

Kapitel 1

De Casteljau Algorithmus und Bézierkurven.

19.10.2010

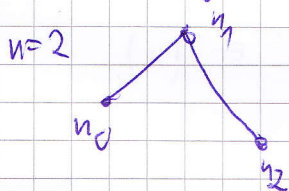
Vorlesung 1

Ziel: Beschreibung glatter Kurven

- Einfache Modellierbarkeit d.h. einfache Veränderung / Anpassung der geometrischen Form der Kurve

Für einfache Modellierung sind so z.B. $\sin/\cos$ und Fouriersdarstellungen ungeeignet

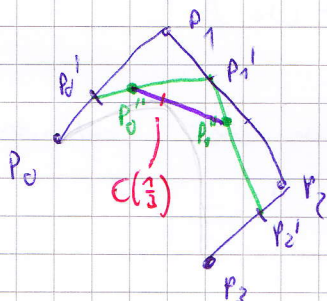
De Casteljau (1955)



Kontrollpolygon bestehend aus $n+1$ Punkten $\{p_0, \dots, p_n\} \subset \mathbb{R}^2$ (oder höheren Dimensionen)

Das Kontrollpolygon bestimmt eine Kurve $c: [0,1] \rightarrow \mathbb{R}^2$ ($\mathbb{R}^d$)

Sei $t = \frac{1}{3}$. Für jede Kante des Kontrollpolygons fängt man in der nächsten Stufe jeweils nach einem Drittel der Kante einen neuen Punkt p_i' an



So kann man das für jeder $t \in [0,1]$ machen. $c(0)$ ist insgesamt p_0 und $c(1)$ ist in dem Beispiel p_2 .

Insgesamt erhält man ein Polynom in einer Variablen in $\mathbb{R}^2$.

Algorithmus

$\{p_0, \dots, p_n\} \subset \mathbb{R}^d$ sei eine Menge von Kontrollpunkten in $\mathbb{R}^d$. Der de Casteljau Algorithmus definiert nun ~~beachtet~~ eine Kurve $c: [0,1] \rightarrow \mathbb{R}^d$, in dem für jeden Wert $t \in [0,1]$ folgende Rekursion ausgeführt wird.

$$p_i^0(t) = p_i \quad \text{mit } i = \{0, \dots, n\}$$

$$p_i^r(t) = (1-t) p_i^{r-1}(t) + t \cdot p_{i+1}^{r-1}(t) \quad \text{mit } i = \{0, n-r\} \text{ und } r = \{1, \dots, n\}$$

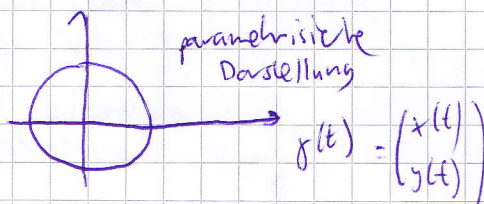
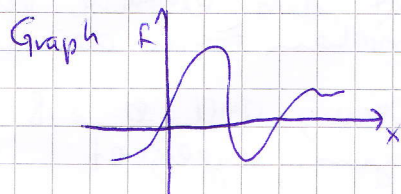
In Stufe $r=n$ gibt es nur noch den Punkt $p_0^n(t)$. Dieser Punkt ist als $c(t)$ definiert.

$$c(t) = p_0^n(t)$$

Funktionale Darstellung der Kurven werden wir herleiten, was dann die Bezierkurven sind.

Vorlesung 2
21.10.2010

Funktion $f: \mathbb{R} \rightarrow \mathbb{R}$



Einfache Funktionen sind Polynome

$$f(x) = a_0 + a_1 x + a_2 x^2 + \dots = \sum_{i=0}^n a_i x^i \in \mathcal{P}(x)$$

$$\gamma = \begin{pmatrix} x \in \mathcal{P}(t) \\ y \in \mathcal{P}(t) \end{pmatrix}$$

$$\gamma(t) = (1-t)A + tB$$

$$= \begin{pmatrix} (1-t)a_1 + tb_1 \\ (1-t)a_2 + tb_2 \end{pmatrix}$$

Was sind die geometrischen Eigenschaften der Koeffizienten?

a_0 = Schnitt durch y-Achse

a_1 = Anstieg in $x=0$

a_i = höhere Ableitungen vom Polynom an der Stelle $x=0$

Mit dieser klassischen Polynomdarstellung kann man allerdings schwer modellieren.

Bernstein Polynome

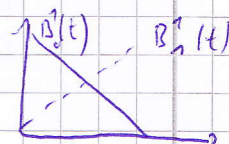
Definition: Zum Grad n gibt es $n+1$ Bernstein-Polynome in der Form

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad \text{mit } i = 0, \dots, n \quad \text{und } t \in [0, 1]$$

Für $n=0$ $B_0^0(t) = 1$

$n=1$ $B_0^1(t) = 1-t$, $B_1^1(t) = t$

$n=2$ $B_0^2(t) = (1-t)^2$, $B_1^2(t) = 2t(1-t)$, $B_2^2(t) = t^2$



Erste Eigenschaften

$$B_0^n(t) = \binom{n}{0} t^0 \cdot (1-t)^{n-0} = (1-t)^n$$

$$B_n^n(t) = \binom{n}{n} t^n \cdot (1-t)^{n-n} = t^n$$

$$B_i^n(0) = \binom{n}{i} 0^i \cdot (1-0)^{n-i} = 0$$

$$B_i^n(1) = \binom{n}{i} 1^i \cdot (1-1)^{n-i} = 0$$

für $0 < i < n$

Lemma: 1) Die Bernsteinpolynome erfüllen folgende Rekursionsformel:

$$B_i^n(t) = \binom{n-1}{i-1} t \cdot B_{i-1}^{n-1}(t) + \binom{n-1}{i} (1-t) \cdot B_i^{n-1}(t)$$

$$B_0^n(t) = 1, \quad B_i^n(t) > 0 \quad \forall i < n \quad \text{oder} \quad i > n$$

2) Symmetrie. B_i^n ist symmetrisch zu B_{n-i}^n bezüglich

Spiegelung der $t = \frac{1}{2}$ Achse.

$$\text{Formal: } B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} = \binom{n}{n-i} (1-t)^{n-i} \cdot t^{n-(n-i)} = B_{n-i}^n(1-t)$$

3) Partition der Einheit

$$\forall t \in [0; 1] \quad \sum_{i=0}^n B_i^n(t) = \sum_{i=0}^n \binom{n}{i} t^i (1-t)^{n-i} = (t + (1-t))^n = 1$$

$$4) \text{ Ableitung } \frac{d}{dt} B_i^n(t) = n \cdot (B_{i-1}^{n-1}(t) - B_i^{n-1}(t))$$

Beweis 1)

$$\binom{n}{i} = \frac{n!}{i!(n-i)!} = \frac{n \cdot (n-1)!}{i!(n-i)!} = \dots = \binom{n-1}{i-1} + \binom{n-1}{i}$$

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

$$= \binom{n-1}{i-1} t^i (1-t)^{n-i} + \binom{n-1}{i} t^i (1-t)^{n-i}$$

$$= \dots$$

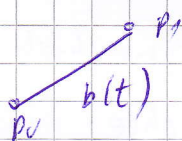
Für festes n bilden Bernsteinpolynome eine Basis des Polynomraums $\mathcal{P}^n(t)$, d.h.

Definition: Seien $\{p_0, \dots, p_n\} \in \mathbb{R}^2 (\mathbb{R}^d)$ eine Menge von Kontrollpunkten, die auch zu einem Kontrollpolygon verbunden werden können. Dann ist die Bézierkurve $b(t)$ vom Grad n definiert durch

$$b(t) = \sum_{i=0}^n p_i \cdot B_i^n(t)$$

Beispiel: p_0, p_1 $n=1$

$$b(t) = p_0 \cdot B_0^1(t) + p_1 \cdot B_1^1(t) = p_0(1-t) + p_1 t$$



Lemma 1) $b(t)$ interpoliert die Endpunkte p_0 und p_n .

