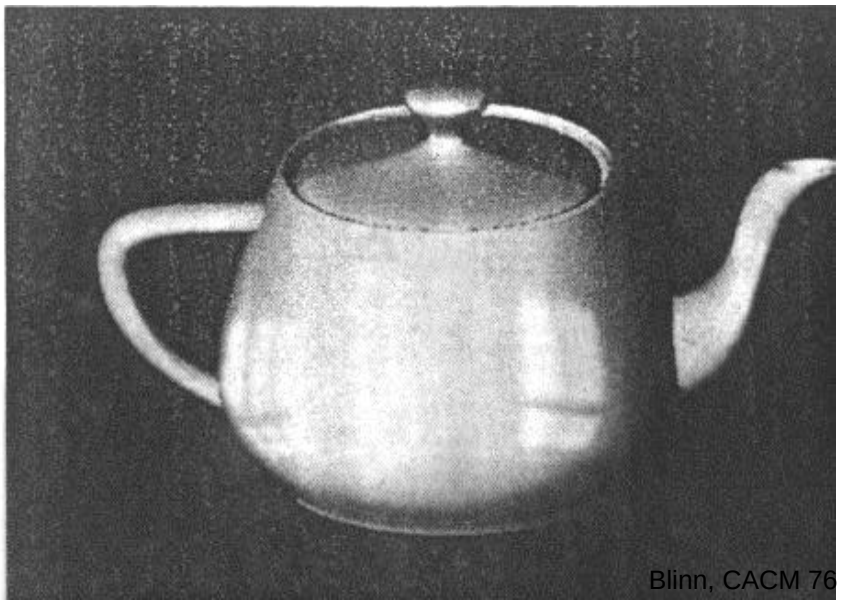
mit



Environment Mapping

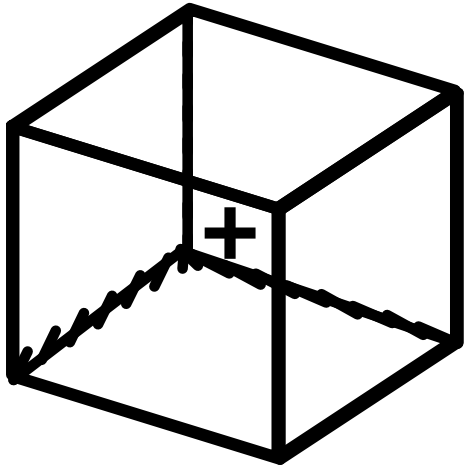
Statt komplexer Szene:
Umgebung per Textur simulieren

