

Gittertypen und Interpolation

Christof Rezk-Salama

Visualisierung WS 04/05, 19.10.2004

computergraphik und multimedia systeme
universität siegen

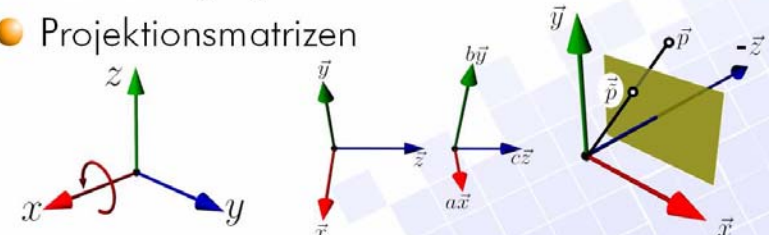


Review: Letzte Stunde

Wiederholung Lineare Algebra:

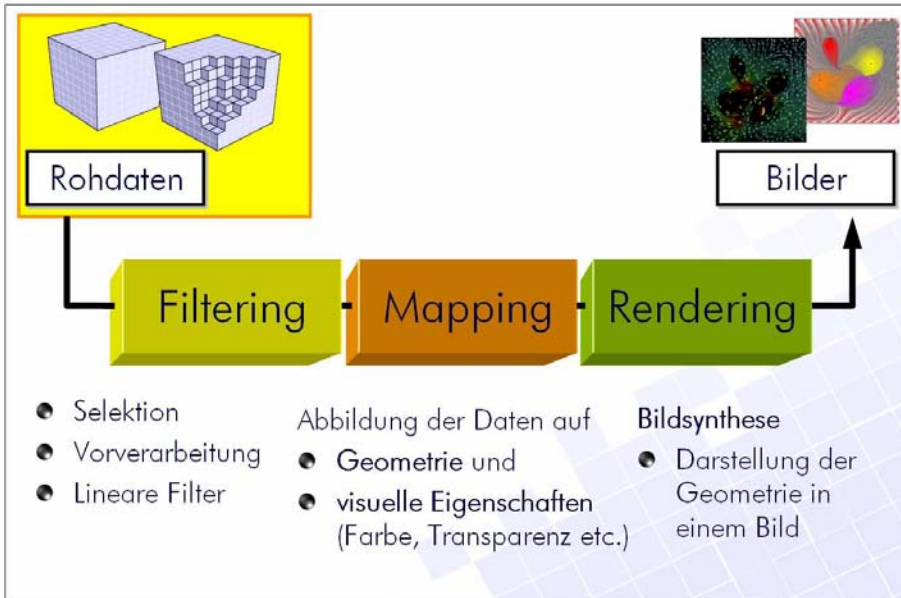
Homogene Koordinaten:

- erlauben Gleichbehandlung von Punkten (Ortsvektoren) und Richtungsvektoren
- Rotation, Translation und Skalierung (Affine Abbildungen) in einer Transformationsmatrix
- Projektionsmatrizen



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Review: Letzte Stunde



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Heute: Gitterstrukturen

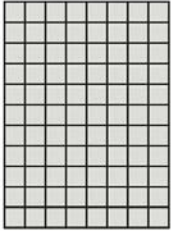
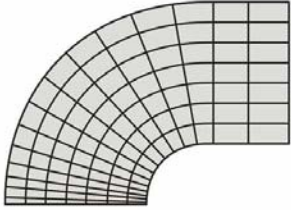
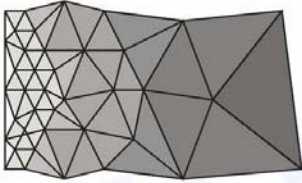
Wozu brauchen wir Gitter?

- Natürliche Phänomene sind in der Regel kontinuierlich (d.h. unendlich hoch aufgelöst)
- Computer können (natürlich) nur mit endlichen Datenmengen arbeiten
- Kontinuierliche Daten müssen **abgetastet** (digitalisiert, rasterisiert) werden, um im Computer verarbeitet werden zu können.

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

2D Gittertypen

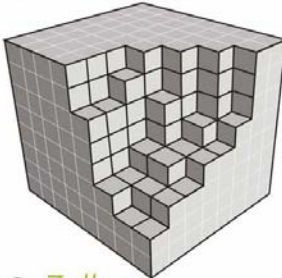
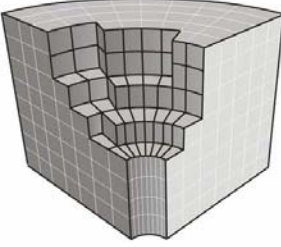

5

uniform rectilinear	curvilinear	unstrukturiert
		
<ul style="list-style-type: none">● Zellen: gleichförmige Rechtecke, Quadrate● Anordnung: uniform regulär	<ul style="list-style-type: none">● Zellen: unterschiedliche Vierecke,● Anordnung: nicht uniform, aber strukturiert (Topologie)	<ul style="list-style-type: none">● Zellen: Dreiecke, (selten Rechtecke, Polygone)● Anordnung: unstrukturiert

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

3D Gittertypen

6

uniform rectilinear	curvilinear	unstrukturiert
		
<ul style="list-style-type: none">● Zellen: gleichförmige Hexaeder,● Anordnung: uniform regulär	<ul style="list-style-type: none">● Zellen: unterschiedliche Hexaeder,● Anordnung: nicht uniform, aber strukturiert	<ul style="list-style-type: none">● Zellen: Tetraeder, Hexaeder, Pyramiden, Prismen● Anordnung: unstrukturiert

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Allgemeine Gitterstrukturen

7

● Speicherbedarf für Gitter

Was muß gespeichert werden?

Allgemein:

- Dimension
- Liste der Vertices (Gitterpunkte) [explizit oder implizit]
- Liste der Zellen [explizit oder implizit]

Optional (redundant):

- Liste der Kanten
- Liste der Flächen (3D)
- Kreuzreferenzen: Zelle → Vertices, Zelle → Flächen, Zelle → Kanten, Fläche → Kanten, usw.

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Uniforme Gitter

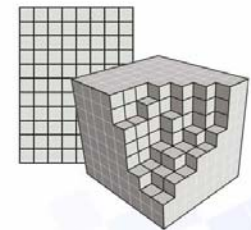
8

● Strukturiertes Gitter

Was muß gespeichert werden?