2) Ableitungen an $t=0$ und $t=1$

$$b'(t=0) = n(p_1 - p_0)$$

$$b'(t=1) = n(p_n - p_{n-1})$$

Damit ist die Kurve in $t=0$ tangential zu der ersten Kante des Kontrollpolygons und für $t=1$ entsprechend an der letzten Kante.

zur 4) Eigenschaft von $B_i^n(t)$

$$\frac{d}{dt} B_i^n(t) = \frac{d}{dt} \binom{n}{i} t^i (1-t)^{n-i}$$

$$= \binom{n}{i} \left[i t^{i-1} (1-t)^{n-i} + (n-i) t^i (1-t)^{n-i-1} (-1) \right]$$

$$\binom{n}{i} = n \binom{n-1}{i-1} \quad \binom{n}{i} (n-i) = n \binom{n-1}{i}$$

$$= n \left(\underbrace{\binom{n-1}{i-1} t^{i-1} (1-t)^{n-i}}_{B_{i-1}^{n-1}(t)} - \underbrace{\binom{n-1}{i} t^i (1-t)^{n-i}}_{B_i^{n-1}(t)} \right)$$

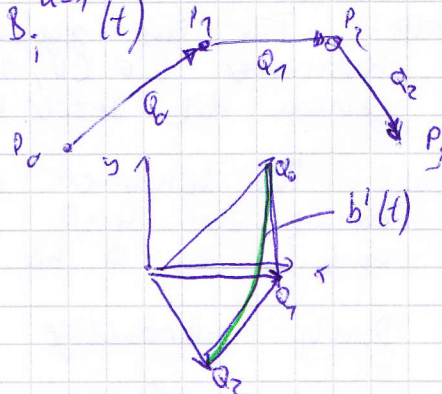
weitere Eigenschaften der Bezierkurven

1a) $b(0) = p_0$, $b(1) = p_n$

1b) $\frac{d}{dt} b(t) = \frac{d}{dt} \left(\sum_{i=0}^n B_i^n(t) p_i \right) = n \cdot \sum_{i=0}^{n-1} \left(B_{i+1}^{n-1}(t) - B_i^{n-1}(t) \right) q_i$

umsoheren $\odot = n \cdot \sum_{i=0}^{n-1} (p_{i+1} - p_i) B_i^{n-1}(t)$

Die Ableitung kann formal auch als Bezierkurve interpretiert werden der Punkte $Q_i = p_{i+1} - p_i$, die das Kontrollpolygon für $b'(t)$ bilden.



Definition: Seien $\Omega \in \mathbb{R}^2$, dann heißt Ω konvex, genau dann wenn für jede zwei Punkte $p, q \in \Omega$ ~~liegt auf~~ das Liniensegment $\overline{pq}$ komplett in Ω liegt.

Definition: Seien $P = \{p_0, \dots, p_n\} \in \mathbb{R}^2$. Dann ist die konvexe Hülle von P die kleinste konvexe Menge, die P enthält. Anders gesagt: Die Menge der Punkte aller Segmente $\overline{pq}$ mit $p, q \in P$.

Lemma: Die konvexe Hülle von P besteht genau aus allen Linearkombinationen $q \in \mathbb{R}^2$ mit

$$q = \sum_{i=0}^n \lambda_i p_i \quad \sum_{i=0}^n \lambda_i = 1 \quad \text{und} \quad \lambda_i \geq 0$$

so ein q heißt auch konvexkombination der p_i .

2) Die Bezierkurve liegt in der konvexen Hülle des Kontrollpolygons.

Beweis: $\forall t \in [0; 1]$ ist $b(t) = \sum_{i=0}^n B_i^n(t) p_i$ eine Konvexkombination der Kontrollpunkte $\{p_0, \dots, p_n\}$ und nach Lemma liegt $b(t)$ in der konvexen Hülle.

3) "Affine Invarianz": Wenn ich das Kontrollpolygon skaliere, drehe oder verschiebe, dann skaliert, dreht oder verschiebt sich die Bezierkurve genauso.

Sei A eine 2×2 -Matrix und v ein Verschiebvektor.

$q_i = A p_i + v$ die transformierten Punkte

$$\sum_{i=0}^n B_i^n(t) \cdot (A p_i + v) = A \cdot \underbrace{\sum_{i=0}^n B_i^n(t) p_i}_{b(t)} + \underbrace{\sum_{i=0}^n B_i^n(t) \cdot v}_{=1} = A \cdot b(t) + v$$

Bezug von de Casteljau und Bezierkurven

Satz: Für ein gegebenes Kontrollpolygon $\{p_0, \dots, p_n\}$ und einem Parameter $t \in [0; 1]$ berechnet der de Casteljau-Algorithmus genau die Bezierkurve $b(t)$.

Beweis: Wir benennen die in de Casteljau eingeführten Punkte zur Stufe r mit $c_i^r(t)$, wobei $i = 0, \dots, n-r$ läuft und r von 0 bis n . $c_i^0(t) = p_i$.

Durch vollständige Induktion zeigen wir, dass

$$c_i^r(t) = \sum_{j=0}^r p_{i+j} B_j^r(t).$$

Behauptung gilt für $r=0$: $c_i^0(t) = p_i = \sum_{j=0}^0 p_{i+j} B_j^0(t) = p_i \cdot \underbrace{1}_{\sum_{j=0}^0 B_j^0(t)} = p_i \quad \checkmark$

Die Annahme gelte für $r-1$, nun soll sie auch für r gelten:

$$c_i^r(t) = (1-t) c_i^{r-1} + t c_{i+1}^{r-1} = (1-t) \left(\sum_{j=0}^{r-1} p_{i+j} B_j^{r-1}(t) \right)$$

$$+ t \left(\sum_{j=0}^{r-1} p_{i+1+j} B_j^{r-1}(t) \right) \\ = (1-t) \sum_{j=0}^{r-1} p_{i+j} B_j^{r-1}(t) + t \sum_{j=i+1}^{r+1} p_j B_{j-i-1}^{r-1}(t) \\ = (1-t) \sum_{j=i}^{r-1} p_j B_{j-i}^{r-1}(t) + t \sum_{j=i}^{r-1} p_j B_{j-i-1}^{r-1}(t)$$

da die hinzugekommenen Summanden 0 sind.

$$\begin{aligned}
&= \sum_{j=1}^{r-1} P_j \left((1-t) B_{j-1}^{r-1}(t) + t B_{j-1}^{r-1}(t) \right) \\
&= \sum_{j=1}^r P_{j-1} B_{j-1}^r(t) = \sum_{j=0}^r P_{i+j} B_j^r(t) \quad \square
\end{aligned}$$

Bemerkung: Zum Modellieren von ^{kompakten} Kurven ist ein einzelner Bezierpolygon ungeeignet, da die geometrische Bedeutung der inneren Kontrollpunkte nicht offensichtlich ist. Daher macht man stückchenweise Bezierkurven ($\rightarrow$ Spline, B-Spline)

Definition: B-Spline $s: [u_i, u_r] \rightarrow \mathbb{R}^2$ ist eine stetige Kurve, die auf jedem Segment $[u_i, u_{i+1}]$ mit Knoten $u_i \leq u_{i+1} \leq \dots \leq u_r$ ein Polynom 1. Grades ist, d.h. jedes Segment $[u_i, u_{i+1}]$ lässt sich als Bezierkurve darstellen.

Für die Verbindung zu Bezierkurven ^{reparametrisieren} ~~repräsentieren~~ wird jedes Intervall $[u_i, u_{i+1}]$ auf $[0, 1]$ d.h.

$$t(u) = \frac{u - u_i}{u_{i+1} - u_i}$$

Bezierdarstellung $s(t(u)) := s(u) \quad u \in [u_i, u_{i+1}]$

Bedingung für Stetigkeit des Tangentialvektors an den Verbindungsstellen $s(u_i)$:

$$\begin{aligned}
\frac{d}{du} s_{i-1}(t(u)) &= \frac{d}{dt} s_{i-1}(t) \cdot \frac{1}{u_i - u_{i-1}} = \frac{d}{dt} s(t) \cdot \frac{1}{u_{i+1} - u_i} \\
&= \frac{d}{du} s_i(t(u))
\end{aligned}$$

Tangentenvektor des letzten Bezierpolygons Tangentenvektor von Bezierpolygon

Bedingung:

1. Tangentenvektoren der anliegenden Bezierkurven müssen parallel sein.
2. Länge der Tangentenvektoren müssen im Verhältnis 1

Geometrie

Visualisierung

28.10.2010

Vorlesung 4

Wir wollen uns Datenstrukturen anschauen.