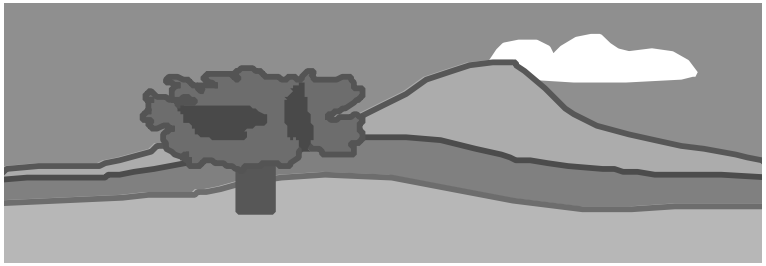


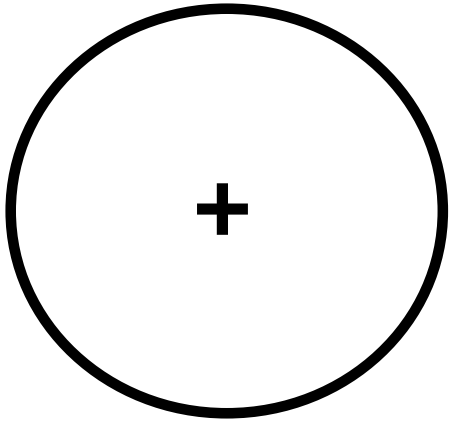


Blinn, CACM 76

6-Seiten Maps, Kugel-Maps





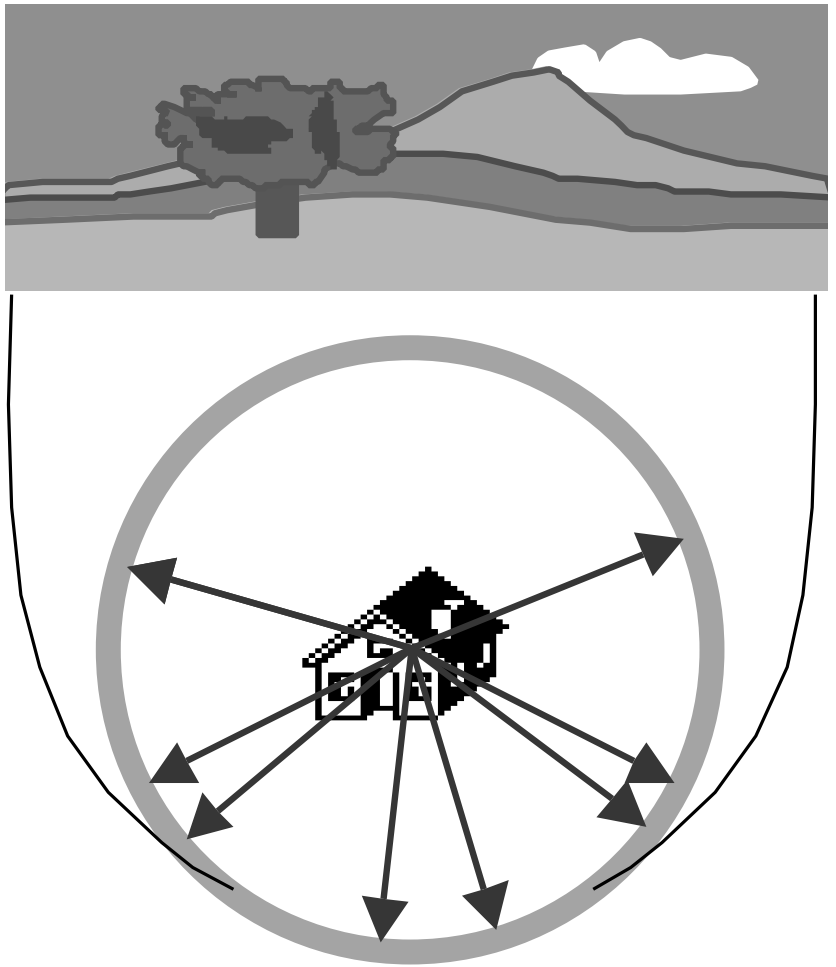


Umgebung wird zuerst auf Textur abgebildet.

Kugelförmige Map

Wenn Kugel groß:

- Speicherung in Polarkoordinaten
- Abruf nur per Richtung



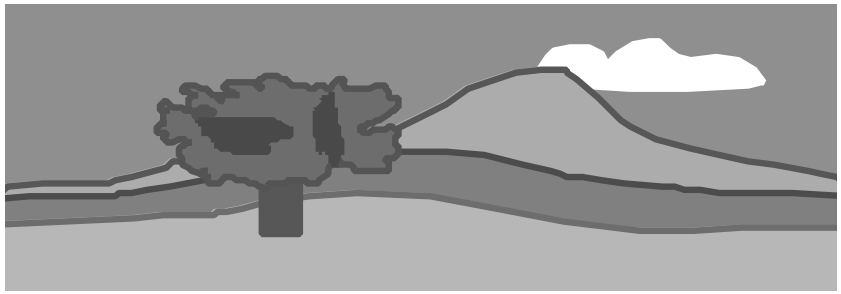
Environment Map – Prefiltern

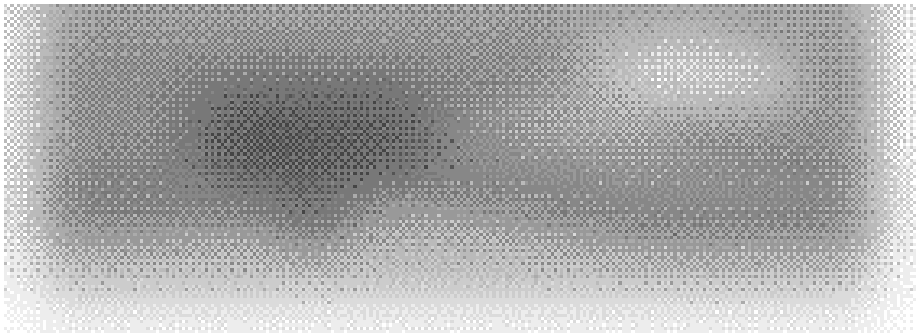
Wenn Objekte scharf reflektieren:

- 1:1 Environment map = o.k.