Zellen und Vertices implizit:

- Dimension N_x, N_y, N_z : Anzahl der Abtastpunkte in x-, y- und z-Richtung
- Zellgröße/Gitterweite $\Delta x, \Delta y, \Delta z$: Abstand der Abtastpunkte in x-, y- und z-Richtung
- Datenwerte als Array mit Index n
 $n = N_x \cdot N_y \cdot z + N_x \cdot y + x$



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

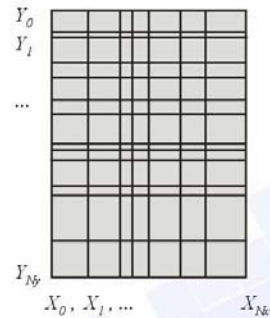
Rectilineare Gitter

9

Strukturiertes Gitter

Was muß gespeichert werden?
Zellen implizit, Vertices explizit:

- Dimension N_x, N_y, N_z :
Anzahl der Abtastpunkte in x-, y- und z-Richtung
- Koordinaten der Linien/Ebenen:
 - X_0, X_1, \dots, X_{N_x} ,
 - Y_0, Y_1, \dots, Y_{N_y} ,
 - Z_0, Z_1, \dots, Z_{N_z} ,
- Datenwerte als Array mit Index n
 $n = N_x \cdot N_y \cdot z + N_x \cdot y + x$



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

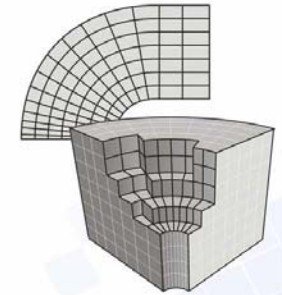
Curvilineare Gitter

10

Strukturiertes Gitter

Was muß gespeichert werden?
Zellen implizit, Vertices explizit:

- Dimension N_x, N_y, N_z :
Anzahl der Abtastpunkte in x-, y- und z-Richtung
- Liste der Gitterpunkte P_{ijk} :
x-, y- und z-Koordinaten der Gitterpunkte
- Datenwerte als Array mit Index n
 $n = N_x \cdot N_y \cdot z + N_x \cdot y + x$



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Unstrukturiertes Gitter

11

Unstrukturiertes Gitter

Was muß gespeichert werden?
Zellen und Vertices explizit:

- Anzahl der Gitterpunkte N
- Anzahl der Zellen Z
- Liste der Gitterpunkte P_{ijk} :
x-, y- und z-Koordinaten der Gitterpunkte
- Liste der Zellen:
z.B. Tetraeder: 4 Indices der Gitterpunkte



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Zusammenfassung Gittertypen

12

Uniform rectilinear:

Vorteile: einfach, effizienter Speicherzugriff, Zellsuche ist trivial,

Nachteile: überall gleiche Auflösung, keine lokale Verfeinerung möglich

Curvilinear:

Vorteile: flexibler als rectilineare Gitter

Nachteile: Gitterstruktur muß zusätzlich gespeichert werden, Algorithmus zur Zellsuche erforderlich

Unstrukturiert:

Vorteile: sehr hohe Flexibilität, lokale Verfeinerung möglich

Nachteile: Position jedes Datenpunktes und Zellen müssen explizit gespeichert werden, Zellsuche erforderlich

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Beispiel: Dreiecksnetz

13

„Winged Edge“

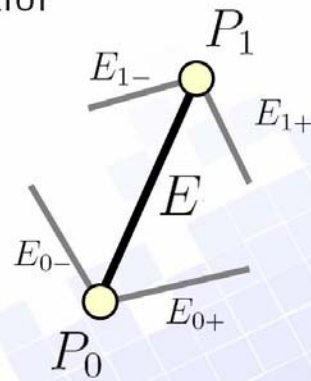
kantenbasierte Datenstruktur

Für jede Kante E :

- 2 Referenzen auf Vertices
- 4 Referenzen auf benachbarte Kanten

Für jeden Vertex P :

- Position (3 Koordinaten)
- 1 Referenz auf eine seiner Kanten



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Beispiel: Dreiecksnetz

14

„Shared Vertex“

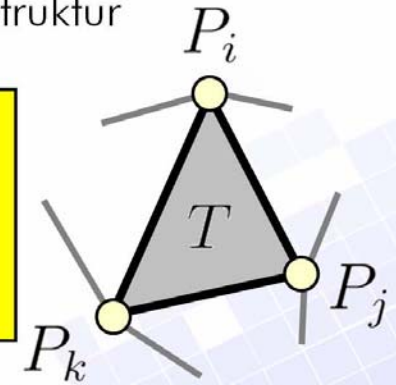
flächenbasierte Datenstruktur

Für jeden Vertex P :

- Position (3 Koordinaten)

Für jedes Dreieck T :

- 3 Referenzen auf Vertices



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Beispiel: Dreiecksnetz

15

Welche Datenstruktur ist effizienter?

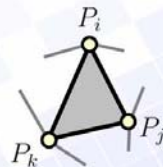
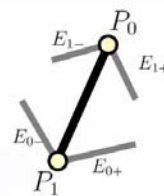
Faustregel: Dreiecksnetz mit n Vertices, hat ca. $3n$ Kanten und $2n$ Flächen

„Winged Edge“

benötigt ca. $(3+1)n + 6 \cdot 3n = 22n$ Werte

„Shared Vertex“

benötigt ca. $3n + 3 \cdot 2n = 9n$ Werte



ABER: Winged Edge ermöglicht schnellere Algorithmen, da die Traversierung (z.B. Aufzählung aller Kanten zu einem geg. Vertex, aller Vertices zu einem geg. Dreieck, etc.) in *konstanter* Zeit möglich ist!

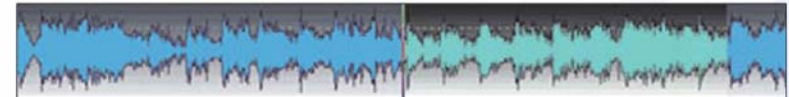
christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Abtastung

16

Beispiel: Ein Audiosignal

- ist in der Natur eine Schallwelle, d.h. eine Schwingung eines Mediums (z.B. Luft)
- Im Rechner werden Audiosignale *gesampelt*, d.h. *abgetastet* und *quantisiert* (AD-Wandlung)



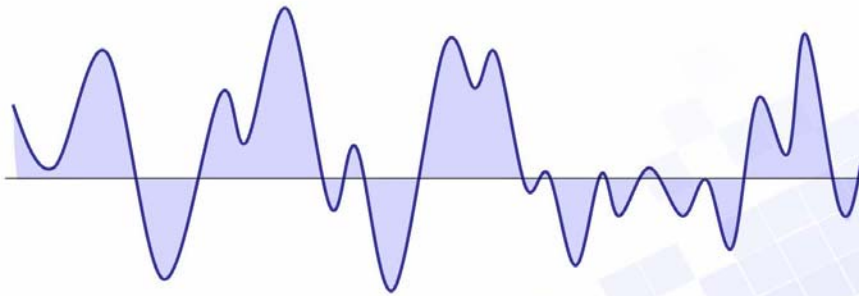
Abtasttheorem: Wie muß das Signal digitalisiert werden, damit bestimmte Eigenschaften erhalten bleiben?