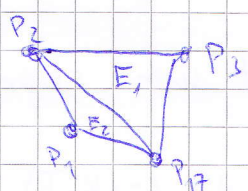
~~Flächenelement~~ ^{Punkt} ~~Werte~~ mit $f: (x, y, z) \in \mathbb{R}^3$. So ein Array $d \times n$ würde vollständig ein Polygon beschreiben.

Flächendarstellungen

$P_0 \quad x_0 \quad y_0 \quad z_0$
 $P_1 \quad x_1 \quad y_1 \quad z_1$
 $\vdots$
 $P_n \quad x_n \quad y_n \quad z_n$

Punkt liste.

Das reicht aber noch nicht. Wir brauchen noch eine Elementliste in der die Indizes der Punkte enthalten sind, die zu dem Flächenelement gehören.



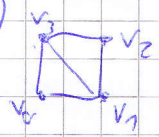
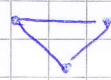
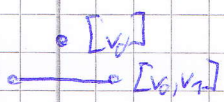
$E_1 = \{2, 3, 17\}$
 $E_2 = \{1, 2, 17\}$

Die Reihenfolge bestimmt die Orientierung.

WRL / VRML indexed face set

Definition: Sei $\{v_0, \dots, v_d\}$ eine Menge von d abstrakten Punkten. Eine geordnete Gruppe von d Punkten $[v_0, \dots, v_d]$ heißt abstrakter d -Simplex.

z.B. $d=0$ $[v_0]$ 0-Simplex (interpretieren wir als Punkt)
 $d=1$ $[v_0, v_1]$ 1-Simplex (Kante)
 $d=2$ $[v_0, v_1, v_2]$ 2-Simplex (Dreieck)
 $d=3$ $[v_0, v_1, v_2, v_3]$ 3-Simplex (Tetraeder)



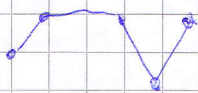
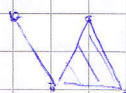
$[v_0, \dots, v_d]$ d -Simplex. Jede Teilmenge $f = [v_{i_1}, \dots, v_{i_k}]$ aus k Punkten von σ ist wieder ein Simplex und heißt Seite von σ .

z.B. Betrachte ein Dreieck $\sigma = [v_0, v_1, v_2]$ aus 8 Seiten:
 $[v_0], [v_1], [v_2], [v_0, v_1], [v_1, v_2], [v_2, v_0], [v_0, v_1, v_2], \emptyset$

2.11.2016 Definition (abstrakter Simplicialkomplex):

Vorlesung 5

Menge von abstrakten Punkten $V = \{v_0, v_1, \dots\}$. Sei S eine Menge von Simplizes, die aus Punkten von V gebildet werden. Von S fordern wir, wenn $\sigma \in S$, dann sind auch alle Seiten von σ in S enthalten. In diesem Fall heißt $K = (V, S)$ abstrakter Simplicialkomplex.



Aus S und allen Seiten erhält man einen Simplicialkomplex.

Definition: Sei $K = (V, S)$ und $L = (V, T)$ zwei Simplicialkomplexe über derselben Punktmenge V . Dann heißt L ein Unterkomplex von K , wenn $T \subset S$ ist.

Definition: Sei T eine Menge von Simplizes über Punktmenge V . Dann besteht der Abschluss von T

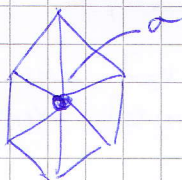
$$\text{Closure}(T) = \{ \sigma \in T \text{ und alle Seiten von } \sigma \}$$

$\text{Closure}(T)$ ist ein Simplicialkomplex.

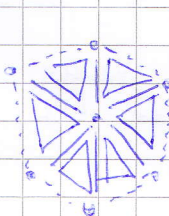
Definition: Sei K ein Simplicialkomplex und $\sigma \in K$ ein Simplex.

$$\text{star } \sigma = \{ \eta \in K, \text{ die } \sigma \text{ enthalten und die Seiten von } \eta \}$$

Beispiel:



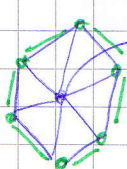
1. —
d.h. nek,
die σ ent-
halten



2. ---
d.h. Seiten
von η

Definition: Sei $\sigma \in K$ ein Simplex, dann ist

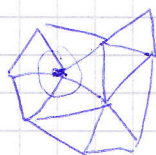
$$\text{link } \sigma = \{ \eta \in \text{star } \sigma \mid \eta \cap \sigma = \emptyset \}$$



link σ

Definition: Eine simpliziale Fläche F ist ein Simplicialkomplex $K = (V, S)$, wobei

1. Jeder Punkt von V in mindestens einem Simplex enthalten ist,
2. S aus einer Menge von Dreiecken und ihren Seiten besteht
3. Der Link $[v]$ zu jedem Punkt ist ein (asthetik) Kreis



← keine simpl. Fläche

Definition: Sei F eine simpliziale Fläche und v die Anzahl der 0-Simplizes (Eckpunkte), e die Anzahl der 1-Simplizes (Kanten) und f die Anzahl der 2-Simplizes (Dreiecke).
Dann ist die Euler-Charakteristik

$$\chi(F) = v - e + f$$

Beispiel $\chi(\text{Tetraeder}) = \chi(\text{Kugel}) = 4 - 6 + 4 = 2$

$$\chi(\text{Torus}) = 0, \quad \chi(\text{Kugel}) = 2$$

Die Euler-Charakteristik ist eine topologische Invariante und hängt nur von der Topologie der Fläche ab.

Wenn F und G zwei simpliziale Flächen, dann folgt aus

$$\chi(F) = \chi(G) \text{ nicht notwendigerweise, dass } F \text{ und } G$$

als topologische Flächen gleich sind.

Lemma: Sei F eine simpliziale Fläche (ohne Rand), dann gilt, dass $3f = 2e$

Beweis: Wir zählen die Anzahl der Kanten über die Anzahl der Dreiecke. Für jedes Dreieck hat man 3 Kanten. Die werden aber jeweils doppelt gezählt, das ergibt

$$e = \frac{3f}{2}$$

□

Lemma: Sei F eine simpliziale Fläche (ohne Rand) und sei die Anzahl der Dreiecke sehr groß im Verhältnis zur $\chi(F)$. Dann gilt, dass

$$f \sim 2v \quad \text{bzw.} \quad e \sim 3v$$

Beweis: $\chi(F) = v - e + f = v - \frac{3f}{2} + f = v - \frac{f}{2}$

$$\Rightarrow 2v - 2\chi(F) = f$$

$$2v - 2\chi(F) = \frac{2}{3}e$$

$$3v - 3\chi(F) = e$$

Induced Face Set

$p_0 \quad x_0 \quad y_0 \quad z_0$

p_1

$\vdots$

$p_{n-1} \quad x_{n-1} \quad y_{n-1} \quad z_{n-1}$

t_0

x

y

z

$\Leftarrow$

Indizes der Eckpunkte vom Dreieck

$\vdots$

t_{n-1}

$\vdots$

Definition: Eine Abbildung φ zwischen zwei Simplicialkomplexen K, K' heißt simpliciale Abbildung, genau dann wenn

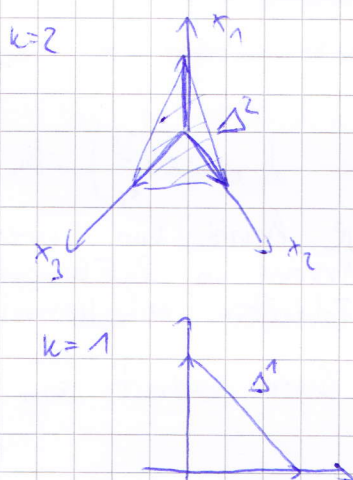
- $\varphi: v \rightarrow v' : \varphi(v_i) = v'_i$ bijektiv
- $\sigma = [v_0, \dots, v_n] \in S$ ein Simplex genau dann, wenn $[\varphi(v_0), \dots, \varphi(v_n)] \in S'$ ein Simplex sein.