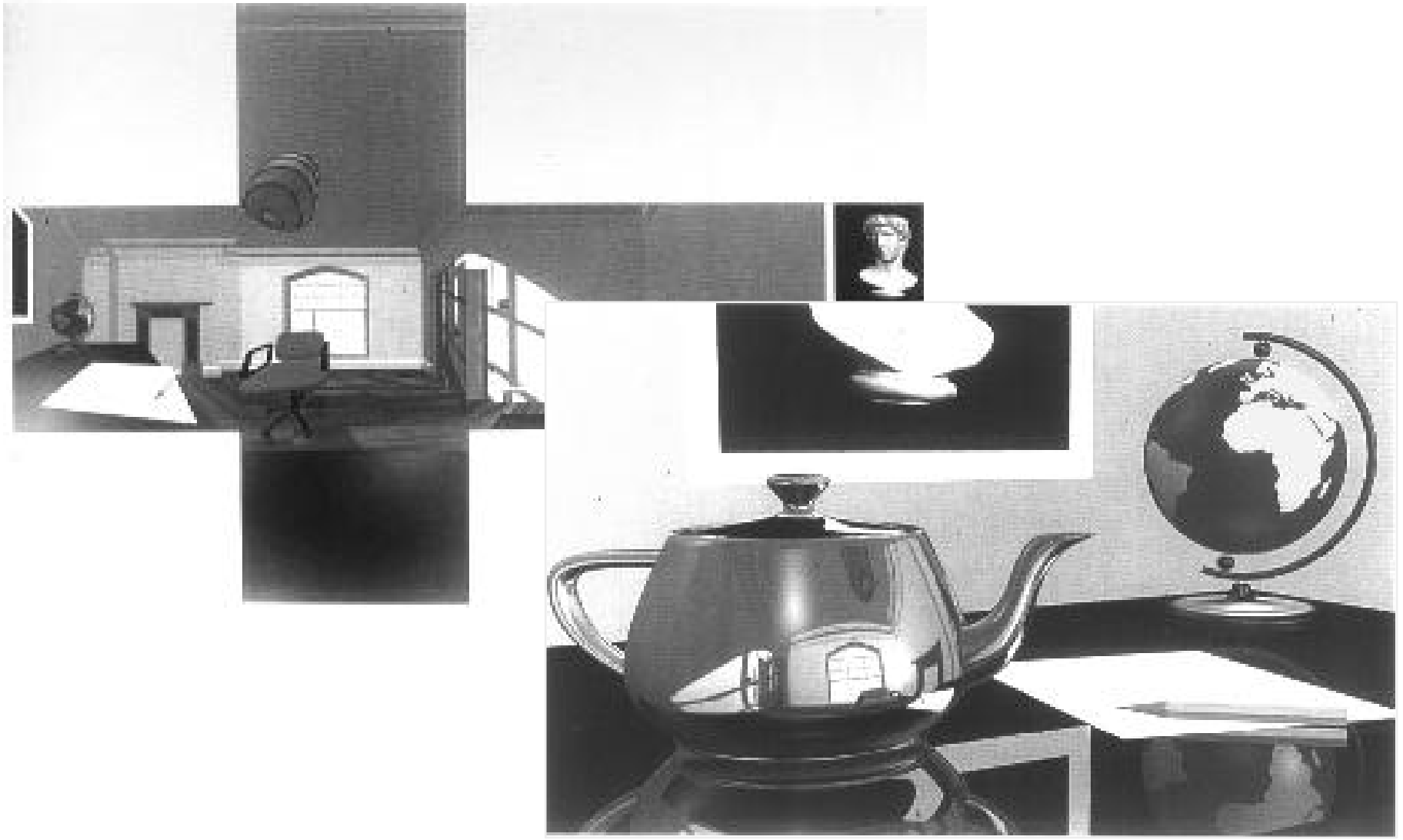
Bei diffusen Oberflächen:

- Zuerst: Preprocessing (low pass)
- Evaluation in Richtung der Flächennormale





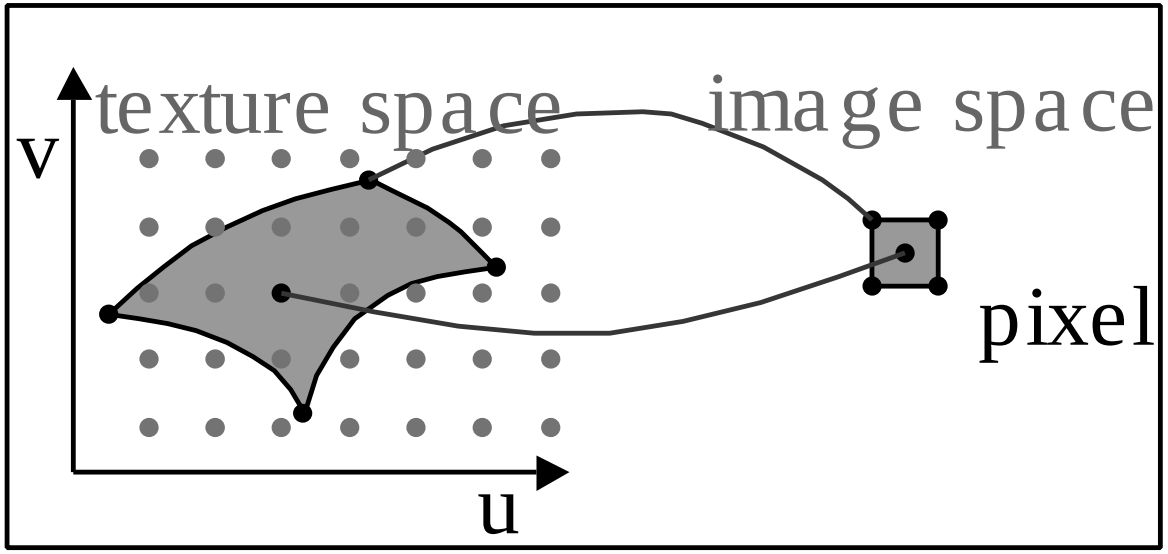
Environment Mapping – Beispiel



Aliasing-Probleme mit Texturen

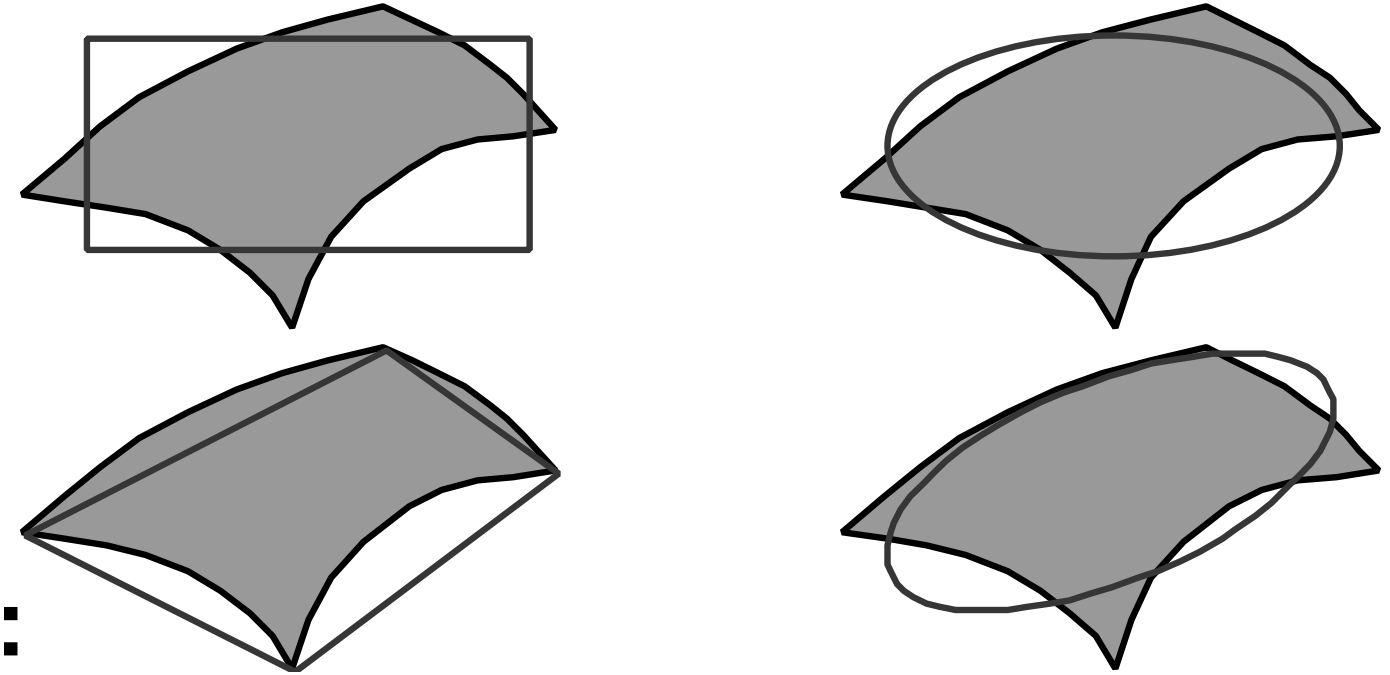
Parametrisierung nicht flächentreu!