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Diskrete Darstellung

17

Kontinuierliche Signale werden abgetastet,

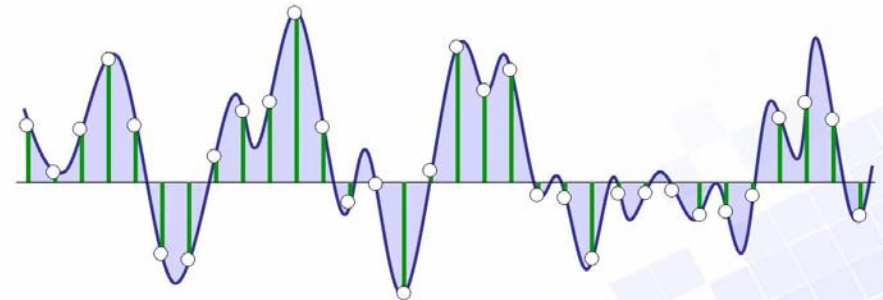


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Diskrete Darstellung

17

Kontinuierliche Signale werden abgetastet,

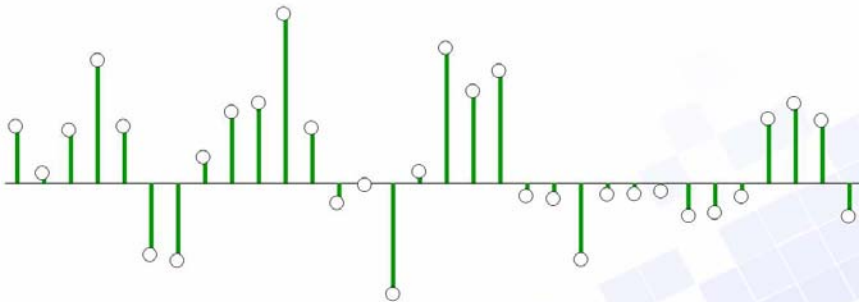


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Diskrete Darstellung

17

Kontinuierliche Signale werden abgetastet,



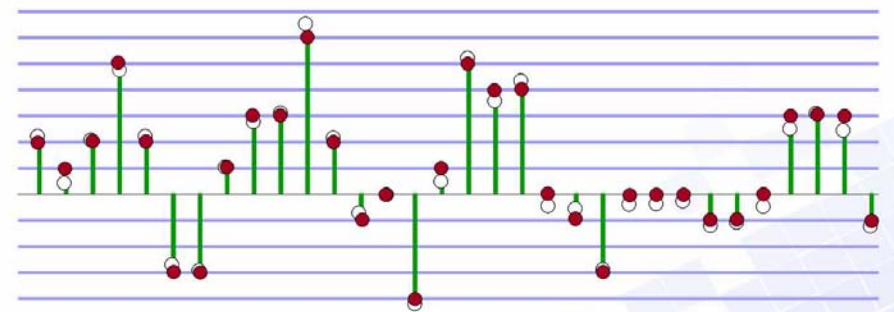
- Das Ergebnis ist ein **diskretes** Signal, das aus einer Folge von Abtastwerten besteht.

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Quantisierung

18

Die diskreten Abtastwerte werden **quantisiert**,



- ... und mit einer bestimmten Genauigkeit (z.B. 32bit) abgespeichert.

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Abtastung und Quantisierung

19

- Der Fehler, der durch die Quantisierung entsteht, wird als **Quantisierungsrauschen** bezeichnet. (Wir vernachlässigen diesen Fehler im Folgenden)
- **Frage:** Wie muß ich den Abstand der Abtastpunkte wählen (**Abtastfrequenz**) und welche Konsequenzen folgen daraus?

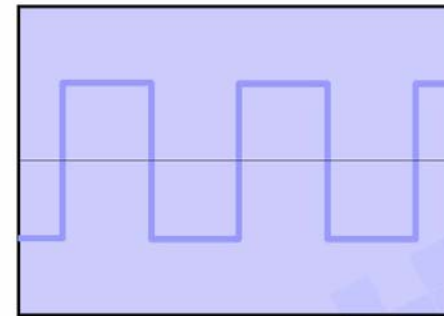


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

20

- Jedes kontinuierliche Signal $f(x)$ kann als Überlagerung von **harmonischen Schwingungen unterschiedlicher Frequenz** (Sinus und Cosinus) dargestellt werden.

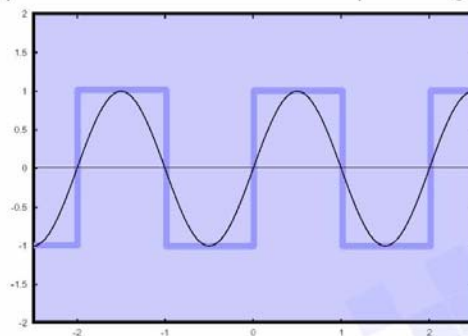


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

20

- Jedes kontinuierliche Signal $f(x)$ kann als Überlagerung von **harmonischen Schwingungen unterschiedlicher Frequenz** (Sinus und Cosinus) dargestellt werden.

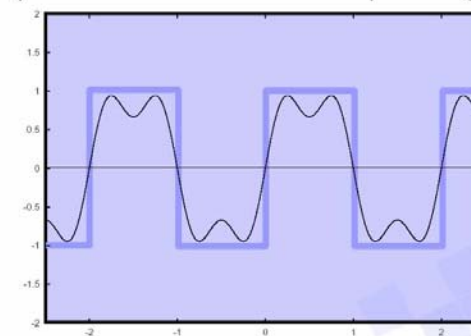


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

20

- Jedes kontinuierliche Signal $f(x)$ kann als Überlagerung von **harmonischen Schwingungen unterschiedlicher Frequenz** (Sinus und Cosinus) dargestellt werden.

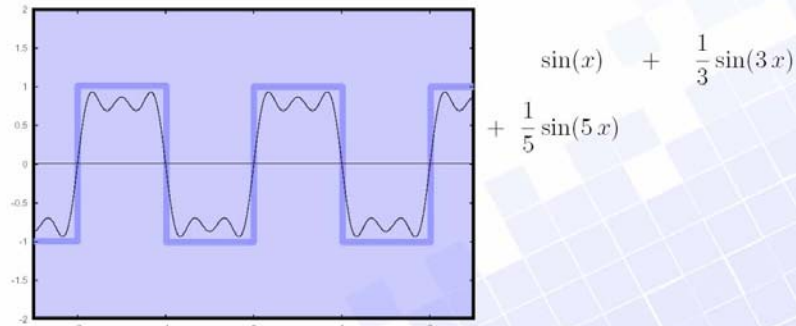


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

20

- Jedes kontinuierliche Signal $f(x)$ kann als Überlagerung von **harmonischen Schwingungen unterschiedlicher Frequenz** (Sinus und Cosinus) dargestellt werden.

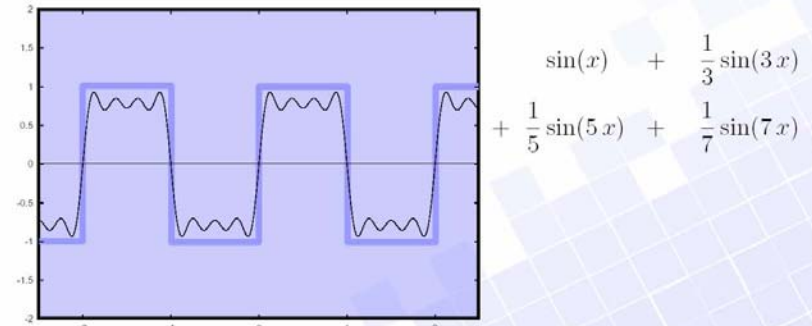


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

20

- Jedes kontinuierliche Signal $f(x)$ kann als Überlagerung von **harmonischen Schwingungen unterschiedlicher Frequenz** (Sinus und Cosinus) dargestellt werden.