Eine simpliciale Abbildung ist eindeutig bestimmt durch Angabe der Abbildung $\varphi: v \rightarrow v'$ der Punkte.

Geometrische Realisierung

Definition: Ein Standard- k -Simplex ist gegeben durch die konvexe Hülle der $(k+1)$ -Einheitsvektoren in $\mathbb{R}^{k+1}$

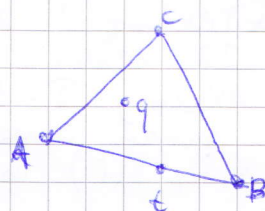
Beispiel: $\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \in \mathbb{R}^3, k=2$



$$\Delta^k = \left\{ \sum_{i=0}^k \lambda_i e_i \mid 0 \leq \lambda_i \leq 1 \text{ und } \sum_{i=0}^k \lambda_i = 1 \right\}$$

Jeder Punkt $q \in \Delta^k$ ist eindeutig durch Angabe der Koeffizienten $\lambda_0(q), \dots, \lambda_k(q)$ beschrieben. Daher werden die Koeffizientenvektoren $(\lambda_0, \dots, \lambda_k)$ auch baryzentrische Koordinaten der Punkte in Δ^k genannt.

Bemerkung: Sei $q \in \Delta^2$



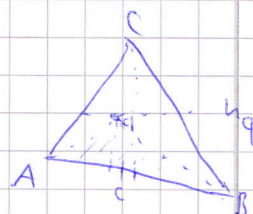
$$A = (1, 0, 0), \text{ da } 1 \cdot A + 0 \cdot B + 0 \cdot C$$

$$t = (1-t)A + tB + 0 \cdot C$$

$$t = ((1-t), t, 0)_b$$

Schwerpunkt hat die Koordinaten $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})_b$

$$\lambda_i(q) = \frac{\text{Fläche}(A, B, q)}{\text{Fläche}(A, B, C)} = \frac{c \cdot h_q / 2}{c \cdot h_c / 2} = \frac{h_q}{h_c}$$



Definition: Ein geometrischer Simplicialkomplex $(K(V, S), P)$ besteht aus einem abstrakten Simplicialkomplex $K = (V, S)$ und einem Vektor $P = (p_0, p_1, \dots)$ von Punkten im Raum $\mathbb{R}^d$ mit der Eigenschaft:

$$i) \exists \text{ Bijektion } \varphi: V \rightarrow P, \varphi(v_i) = p_i$$

$$ii) \text{ Für jeden } \sigma = [v_0, \dots, v_n] \in S \text{ gibt es eine affine lineare Abbildung } \varphi_\sigma: \Delta^n \rightarrow \mathbb{R}^d \text{ mit } \varphi_\sigma(e_i) = p_{v_i}$$



Bemerkung: K ist die simpliziale Struktur und P ist die geometrische Realisierung. Durch die Angabe von P ist die geometrische Lage in $\mathbb{R}^d$ eindeutig bestimmt.

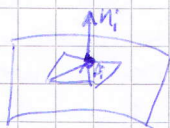
Die Abbildungen $\varphi_k: \Delta^k \rightarrow \mathbb{R}^d$ ist implizit gegeben durch

$$\varphi_k \left(\sum_{j=0}^k \lambda_j e_j \right) = \sum_{j=0}^k \lambda_j p_{kj}$$

beide Punkte besitzen die selben baryzentrischen Koordinaten

In der Praxis werden weitere Informationen in den Datenstrukturen benötigt.

1. Normalektoren $N = (n_1, n_2, \dots, n_k)$, $n_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}$, $|n_i| = 1$



je Punkt ein weiterer Array in P_k Element-Set PdVector vertexNormal.

2. Vektorfelder: ~~noch~~ zu jedem Punkt existiert ein Vektor $v_i \in \mathbb{R}^d$ (vector field)

3. Texturen:

vertexTexture: an jedem Punkt existiert ein 2D Vektor $t \in \mathbb{R}^2$. Dies induziert eine Abbildung $T: \mathbb{R}^d \rightarrow \mathbb{R}^2$. Dadurch erhält man einen neuen ~~komplexen~~ simplizialkomplex $(K = (V, S), T(P))$ in $\mathbb{R}^2$

Geometrische Transformationen

(Lineare Abbildung von Geometrien in $\mathbb{R}^3$)

9.11.2010

Vorlesung 7

Skalierung $S = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{pmatrix}$

Rotation um z-Achse

$$R_z = \begin{pmatrix} \cos \alpha & \sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Translation

$$T = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

Die Transformationen werden jeweils auf alle Punkte P der Geometrie angewendet.

Kombination

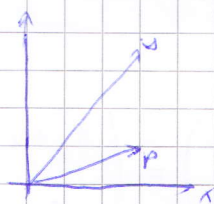
$$p_i \rightarrow S p_i + T p$$

das möchten wir gerne in eine große Matrix zusammen fassen.

Homogene Koordinaten in $\mathbb{R}^4$

Zu jedem Punkt $P = (p_x, p_y, p_z) \in \mathbb{R}^3$ ordnen wir eine Linie $L_P \in \mathbb{R}^4$ zu

$$L_P = \left\{ \lambda \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} \mid \lambda \in \mathbb{R} \setminus \{0\} \right\}$$



1. Der Punkt $p^n = \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} \in L$ kann als Repräsentant der Linie aufgefasst werden, da alle anderen Linienpunkte durch Skalierung eindeutig aus p^n konstruiert werden.
2. Zwei Punkte p^n und q^n liegen auf der gleichen Linie, genau dann wenn $p^n = \mu q^n$ für ein $\mu \in \mathbb{R} \setminus \{0\}$.
3. Wir führen auf dem ~~ganzen~~ $\mathbb{R}^4 \setminus \{0\}$ eine Äquivalenzrelation ein, d. h. $p, q \in \mathbb{R}^4$ mit $p \sim q$ genau dann wenn $\exists \mu \in \mathbb{R} : p = \mu q$.
4. $\exists$ ist $p \sim q \Leftrightarrow p, q$ liegen beide auf einer Geraden L .

5. Für einen Punkt $p \in \mathbb{R}^3$ bezeichnen wir den durch $L(p)$ Äquivalenzklasse $[p] = \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix}$ als homogene Koordinate.

Beispiel: $p = \begin{pmatrix} 2 \\ 3 \\ 7 \end{pmatrix}$ $p_H = \begin{pmatrix} 2 \\ 3 \\ 7 \\ 1 \end{pmatrix}$

Ähnlich wie p_H liegen alle Vielfachen von p_H

Sei $L \subset \mathbb{R}^4$ eine Gerade. Wie findet man den zugehörigen Punkt in $\mathbb{R}^3$? Wähle einen Punkt $q \in (q_x, q_y, q_z, q_t) \in L$ und dann ist $(q_x/q_t, q_y/q_t, q_z/q_t, 1) \in L$ und

$$L^{-1}(q) = \begin{pmatrix} q_x/q_t \\ q_y/q_t \\ q_z/q_t \end{pmatrix} \in \mathbb{R}^3 \quad (q_t \text{ muss natürlich } \neq 0 \text{ sein})$$

Wir haben also eine Bijektion zwischen Punkten in $\mathbb{R}^3$ und Geraden in $\mathbb{R}^4$.