Unter-schiedlich viel Textur pro Pixel



Anti-Aliasing von Texturen

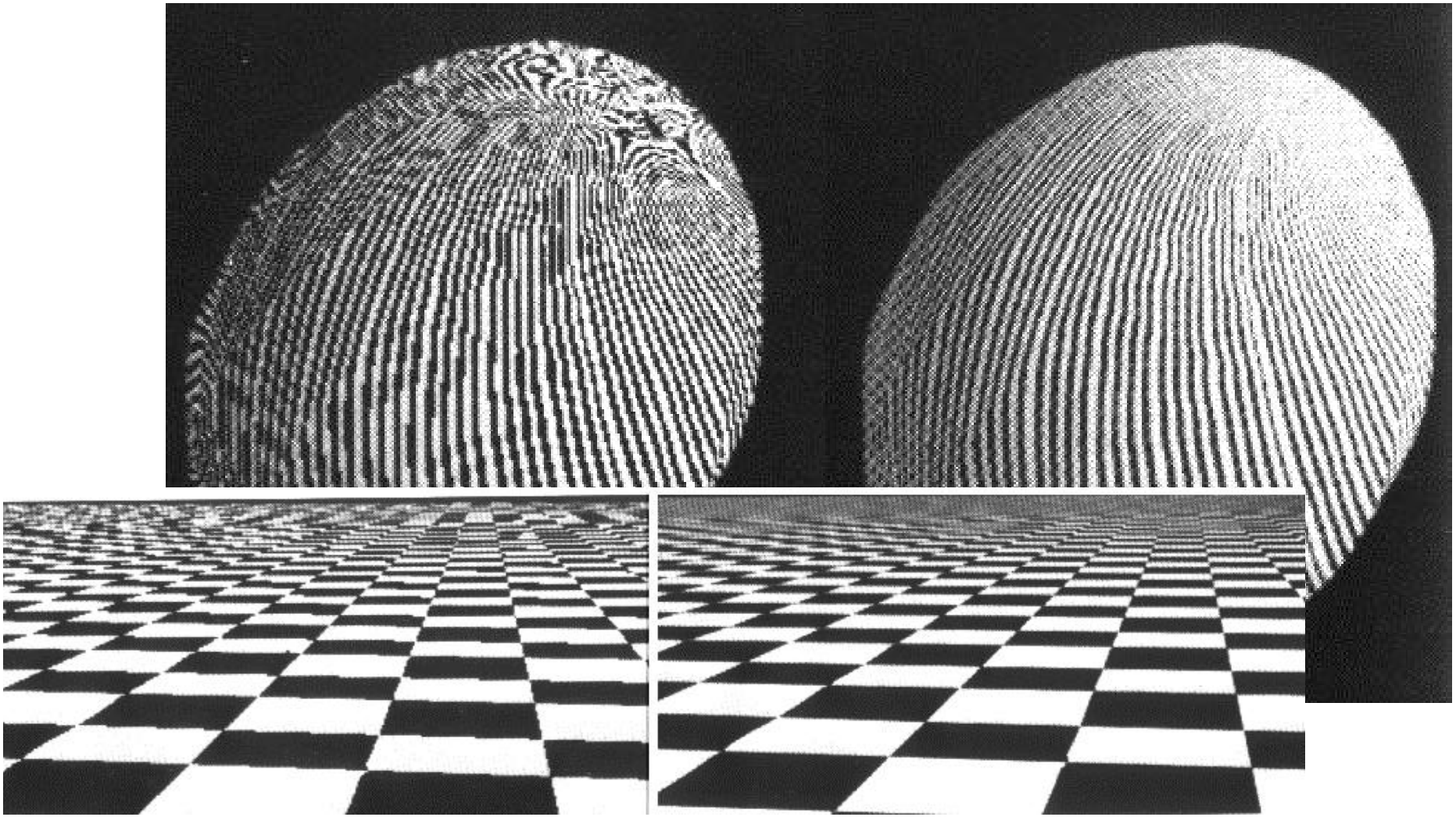
In Verwendung: Annäherungen:



Optionen:

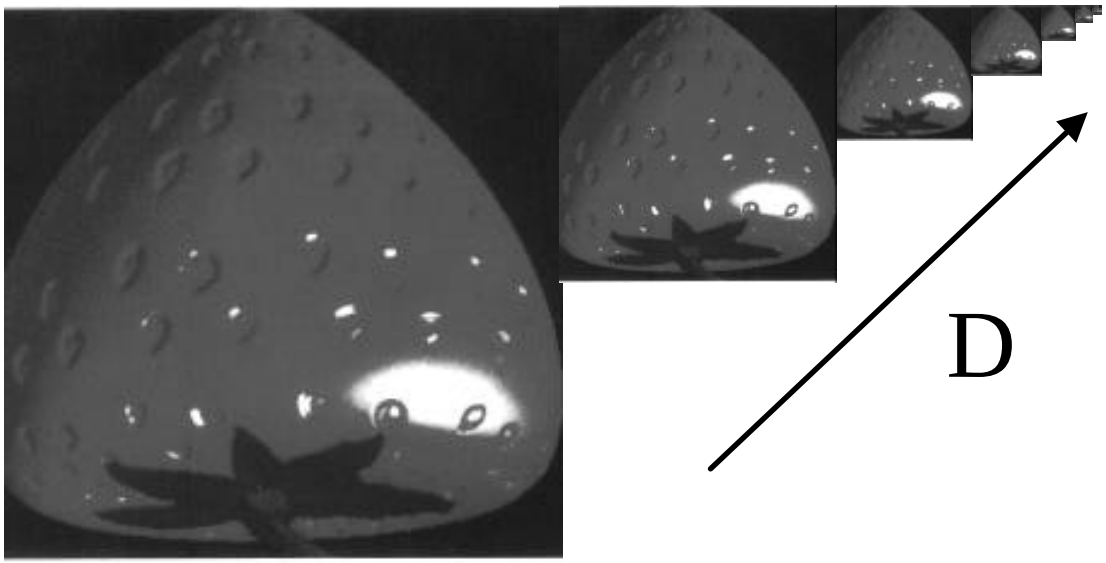
- convolution on demand
- pre-filtering

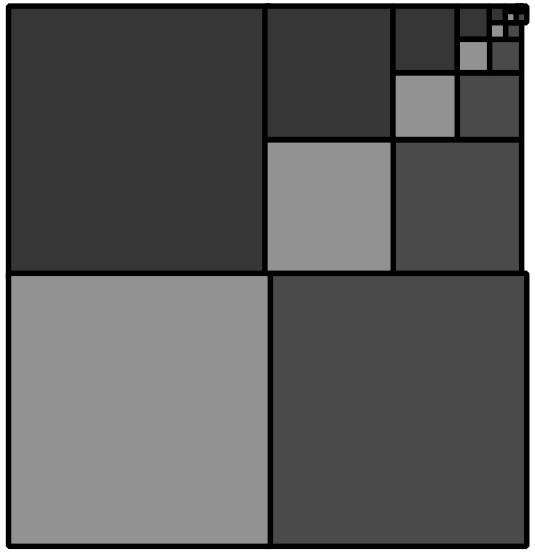
Aliasing mit Texturen – Beispiele



Mip-Mapping

Verschiedene Auflösungen vorberechnet
Drei Farben: effiziente Speicherung

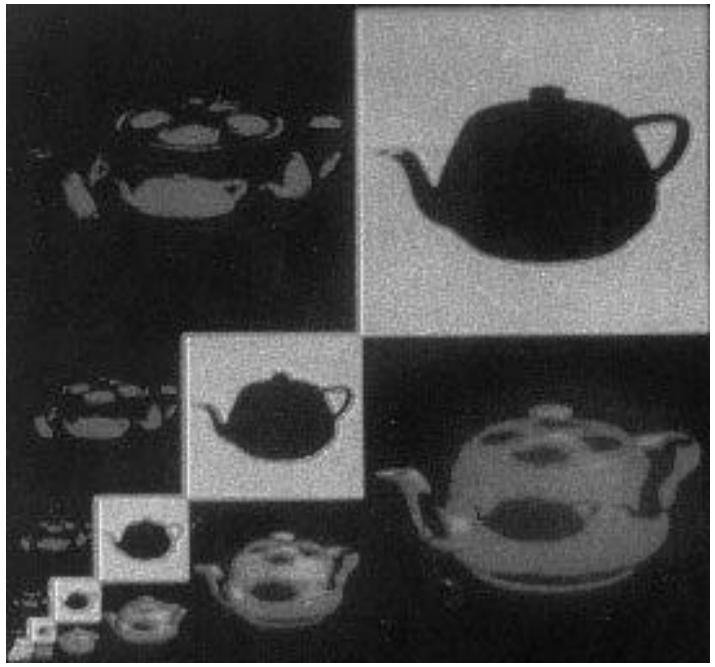


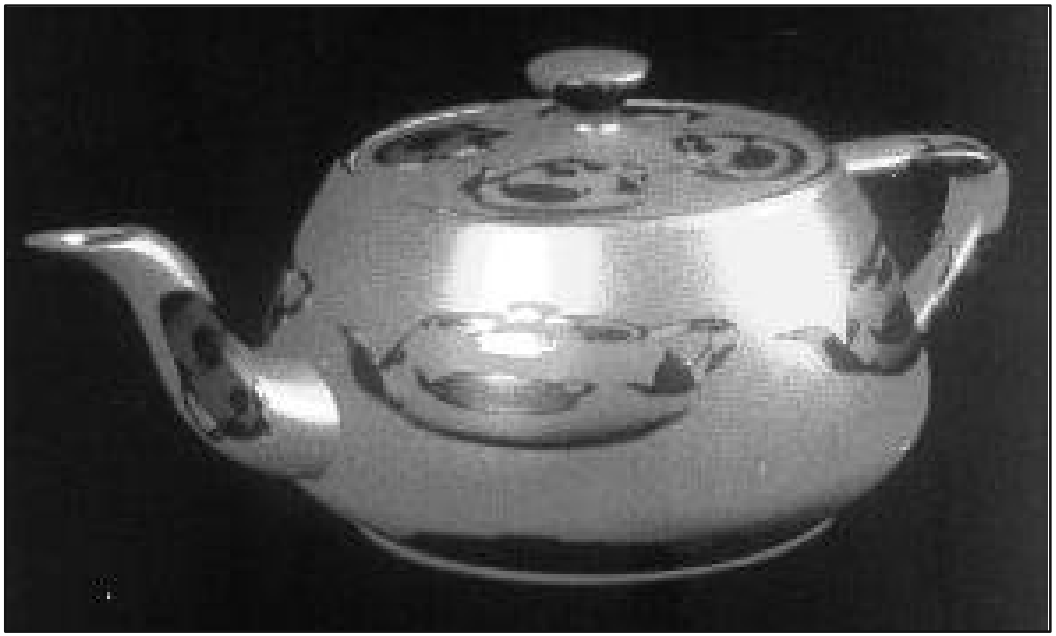


Mip: multum in parvo

Mip Mapping – Beispiel

Je nach Verzerrungsverhältnis,
wird die entsprechende Textur gewählt.