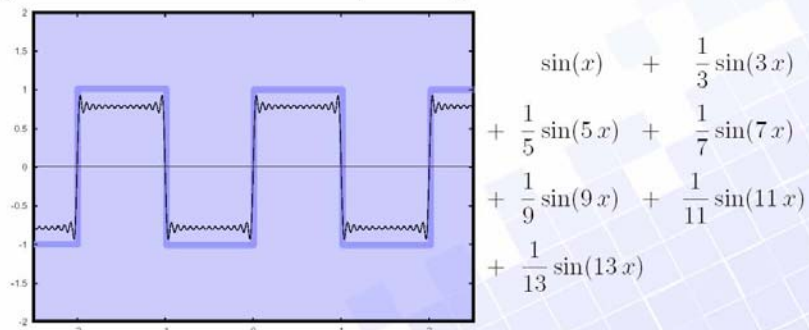


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

20

- Jedes kontinuierliche Signal $f(x)$ kann als Überlagerung von **harmonischen Schwingungen unterschiedlicher Frequenz** (Sinus und Cosinus) dargestellt werden.

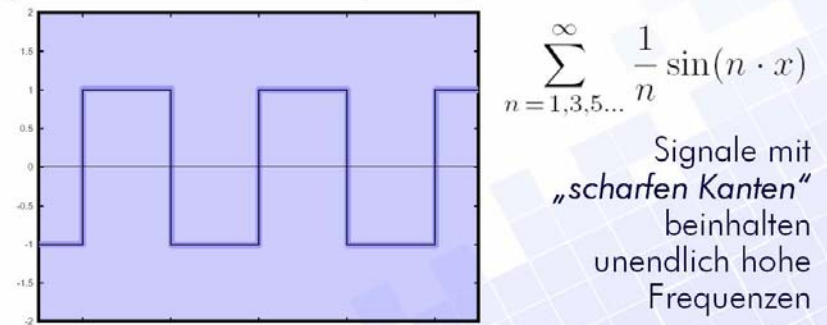


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

20

- Jedes kontinuierliche Signal $f(x)$ kann als Überlagerung von **harmonischen Schwingungen unterschiedlicher Frequenz** (Sinus und Cosinus) dargestellt werden.



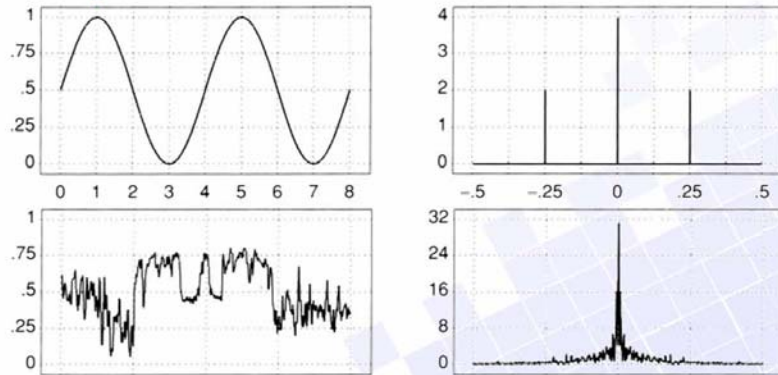
christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

21

● Jedes kontinuierliche Signal $f(x)$ hat ein eindeutiges **Frequenzspektrum** $F(t)$.

● **Beispiele:**

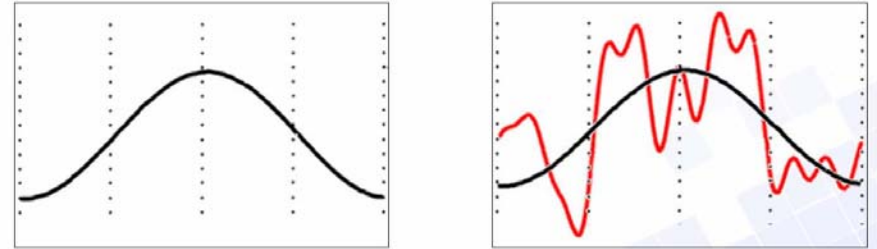


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

22

● Beispiel eines Signals mit beschränktem Frequenzspektrum:

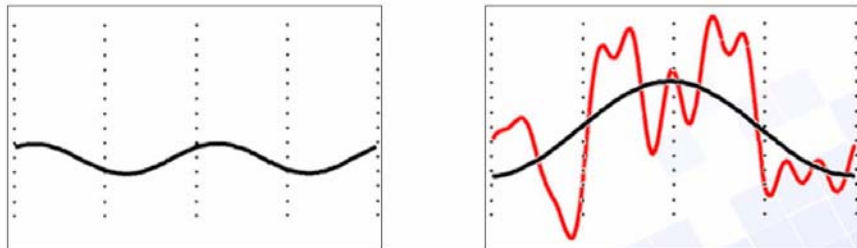


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

22

● Beispiel eines Signals mit beschränktem Frequenzspektrum:

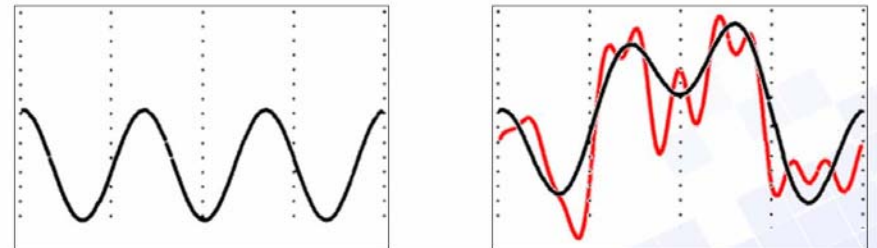


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

22

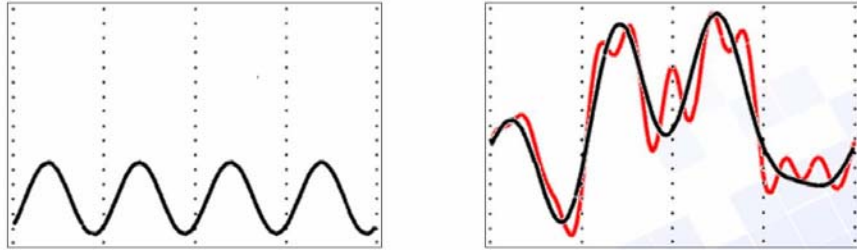
● Beispiel eines Signals mit beschränktem Frequenzspektrum:



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

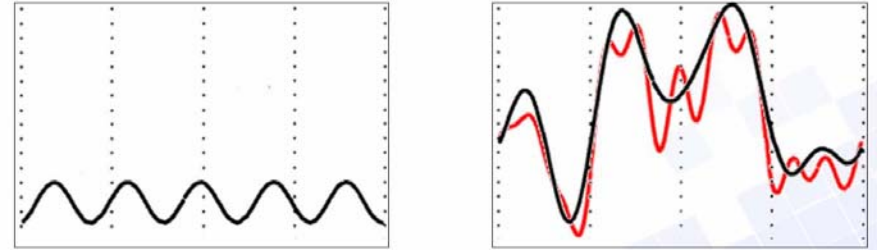
Frequenzspektrum

- Beispiel eines Signals mit beschränktem Frequenzspektrum:



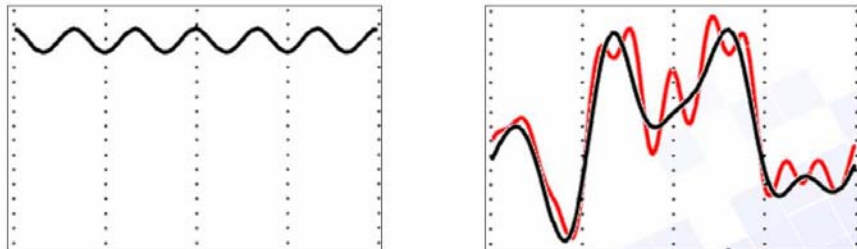
Frequenzspektrum

- Beispiel eines Signals mit beschränktem Frequenzspektrum:



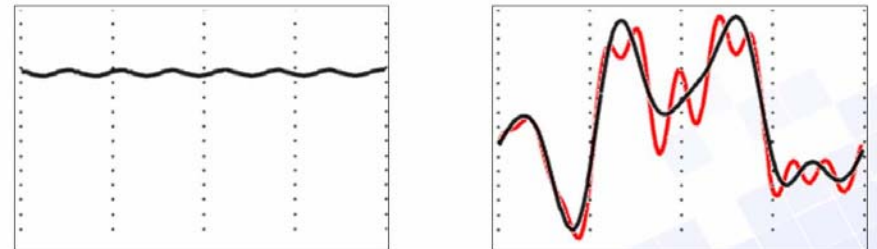
Frequenzspektrum

- Beispiel eines Signals mit beschränktem Frequenzspektrum:



Frequenzspektrum

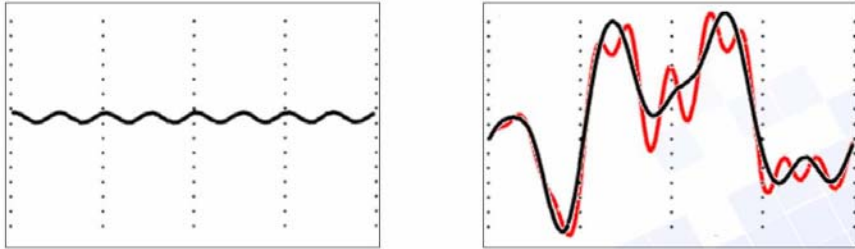
- Beispiel eines Signals mit beschränktem Frequenzspektrum:



Frequenzspektrum

22

- Beispiel eines Signals mit beschränktem Frequenzspektrum:

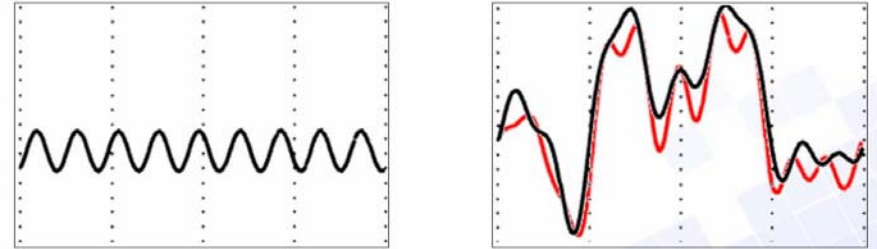


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

22

- Beispiel eines Signals mit beschränktem Frequenzspektrum:

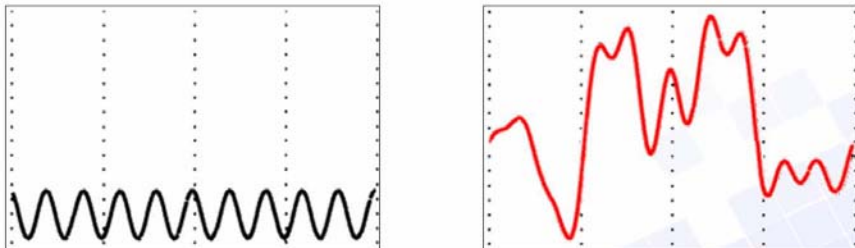


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Frequenzspektrum

22

- Beispiel eines Signals mit beschränktem Frequenzspektrum:



- Ein Signal heißt **bandbegrenzt**, wenn sein Frequenzspektrum auf ein endliches Intervall beschränkt ist.

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Abtasttheorem

23

- Ein kontinuierliches Signal $f(x)$, dessen Frequenzspektrum $F(t)$ bandbegrenzt ist, d.h.

$$F(t) = 0 \quad \text{für} \quad |t| > t_{max}$$

läßt sich **exakt rekonstruieren** aus einer diskreten Abtastung mit einer Schrittweite:

$$\tau < \frac{1}{2t_{max}}$$

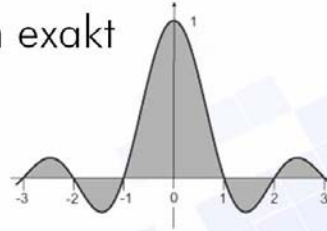
christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Rekonstruktion

24

- Ein hinreichend genau abgetastetes Signal kann aus den diskreten Abtastwerten mit Hilfe der sinc-Funktion exakt rekonstruiert werden.

$$\text{sinc}(x) = \frac{\sin(x)}{x}$$

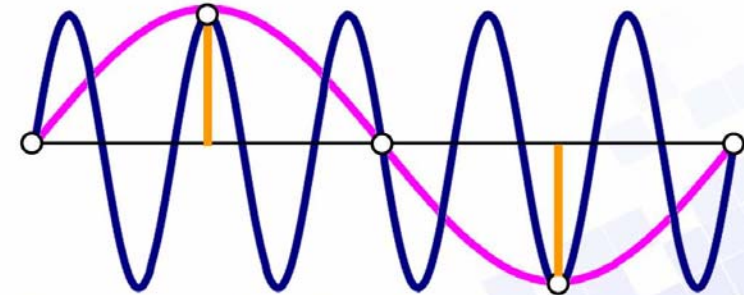


christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Abtasttheorem

25

Was passiert nun, wenn die Abtastfrequenz **zu gering** ist ?



- Es wird ein falsches Signal rekonstruiert!

Aliasing

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Literatur

26

Abtasttheorem nachzulesen in:

- J. Foley, A. van Dam, S. Feiner, and J. Hughes. *Computer Graphics, Principle And Practice*. Addison-Wesley, 1993.

Chapter 14.10.3 Sampling Theory

- oder in jedem Lehrbuch über Signalverarbeitung

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Differenzieren

27

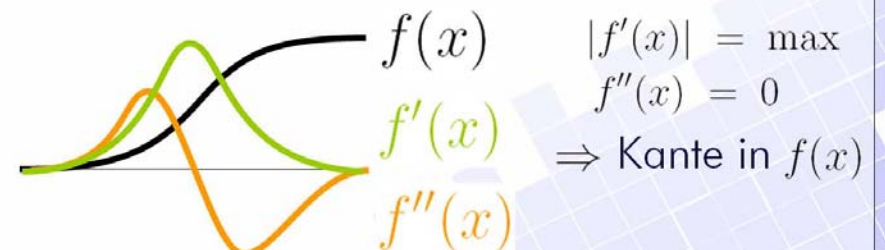
- Ableitungen im 1D: $f'(x) = \frac{df(x)}{dx}$

Eigenschaften der Ableitung:

$f'(x)$ beschreibt die Steigung der Kurve $f(x)$

Lokale Extrema: $f(x) = \max \Rightarrow f'(x) = 0$

- Kantendetektion:



christof rezk-salama, computergraphik und multimediasysteme, universität siegen

- 1. Ableitung im 2D:

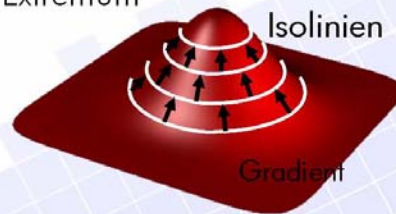
Der Gradient $\text{grad}f(x, y) = \begin{pmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{pmatrix}$

Eigenschaften des Gradientenvektors:

- zeigt in die Richtung des größten Anstiegs
- steht senkrecht auf den Isolinien
- ist gleich Null bei lokalem Extremum

Beispiel:

2D Höhenfeld



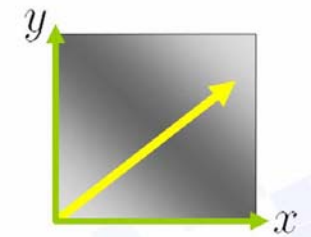
Partielle Ableitung im 2D:

- in x-Richtung: $\frac{\partial f(x, y)}{\partial x}$

- in y-Richtung: $\frac{\partial f(x, y)}{\partial y}$

- in beliebige Richtung $\vec{w} = \begin{pmatrix} w_x \\ w_y \end{pmatrix}$

$$\begin{aligned} \frac{\partial f(x, y)}{\partial \vec{w}} &= w_x \frac{\partial f(x, y)}{\partial x} + w_y \frac{\partial f(x, y)}{\partial y} \\ &= \langle \vec{w} \circ \text{grad}f(x, y) \rangle \end{aligned}$$



- Gradientenvektor:

$$\text{grad}f(x, y, z) = \begin{pmatrix} \frac{\partial f(x, y, z)}{\partial x} \\ \frac{\partial f(x, y, z)}{\partial y} \\ \frac{\partial f(x, y, z)}{\partial z} \end{pmatrix}$$

- Richtungsableitung in Richtung $\vec{w} = \begin{pmatrix} w_x \\ w_y \\ w_z \end{pmatrix}$:

$$\begin{aligned} \frac{\partial f(x, y, z)}{\partial \vec{w}} &= w_x \frac{\partial f(x, y, z)}{\partial x} + w_y \frac{\partial f(x, y, z)}{\partial y} + w_z \frac{\partial f(x, y, z)}{\partial z} \\ &= \langle \vec{w} \circ \text{grad}f(x, y, z) \rangle \end{aligned}$$

Wie kann ich Ableitungen berechnen auf diskreten Daten?

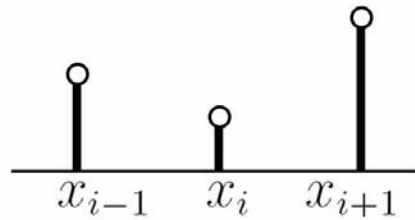
- Möglichkeit 1:** Approximiere die diskreten Daten (lokal) durch eine differenzierbare Funktion und differenziere diese.

- Möglichkeit 2:** Ersetze das Differential durch finite Differenzen

$$f'(x) = \frac{df(x)}{dx} \mapsto \frac{\Delta f(x)}{\Delta x}$$

Möglichkeit 1: Curve Fitting

32

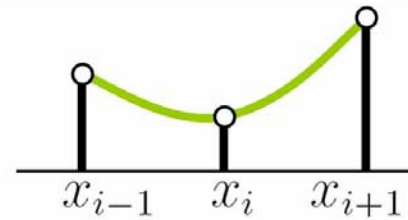


Bestimme die
Ableitung $f'(x_i)$
an der Stelle x_i

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Möglichkeit 1: Curve Fitting

32



Bestimme die
Ableitung $f'(x_i)$
an der Stelle x_i

1. Finde ein Polynom 2. Grades,

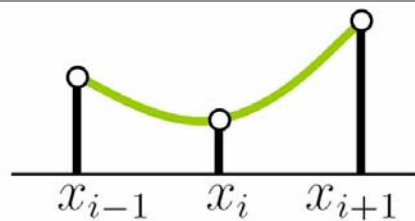
$$f(x) = ax^2 + bx + c$$

das die diskreten Werte lokal approximiert.

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Möglichkeit 1: Curve Fitting

32



Bestimme die
Ableitung $f'(x_i)$
an der Stelle x_i

1. Finde ein Polynom 2. Grades,

$$f(x) = ax^2 + bx + c$$

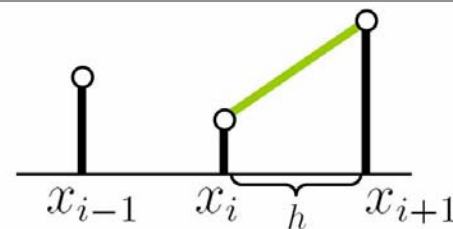
das die diskreten Werte lokal approximiert.

2. Differenziere dieses Polygon an der Stelle x_i

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Möglichkeit 2: Finite Differenzen

33



Bestimme die
Ableitung $f'(x_i)$
an der Stelle x_i

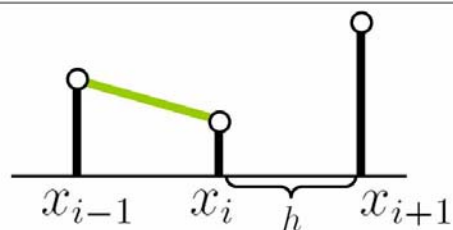
Vorwärtsdifferenzen:

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{h}$$

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Möglichkeit 2: Finite Differenzen

33



Bestimme die
Ableitung $f'(x_i)$
an der Stelle x_i

Vorwärtsdifferenzen:

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{h}$$

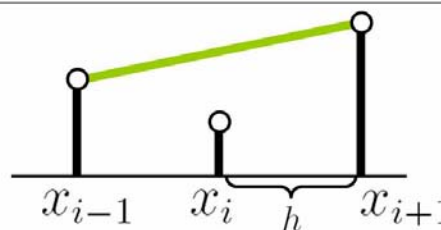
Rückwärtsdifferenzen:

$$f'(x_i) \approx \frac{f(x_i) - f(x_{i-1}))}{h}$$

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Möglichkeit 2: Finite Differenzen

34



Bestimme die
Ableitung $f'(x_i)$
an der Stelle x_i

Vorwärtsdifferenzen:

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{h}$$

Rückwärtsdifferenzen:

$$f'(x_i) \approx \frac{f(x_i) - f(x_{i-1}))}{h}$$

Zentrale Differenzen:

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_{i-1}))}{2h}$$

Fehlerordnung

$o(h)$

$o(h)$

$o(h^2)$

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Interpolation

35

In nichts zeigt sich der Mangel an
mathematischer Bildung mehr, als in
einer übertrieben genauen Rechnung

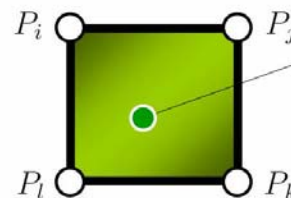
(Carl Friedrich Gauss, dt. Mathematiker, 1777-1855)

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Lokale Interpolation

36

- An den Datenpunkten P_i sind Datenwerte gegeben.