In der Computergrafik ist eine zentrale Anwendung von homogenen Koordinaten die vereinheitlichte Schreibweise von affinen Abbildungen, die dadurch lineare Abbildungen von homogenen Koordinaten geschrieben werden können:

Sei $p = (p_x, p_y, p_z) \in \mathbb{R}^3$. p soll mit affiner Abbildung

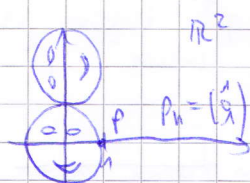
$A_p + v$ abgebildet werden
 $\uparrow$ lin. Abbildung $\uparrow$ Translation

$$p_H = \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} \quad \underbrace{\begin{pmatrix} | & A & | \\ \hline 0 & 0 & 0 & 1 \end{pmatrix}}_{A_v^H} \begin{pmatrix} v \\ 1 \end{pmatrix} = \begin{pmatrix} p_x + v_x \\ p_y + v_y \\ p_z + v_z \\ 1 \end{pmatrix}$$

A_v^H ist eine lineare Abbildung in $\mathbb{R}^4$ mit

$$q_H \rightarrow A_v^H(p_H) = (A_p + v)_H$$

Beispiel



$$A_v^H = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 2 \\ 0 & 0 & 1 \end{pmatrix}$$

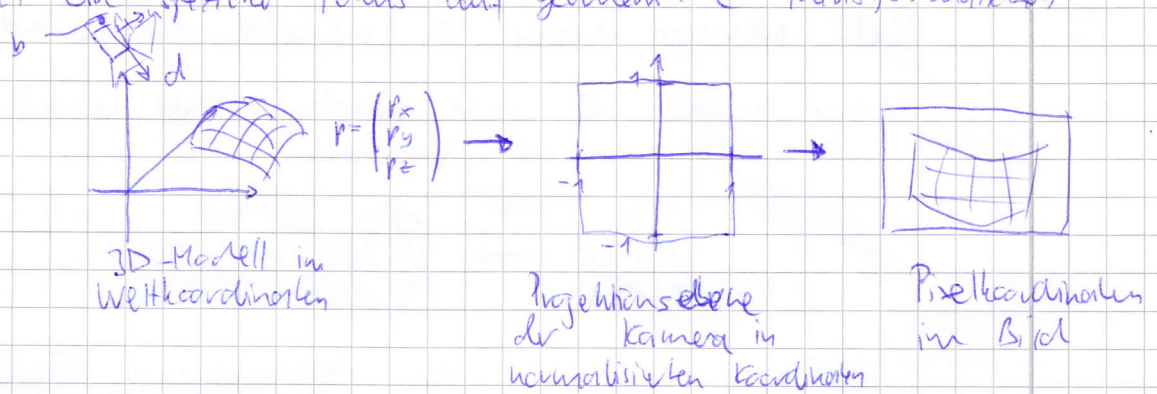
Matrizen erstellen für Abbildung der Rotation um 90° und Translation um $(0, 2)$.

$$A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad t = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$$

$$A_v^H \cdot p_H = \begin{pmatrix} 0 \\ 3 \\ 1 \end{pmatrix} \Rightarrow A_p + v$$

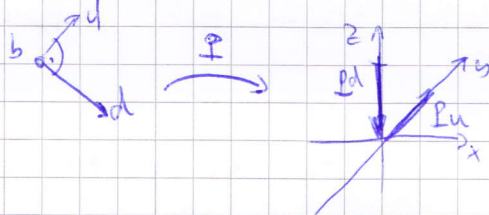
Rendering-Pipeline: Die Erstellung eines (digitalen) 2D-Bildes aus einem (digitalen) 3D-Bild.

Jetzt ein spezieller Fokus auf geometrische Transformationen



Aufgabe: Berechnung der Projektionsabbildung von 3D-Modellkoordinaten in Koordinaten.
affine Abbildung P

Gegeben: Kameraposition $b \in \mathbb{R}^3$ und Kamerarichtung $d \in \mathbb{R}^3$ und den u -Vektor $u \in \mathbb{R}^3$



P soll die Kamerarichtung auf die negative z-Achse abbilden und u soll auf positive y-Richtung abgebildet werden. b soll auf den Nullpunkt abgebildet werden.

Vorgehensweise:

1. Verschiebe b nach 0 $T = \begin{pmatrix} -b_x \\ -b_y \\ -b_z \end{pmatrix}$
2. Drehe den Vektor d zur z-Achse
3. Drehe um z-Achse, bis u in y-Richtung zeigt.
4. Normalisierung auf $[-1, 1] \times [-1, 1]$

Dadurch erhalten wir die Projektionsmatrix P . Anschließend wird jeder Punkt der Fläche mit P ~~multipliziert~~ ^{abgebildet} und kommt dadurch im ~~normalisierten~~ ^{normalisierten} Normalkoordinatensystem zu liegen.

Projektion auf die jetzt in xy-Ebene liegende Kameraebene geschieht einfach durch Weglassen der 2. Komponente.

Dadurch liegt das Bild der projizierten Geometrie im Quadrat $[-1, 1] \times [-1, 1]$.

Anschließend wird das Bild gemäß der Pixelgröße $w \times h$ auf die richtige Ausgangsgröße skaliert.

P charakterisiert vollständig Position und Projektion der Kamera.

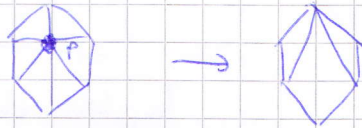
Summary: $p \in \mathbb{R}^3 \xrightarrow{P} P \cdot p \in \text{Normaloord.} \xrightarrow{w \times h} \text{Bildgröße}$
Weglassen der z-Komp.

Simplifizierung von Simplicialen Flächen

Gegeben: Fläche S mit vielen Dreiecken

Ziel: weniger Dreiecke bei möglichst gleichbleibender Fläche

Ad-hoc-Ansatz: 1. läßt einzelnen Punkt und 2. verknüpfen den $\text{unk}(p)$.



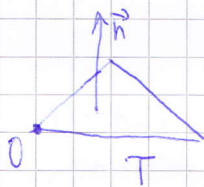
Nachteil: Fehler bei der Verknüpfung laufen sich an und es besteht nur schwer Kontrolle zum ursprünglichen Objekt

Wir brauchen ein gewisses Kriterium, um den Abstand der jeweils vergrößerten Fläche zur Ausgangsfläche messen zu können.

11.11.2010

Alaaf!
Hecken!

Vorlesung 8



$p \in \mathbb{R}^3$

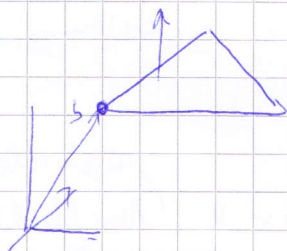
Abstand von p zur Ebene des Dreiecks

$$d(p, T) = \langle n, p \rangle \quad (\text{Skalarprodukt})$$

Diese Definition hat noch ein Vorzeichen

Daher nimmt man das quadratische

$$d(p, T)^2 = \langle n, p \rangle^2 = \underbrace{(-p)^T}_{\langle p, n \rangle} \underbrace{\begin{pmatrix} 1 \\ n \end{pmatrix}}_{\langle n, p \rangle} = (-p)^T \begin{pmatrix} 1 \\ n \end{pmatrix}$$



$$d(p, T)^2 = \langle n, p-b \rangle^2 = (-p-b)^T \begin{pmatrix} 1 \\ n \end{pmatrix}$$

$$= p^T \cdot (n \times n) \cdot p - b^T \cdot (n \times n) \cdot p - p^T (n \times n) \cdot b + b^T (n \times n) \cdot b$$

In homogenen Koordinaten:

$$\begin{pmatrix} p & 1 \end{pmatrix} \begin{pmatrix} (n \times n) & (n \times n) \cdot b \\ -b^T \cdot n & b^T (n \times n) \cdot b \end{pmatrix} \begin{pmatrix} 1 \\ p \end{pmatrix}$$

Zeilenvektor

Q ist 4×4 Matrix

$$d(p, T)^2 = p^T \cdot Q \cdot p$$

Der quadrierte Abstand zu einem beliebigen Dreieck ist ein quadratischer Ausdruck in homogenen Koordinaten

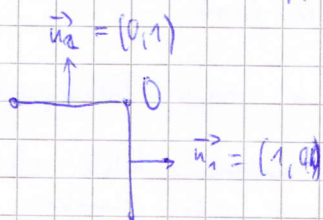
Beispiel in $\mathbb{R}^2$



Summe der Abstände von p zu zwei Geraden mit gemeinsamer Ecke b

$$d(p, \wedge) = \sum_{i=1}^2 p_H^T Q_i p_H$$

1.