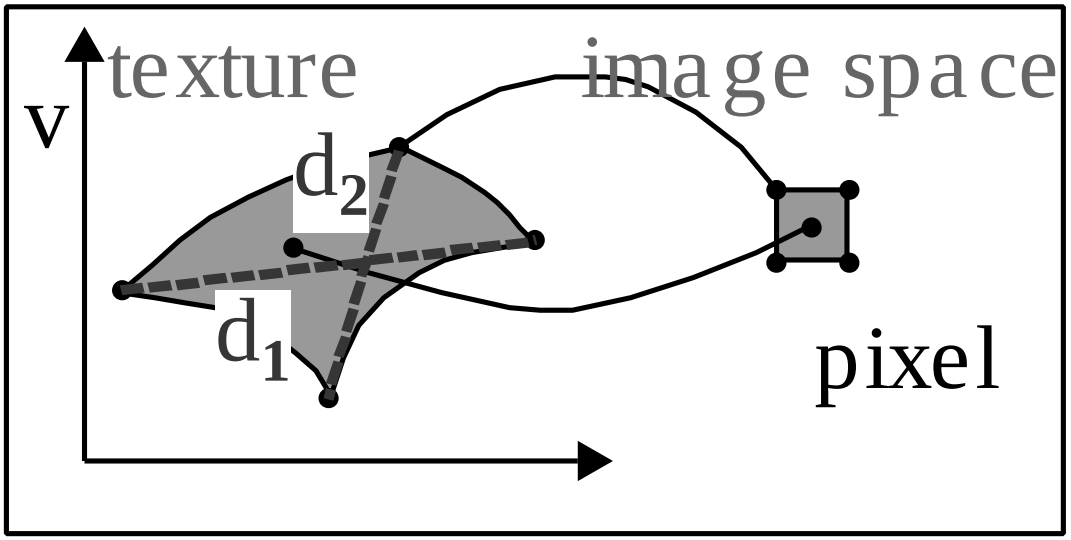


Mip Mapping – Interpolation

$2^D = \max(d_1,d_2)$

D: Texturebene

Beispiel: D=2.3



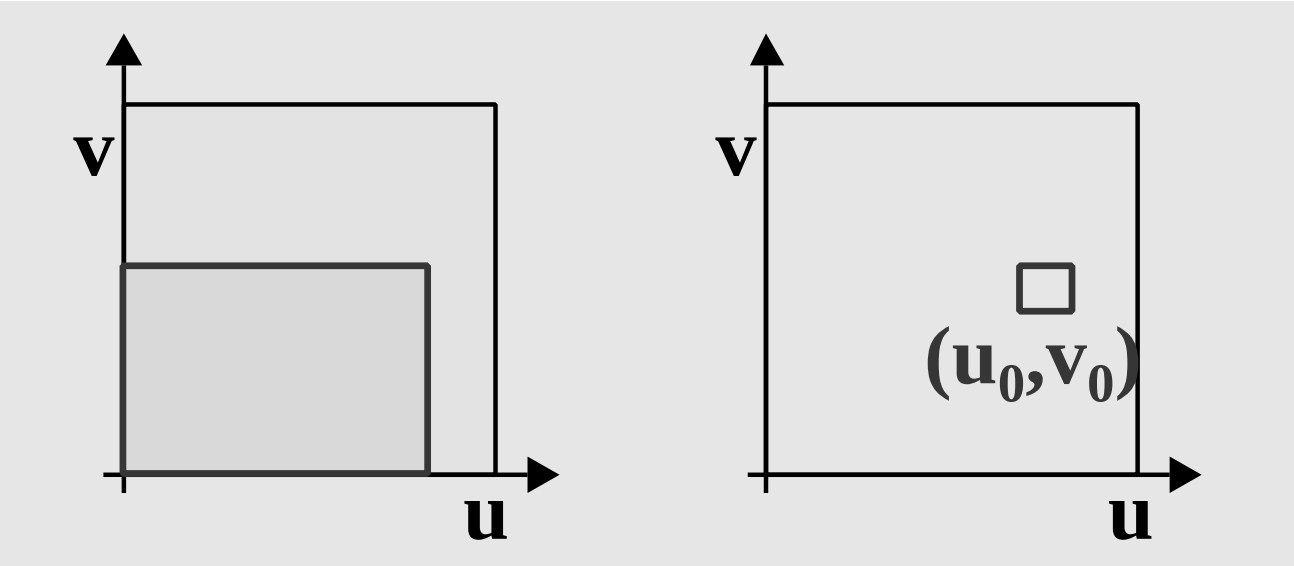
T_0 = Texturwert aus Ebene trunc(D)