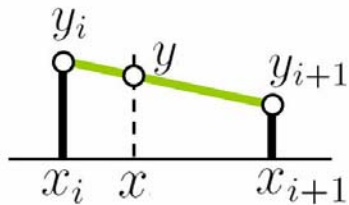
Wie bestimme ich einen
Datenwert für einen Punkt,
der innerhalb einer Zelle
liegt?

➔ **Interpolation**

d.h. approximative Rekonstruktion des Signals

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

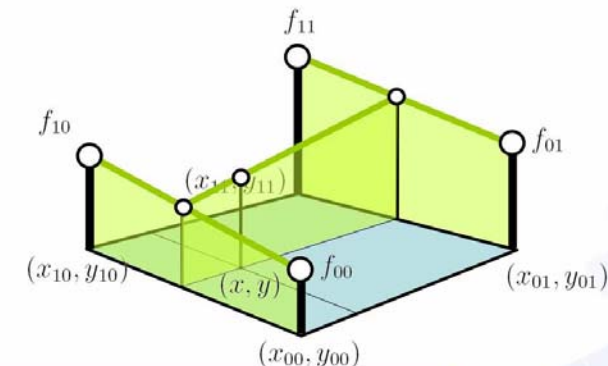
Lineare Interpolation (1D)



Lineares **Blenden**,
 $f(x) = \alpha y_{i+1} + (1 - \alpha)y_i$
 mit $\alpha = \frac{x - x_i}{x_{i+1} - x_i}$

$x = x_i \Rightarrow \alpha = 0 \Rightarrow f(x) = y_i$
 $x = x_{i+1} \Rightarrow \alpha = 1 \Rightarrow f(x) = y_{i+1}$

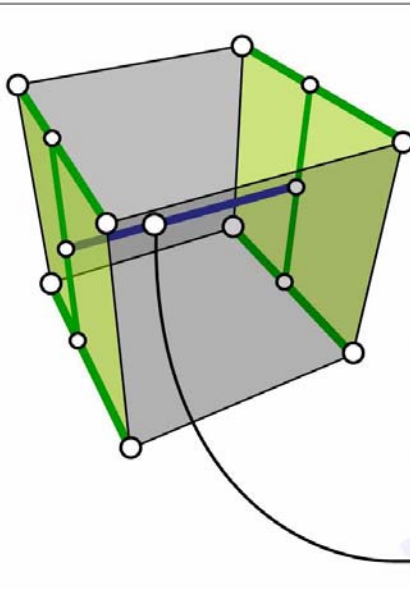
Bilineare Interpolation (2D)



$f_0 = \alpha f_{10} + (1 - \alpha)f_{00}$
 $f_1 = \alpha f_{11} + (1 - \alpha)f_{01}$
 $f(x, y) = \beta f_1 + (1 - \beta)f_0$

$\alpha = \frac{x - x_{00}}{x_{10} - x_{00}}$
 $\beta = \frac{y - y_{00}}{y_{01} - y_{00}}$

Trilineare Interpolation (3D)

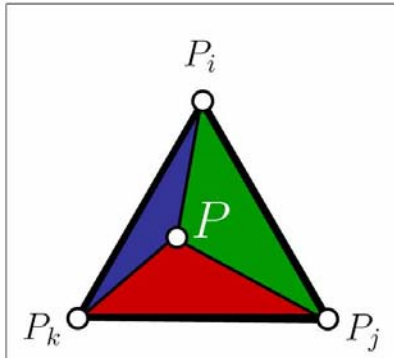


4 lineare Interpolationen
 $f_{00} = \alpha f_{001} + (1 - \alpha)f_{000}$
 $f_{01} = \alpha f_{011} + (1 - \alpha)f_{010}$
 $f_{10} = \alpha f_{101} + (1 - \alpha)f_{100}$
 $f_{11} = \alpha f_{111} + (1 - \alpha)f_{110}$

2 bilineare Interpolationen
 $f_0 = \beta f_{01} + (1 - \beta)f_{00}$
 $f_1 = \beta f_{11} + (1 - \beta)f_{10}$

1 trilineare Interpolation
 $f = \gamma f_1 + (1 - \gamma)f_0$

Dreieckige Zellen (2D)



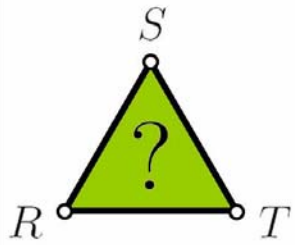
Der Punkt P kann ausgedrückt werden als
 $P = a_i P_i + a_j P_j + a_k P_k$
 mit $a_i + a_j + a_k = 1$

Schwerpunktskoordinaten
 (Baryzentrische Koordinaten)
 $a_i = a_j = a_k = \frac{1}{3}$
 $\Rightarrow P$ ist Schwerpunkt

$A = Area(\Delta P_i P_j P_k)$
 $a_i = \frac{1}{A} Area(\Delta P P_j P_k)$
 $a_j = \frac{1}{A} Area(\Delta P_i P P_k)$
 $a_k = \frac{1}{A} Area(\Delta P_i P_j P)$

Dreieckige Zellen (2D)

41



Wie bestimme ich den Flächeninhalt eines Dreiecks $Area(\triangle RST)$?

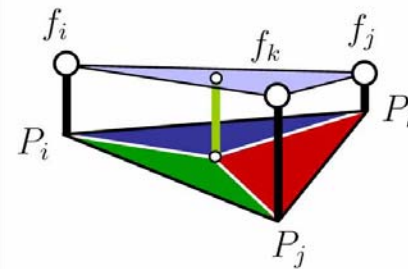
$$R = \begin{pmatrix} R_x \\ R_y \end{pmatrix} \quad S = \begin{pmatrix} S_x \\ S_y \end{pmatrix} \quad T = \begin{pmatrix} T_x \\ T_y \end{pmatrix}$$

$$\begin{vmatrix} 1 & 1 & 1 \\ R_x & S_x & T_x \\ R_y & S_y & T_y \end{vmatrix} = \pm 2 \cdot Area(\triangle RST)$$

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Lineare Interpolation im Dreieck

42



Bestimme die baryzentrischen Koordinaten des Punkts P
 $P = a_i P_i + a_j P_j + a_k P_k$

Bestimme den Funktionswert mit den gleichen Gewichten
 $f = a_i f_i + a_j f_j + a_k f_k$

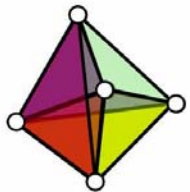
Anmerkung: Auch Punkte P außerhalb des Dreiecks lassen sich in baryzentrischen Koordinaten ausdrücken. Die Gewichte sind dann nicht mehr alle positiv. Es gilt aber nach wie vor $a_i + a_j + a_k = 1$

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Tetraeder

43

Baryzentrische Koordinaten im Tetraeder



1. Verbinde den inneren Punkt mit allen Randpunkten.

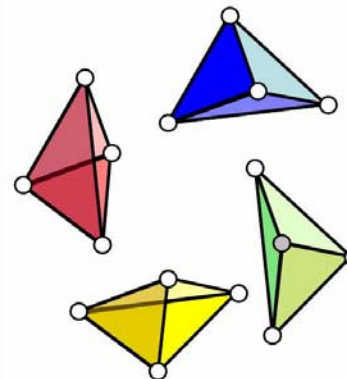
Es entstehen 4 Tetraeder als Unterteilung