Wir rechnen Q aus.
 $(n_1 \times n_2) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$

$$p_H^T Q_1 p_H = p_1^2$$

$$p_H^T Q_2 p_H = p_2^2$$

$$d(p, \gamma) = p_1^2 + p_2^2 = p_H^T \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} p_H$$

$$Q_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$Q_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

2.



$$d(p, \rightarrow) = 2 \cdot p_2^2$$

Sei S eine simpliziale Fläche mit einem Vektor b .
 Wir berechnen jetzt den Abstand von $p \in \mathbb{R}^2$ zu $\text{star}(S)$.

$$\begin{aligned} d(p, \text{star}(S)) &= \sum_{j=1}^k d(p, T_j) \\ &= \sum_{j=1}^k p_H^T Q_j p_H \\ &= p_H^T \left(\sum_{j=1}^k Q_j \right) p_H \\ &= p_H^T Q(S) p_H \end{aligned}$$



D.h. $d(p, \text{star}(S))$ wird berechnet aus der quadratischen Form $Q(S) = \sum_{j=1}^k Q_j$, die zum Punkt b assoziiert ist

Definition: $Q(b)$ heißt quadratische Fehlermetrik, $Q(S)$ ist eine 4×4 -Matrix.

Da $Q(b)$ eine quadratische, symmetrische und ~~positive definite~~ Matrix ist, hat $Q(S)$ in $\mathbb{R}^2$ genau $\det 1$

$Q(S)$ ist Ellipse in $\mathbb{R}^2$ und Ellipsoid in $\mathbb{R}^3$, wobei die Hauptachsen in einem naheliegen flachen Stern in k "Normalenvektoren der Seiten und tangential dazu zeigen.

Wenn b im Ursprung liegt, dann ist von Q nur der 3×3 Anteil relevant.

$$Q_n = \sum_{j=1}^k n_j \otimes n_j$$

Q_n ist eine symmetrische Matrix und ihre drei (in $\mathbb{R}^3$) Eigenwerte sind die Hauptachsen der Abstandselipsen.

Algorithmus von Garland und Heckbert

Sei S eine simpliziale Fläche mit N Punkten.
Ziel: Neue Fläche mit $M < N$ Punkten.

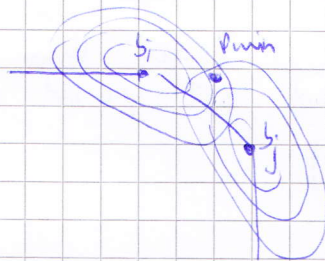
1. Berechne für jeden Vertex b_i ein Fehlerquadrat Q_i .

2. Für jede Kante $e = (b_i, b_j)$ berechne die Summe $Q_e = Q_i + Q_j$ und berechne den Punkt $p_{\min} \in \mathbb{R}^3$, so dass $\|p_{\min}\|_{Q_e}^2 := \min_{p \in \mathbb{R}^3} p^T Q_e p$

$$= \min_{p \in \mathbb{R}^3} \{ p^T Q_i p + p^T Q_j p \}$$

$$= \min_{p \in \mathbb{R}^3} \{ d^2(p, \text{star}(b_i)) + d^2(p, \text{star}(b_j)) \}$$

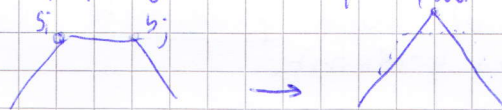
Da der Abstand quadratisch ist, ist das Minimum eindeutig gegeben.



Der Fehlerwert von $p_{\min}$ $Q_e p_{\min} = Q(p_{\min})$ wird der Kante zugeordnet.

3. Die Fehler aller Kanten kommen in eine priority queue (d.h. werden effizient der Größe nach sortiert) ein.

4. Sei e die Kante mit ~~kleinstem~~ ^{kleinstem} Fehler: "edge collapse", d.h. verschmelze Eckpunkte der Kante zu neuem Punkt $p_{\min}$. $p_{\min}$ erhält Fehlerquadrat $Q(p)$ zugeordnet.



5. Aktualisiere die Kanten von $p_{\min}$ in der priority queue bis Anzahl der Vertices ~~gleich~~ ^{gleich} M ist. Dann Rech ab.

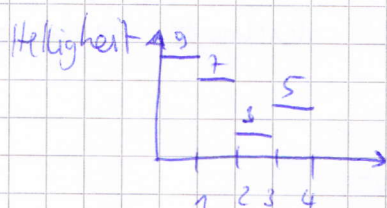
6. Goto 4.

16.11.2010

Wavelet - Darstellung von Pixel-Bildern

Vorlesung 9

Betrachte ein Bild über der reellen Achse

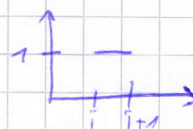


Als Vektor $[9, 7, 3, 5]$

Das Bild ist mathematisch dargestellt mittels Boxfunktionen:

$$u_i: \mathbb{R} \rightarrow \mathbb{R}$$

$$u_i(x) = \begin{cases} 1 & i \leq x < i+1 \\ 0 & \text{sonst} \end{cases}$$



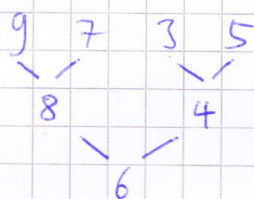
Basis für unsere Pixelbilder:

jedes Bild eindeutig darstellbar $u(x) = \sum_{i=-\infty}^{\infty} c_i u_i(x)$

Hier: $u(x) = 7u_1(x) + 3u_2(x) + 5u_3(x) + 9u_4(x)$

Wavelet - Transformation

Beispiel



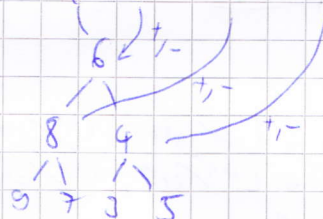
Differenz zum Mittelwert

1 -1 2 ← äquivalente Darstellung

Wavelet-Darstellung von $[9, 7, 3, 5]$ ist $[6, 2, 1, -1]_w$

Rekonstruktion

$$[6, 2, 1, -1]$$



Mittelwert des gesamten Bildes

Zustände des Gesamtbildes sei immer mehr info



Wir betrachten den Vektorraum von Funktionen

Für festes j : $V^j = \{f: [0,1) \rightarrow \mathbb{R} \mid f|_{[\frac{i}{2^j}, \frac{i+1}{2^j})} \equiv \text{konstant} \text{ und reell}\}$

Das Intervall $[0,1]$ wird in 2^j gleichgroße Teilintervalle geteilt.

Beispiel: $[9, 35] \rightarrow \subset V^2$

Bemerkung: Die Vektorräume sind geschachtelt

$$V^0 \subsetneq V^1 \subsetneq V^2$$

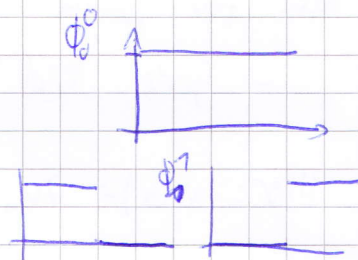
Basisfunktionen von V^j

$$\phi_i^j(x) = \begin{cases} 1 & \text{auf } [\frac{i}{2^j}, \frac{i+1}{2^j}) \\ 0 & \text{sonst} \end{cases}$$

Beispiel: V^0 aufgespannt von $\{\phi^0\}$

$$V^1 \quad \{\phi_0^1, \phi_1^1\}$$

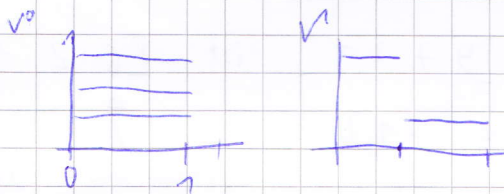
$$V^2 \quad \{\phi_0^2, \phi_1^2, \phi_2^2, \phi_3^2\}$$