T_1 = Texturwert aus Ebene trunc(D)+1

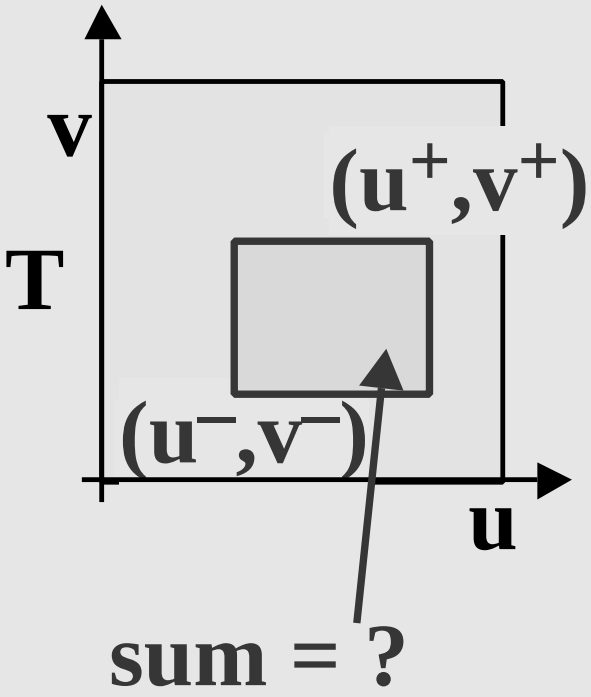
Ergebniswert: lineare Interpolation

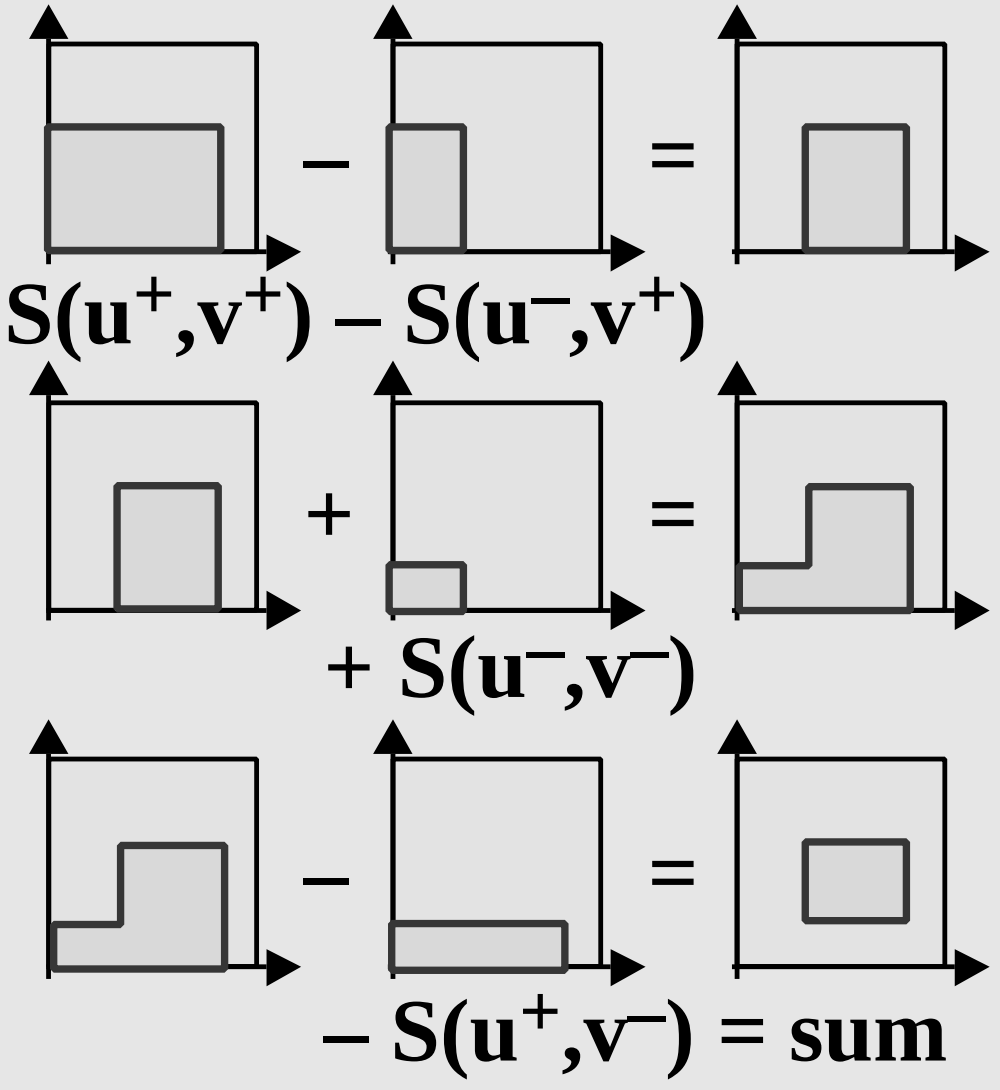
Summed Area Table

Summen speichern statt Texturwerte:



Summed Area Table – Evaluieren





Aufwand: konstant!

Anti-Aliasing von Texturen – Bsp.