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Tetraeder

43

Baryzentrische Koordinaten im Tetraeder



1. Verbinde den inneren Punkt mit allen Randpunkten.

Es entstehen 4 Tetraeder als Unterteilung

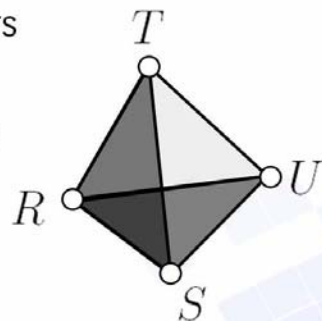
2. Das Volumen eines Teil-Tetraeders ist das baryzentrische Gewicht für den gegenüberliegenden Punkt

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Tetraeder

44

- Volumen eines Tetraeders
 $Vol(Tet(R, S, T, U))$ mit
 Vertices R, S, T und U :



$$\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & U_x \\ R_y & S_y & T_y & U_y \\ R_z & S_z & T_z & U_z \end{vmatrix} = \pm 6 \cdot Vol(Tet(R, S, T, U))$$

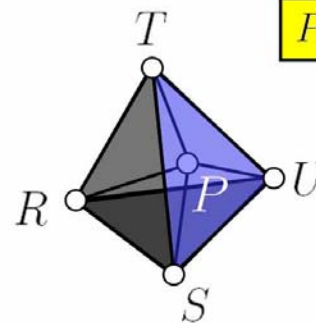
christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Tetraeder

45

Baryzentrische Koordinaten im Tetraeder

$$P = a_R R + a_S S + a_T T + a_U U$$



$$a_R = \frac{\begin{vmatrix} 1 & 1 & 1 & 1 \\ P_x & S_x & T_x & U_x \\ P_y & S_y & T_y & U_y \\ P_z & S_z & T_z & U_z \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & U_x \\ R_y & S_y & T_y & U_y \\ R_z & S_z & T_z & U_z \end{vmatrix}}$$

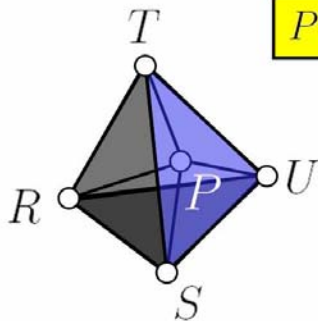
christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Tetraeder

45

Baryzentrische Koordinaten im Tetraeder

$$P = a_R R + a_S S + a_T T + a_U U$$



$$a_S = \frac{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & P_x & T_x & U_x \\ R_y & P_y & T_y & U_y \\ R_z & P_z & T_z & U_z \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & U_x \\ R_y & S_y & T_y & U_y \\ R_z & S_z & T_z & U_z \end{vmatrix}}$$

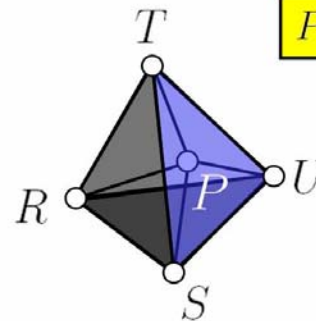
christof rezk-salama, computergraphik und multimediasysteme, universität siegen

Tetraeder

45

Baryzentrische Koordinaten im Tetraeder

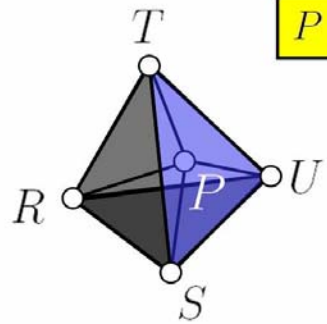
$$P = a_R R + a_S S + a_T T + a_U U$$



$$a_T = \frac{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & P_x & U_x \\ R_y & S_y & P_y & U_y \\ R_z & S_z & P_z & U_z \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & U_x \\ R_y & S_y & T_y & U_y \\ R_z & S_z & T_z & U_z \end{vmatrix}}$$

christof rezk-salama, computergraphik und multimediasysteme, universität siegen

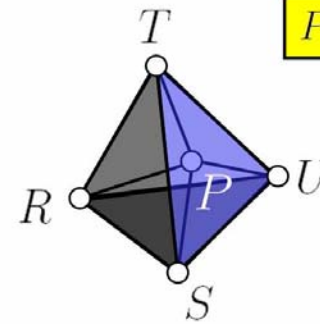
Baryzentrische Koordinaten im Tetraeder



$$P = a_R R + a_S S + a_T T + a_U U$$

$$a_U = \frac{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & P_x \\ R_y & S_y & T_y & P_y \\ R_z & S_z & T_z & P_z \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & U_x \\ R_y & S_y & T_y & U_y \\ R_z & S_z & T_z & U_z \end{vmatrix}}$$

Baryzentrische Koordinaten im Tetraeder



$$P = a_R R + a_S S + a_T T + a_U U$$

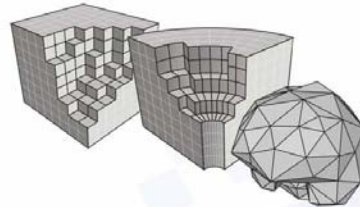
$$a_U = \frac{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & P_x \\ R_y & S_y & T_y & P_y \\ R_z & S_z & T_z & P_z \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 \\ R_x & S_x & T_x & U_x \\ R_y & S_y & T_y & U_y \\ R_z & S_z & T_z & U_z \end{vmatrix}}$$

Interpolation:

$$f = a_R f_R + a_S f_S + a_T f_T + a_U f_U$$

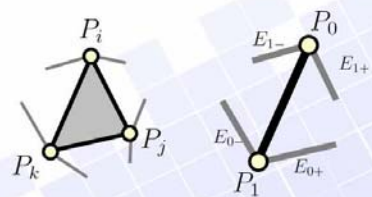
Gitterstrukturen in 2D und 3D

- uniforme Gitter
- rectilineare Gitter
- curvilineare Gitter
- unstrukturierte Gitter



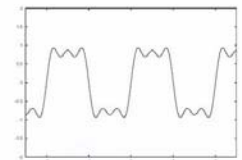
Datenstrukturen für Dreiecksnetze

- „Shared Vertex“
- „Winged Edge“



Abtasttheorem

- Kontinuierliche Signale lassen sich in harmonische Schwingungen zerlegen (Frequenzspektrum)
- Bandbegrenzte Signale lassen sich aus diskreten Abtastwerten exakt rekonstruieren
- Exakte Rekonstruktion erfolgt mit dem Sinc-Filter
- Unterabtastung führt zu **Aliasing**-Effekten



● Differenzieren auf Gittern

- Gradientenvektor
 - zeigt in Richtung des größten Anstiegs
 - steht senkrecht auf den Isolinien (2D) bzw Isoflächen (3D)
- Differenzieren auf Gittern
 - Curve/Surface Fitting
 - Finite Differenzen
 - Vorwärts, Rückwärts,
 - Zentrale Differenzen



● Lokale Interpolation

- Stückweise lineare Interpolation
 - Kartesisch: Linear, bilinear, trilinear
 - Baryzentrisch: Dreieck und Tetraeder