Definition: Sei W^j das ~~Rechtskomplement~~ Komplement von V^j in V^{j+1}

Komplement von V^j in V^{j+1}

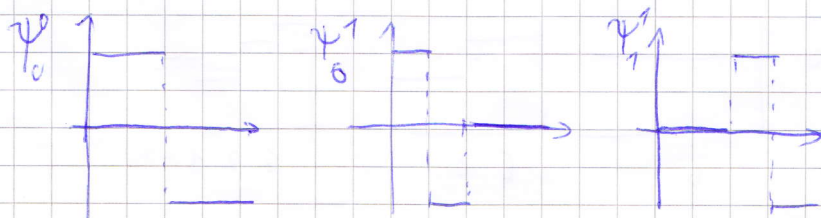
Wichtig: Basis für W^0



Betrachte Funktionen:

$$\psi_i^j(x) := \begin{cases} 1 & x \in [\frac{i}{2^{j+1}}, \frac{i+1}{2^{j+1}}) \\ -1 & x \in [\frac{i+1}{2^{j+1}}, \frac{i+2}{2^{j+1}}) \\ 0 & \text{sonst} \end{cases}$$

Diese Funktionen werden Haar-Wavelets genannt.



Lemma: Der Raum W^j wird aufgespannt von den Basisfunktionen ~~der~~

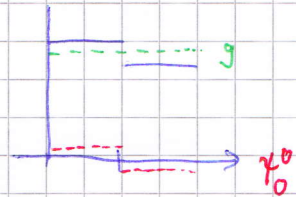
$$W^j = \text{span} \left\{ \begin{matrix} \psi_0^j \\ 0 \\ \vdots \\ 0 \\ \psi_{2^j-1}^j \end{matrix} \right\}$$

Sei $u(x) = 9\phi_0^1(x) + 7\phi_1^1(x) \in V^1$

$g(x) = 8\phi_0^0(x) \in V^0$

$h(x) = 1 \cdot \psi_0^0(x) \in W^0$

$u(x) = g(x) + h(x)$



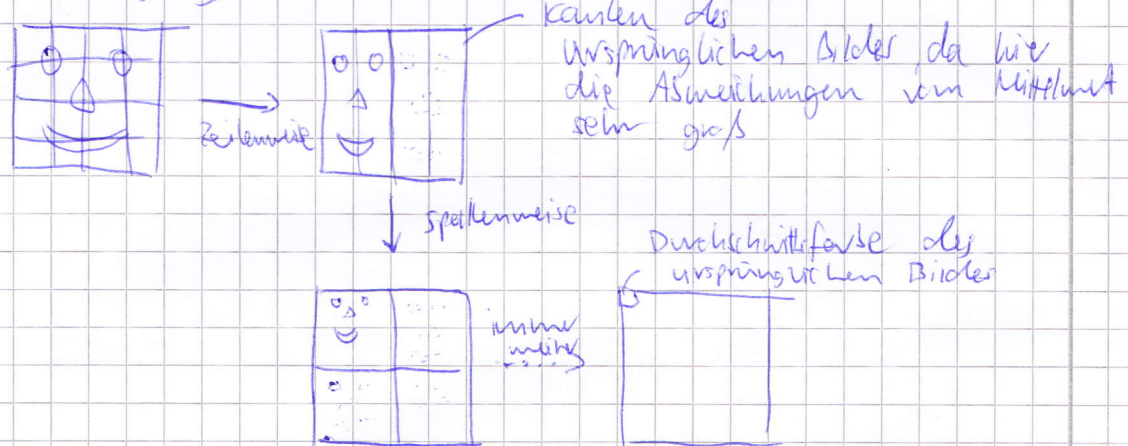
Allgemein: $V^1 = V^0 \oplus W^0$

$V^{j+1} = V^j \oplus W^j = \dots = V^0 \oplus W^0 \oplus \dots \oplus W^j$

$$\begin{aligned} u(x) &= \underbrace{9\phi_0^2(x) + 7\phi_1^2(x)}_{\in V^1} + 3\phi_2^2(x) + 5\phi_3^2(x) \in V^2 \\ &= 8\phi_0^1(x) + \psi_0^1(x) + 4\phi_1^1(x) + (-1)\psi_1^1(x) \in V^1 \oplus W^1 \\ &= 6\phi_0^0(x) + 2 \cdot \psi_0^0(x) + 1 \cdot \psi_0^1(x) + \dots + \psi_1^1(x) \\ &\quad \in V^0 \oplus W^0 \oplus W^1 \end{aligned}$$

Wavelet-Zerlegung

Wavelet-Zerlegung von Bildern



18.11.2010

$$f, g \in V^j \quad g, f: \mathbb{R} \rightarrow \mathbb{R}$$

Wir wollen eine Norm $\|g-f\|$ definieren.

Das Skalarprodukt definieren wir als

$$\langle f, g \rangle = \int_{\mathbb{R}} f(x) \cdot g(x) dx$$

Es folgt, wenn f orthogonal zu g ,
 $\Leftrightarrow \langle f, g \rangle = 0$

Die Norm ist nun definiert als $\|f\|^2 := \langle f, f \rangle$. Für

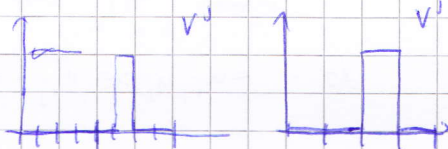
$$\text{gilt } \|f-g\|^2 = \int_{\mathbb{R}} |f(x)-g(x)|^2 dx$$

← L_2 -Abstand von zwei Bildern ist definiert als Integral über die Abstände der Funktionswerte.

1) Basisfunktionen der Boxfunktionen von V^j sind orthogonal, da sie unterschiedliche Träger haben

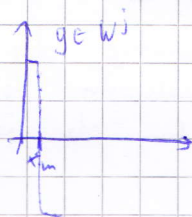
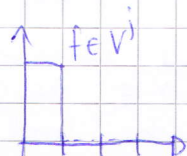


2) Basisfunktionen von V^j verglichen mit V^{j-1} sind nicht orthogonal: $V^j \not\perp V^{j-1}$. Das ist auch logisch, denn V^{j-1} ist Unterraum von V^j



3) $V^j \oplus W^j$

Beweis:



f und g haben entweder

- gemeinsamen Träger $\Rightarrow \langle f, g \rangle = 0$ oder
- sind über dem selben Teilintervall definiert, und

$$\int_{1/2^j}^{1/2^{j-1}} f(x)g(x) dx = \int_{1/2^j}^{1/2^{j-1}} 1 \cdot 1 dx + \int_{1/2^{j-1}}^{1/2^{j-2}} 1 \cdot (-1) dx = 0$$

$$\Rightarrow V^j \oplus W^j$$

Korollar: Die Wavelet-Zerlegung von V^j ist orthogonal.

$$V^j = V^0 \oplus W^0 \oplus W^1 \oplus \dots \oplus W^{j-1}$$

Aus technischen Gründen ersetzen wir die Basisfunktionen ϕ_i^j und ψ_i^j durch normierte Funktionen

$$\bar{\phi}_i^j = \frac{\phi_i^j}{\|\phi_i^j\|}$$

$$\text{und } \bar{\psi}_i^j = \frac{\psi_i^j}{\|\psi_i^j\|}$$

$$\|\phi_i^j\| = \sqrt{\frac{1}{2^j}}$$

damit $\|\bar{\phi}_i^j\| = 1$ und $\|\bar{\psi}_i^j\| = 1$

Dadurch wird $\{\bar{\phi}_0^0, \bar{\psi}_0^0, \bar{\phi}_0^1, \dots, \bar{\psi}_{2^j-1}^{j-1}\}$ eine Orthonormalbasis von V_j .

Bild f : $f(x) = \sum_{i=1}^m c_i u_i(x)$ und $\{u_1, \dots, u_m\}$ Orthonormalbasis
 Koeffizienten Basisfunktionen

Bemerkung: Bei gegebenem Satz von Basisfunktionen speichert der Koeffizientenvektor $(c_1, \dots, c_m)$ die gesamte Information der Abbildung.

Ziel: Versuche die Anzahl der Koeffizienten zu reduzieren, ohne dabei einen vorgegebenen Fehler von der reduzierten Version zu überschreiten: $\|\tilde{f} - f\| \leq \epsilon$.

π sei eine Permutation der Zahlen $1, 2, \dots, m$. Dann betrachte die ersten m Koeffizienten $\tilde{f}(x) = \sum_{i=1}^m c_{\pi(i)} u_{\pi(i)}(x)$

Der Abstand des reduzierten Bildes $\tilde{f}$ zu f ist:

$$\begin{aligned} \|f - \tilde{f}\|^2 &= \langle f - \tilde{f}, f - \tilde{f} \rangle = \left\langle \sum_{i=m+1}^m c_{\pi(i)} u_{\pi(i)} \right\rangle \left\langle \sum_{i=m+1}^m c_{\pi(i)} u_{\pi(i)} \right\rangle \\ &= \sum_{i=m+1}^m c_{\pi(i)}^2 \quad \text{da} \quad \langle u_i, u_j \rangle = \delta_{ij} = \begin{cases} 1 & \text{für } i=j \\ 0 & \text{sonst} \end{cases} \end{aligned}$$

D.h. der Fehler ist die Summe der Quadrate der weggelassenen Koeffizienten.

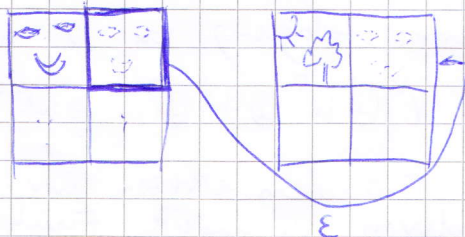
Algorithmus: Streiche solange die jeweils kleinsten Koeffizienten $\{c_1, \dots, c_m\}$ bis irgendwann der Fehler ϵ überschritten wird, bzw. sortiere $c_1, \dots, c_m$ der Größe nach und streiche die kleinsten Koeffizienten.

Dies ist die Wavelet-Kompression.

Multiresolution-Analyse eignet sich für

- Kompression
- Hierarchisches Zeichnen
- Übertragung von Detailinformationen (water marking) von einem Bild auf ein anderes

Beispiel:

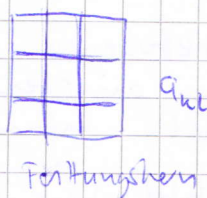


Filtern von Bildern / Faltung

23.11.2010

Vorlesung 11

Pixelbild
 f



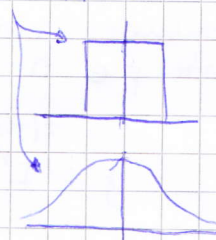
$$\bar{f}(x_{ij}) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} a_{kl} f(x_{i+k, j+l})$$

$f(x_{ij})$ ist Farbe im ursprünglichen Bild an der Stelle x_{ij}

Glatte (nicht diskrete) Formulierung:

Sei $f: \mathbb{R} \rightarrow \mathbb{R}$ eine Funktion, $h: \mathbb{R} \rightarrow \mathbb{R}$ ist der Filterkern.

$$\bar{f}(x) = \int_{-\infty}^{\infty} f(\tau) \cdot h(\tau - x) d\tau$$



z.B. $h(x) = \begin{cases} 1/2 & -2 \leq x \leq 2 \\ 0 & \text{sonst} \end{cases}$

$$\bar{f}(x) = \int_{-\infty}^{\infty} f(\tau) h(\tau - x) d\tau = \frac{1}{2} \int_{x-2}^{x+2} f(\tau) d\tau$$

Filterkern $\begin{bmatrix} -1 & 1 \end{bmatrix}$

$$\bar{f}(x_i) = f(x_i) - f(x_{i-1})$$

das entspricht der diskreten Ableitung / Differentialquotient

Kantendetektion $\bar{f}: [-1, 1]$ ist nahezu 0 in homogenen Bildbereichen und hat starke Ausschläge $\neq 0$ an Kanten.

Visualisierung von Vektorfeldern / Strömungen

Typisches Beispiel: Wetterbericht mit dunklen Pfeilen, die die Windrichtung angeben. Das ist nicht so schön und verdeckt viel Information.

Daher ~~das~~ Portal über das Bild strömen lassen.
Idee: lasse Blende der Kamera für kurze Zeit 21 offen

Vektorfeld

Fläche $M \in \mathbb{R}^2$. Das Vektorfeld ordnet jedem $p \in M$ einen Vektor zu

$$v: M \rightarrow \mathbb{R}^2 \\ p \mapsto v(p) \in \mathbb{R}^2 / T_p M$$

Bestimmung einer Bahnkurve welches zum Zeitpunkt 0 am Punkt $p \in M$ startet

$$\gamma: \mathbb{I} \rightarrow M \\ t \mapsto \gamma(t)$$

$$\text{AWP: } \gamma'(t) = v(\gamma(t)) \\ \gamma(0) = p$$

Anfangswertproblem für gewöhnliche Differentialgleichung.

Wenn v Lipschitz-stetig, so existiert eine eindeutige Bahnkurve.

Numerische Lösung eines Anfangswertproblems

$$\gamma(0) = p$$

$$x_1 := p + h \cdot v(p) \quad \text{mit } h \text{ klein!}$$

$$x = x_i + h \cdot v(x_i) \quad \text{Euler-Verfahren}$$

Bei rotations-symmetrischen Vektorfeldern erzeugen wir nicht spiralförmige Bahnkurven

Runge-Kutta Verfahren: ähnlich wie Euler, allerdings mehrfache Auswertungen um die "Schnitt-Richtung" zu bestimmen.

$$v^1 = v(x_i)$$

$$v^2 = v\left(x_i + \frac{h}{2} v^1\right)$$

$$v^3 = v\left(x_i + \frac{h}{2} v^2\right)$$

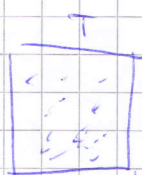
$$v^4 = v\left(x_i + h v^3\right)$$

$$v_i := \frac{1}{6} (v^1 + 2v^2 + 2v^3 + v^4)$$

$$x_{i+1} = x_i + h \cdot v_i$$

Methode der Linienintegralfaltung (LIC)

Texturfunktion $T: M \rightarrow [0,1]$ Graustufen mit 0 schwarz und 1 weiß
 $p \rightarrow \text{random}(p)$ welcher Rauschen ist eine relativ zufällige gewählte Farbverteilung



Für jedes $p \in M$ berechne das LIC-Bild F durch folgende Faltung (längs einer Bahnkurve durch p)

$$F(p) = \int_{-l}^l T(\gamma(\tau)) h(\tau) d\tau$$

gewichtet Mittelwert des Rauschens längs der Bahnkurve durch p .

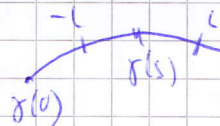
Vorlesung 12 Diskretisierung vom Faltungsintegral

25.11.2010

$$F(\gamma(s)) = \int_{s-l}^{s+l} T(\gamma(\tau)) h(\tau) d\tau$$

$$h = \text{Boxfunktion} \\ = \frac{1}{2l}$$

$$\text{damit } F(\gamma(s)) = \frac{1}{2l} \int_{s-l}^{s+l} T(\gamma(\tau)) d\tau$$



Diskretisiere $[-l, l]$ in $2L$ Intervalle.

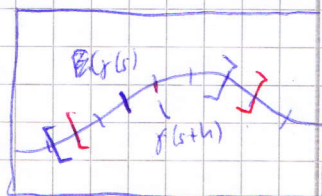
$$\approx \frac{1}{2L+1} \sum_{i=-L}^L T(\gamma(s+ih)) \quad \text{mit } h = \frac{l}{L} \quad (*)$$

Durch F wird das LIC-Bild als Visualisierung des Vektorfeldes erzeugt.

Idee: • zur Reduzierung des Aufwands bei der Berechnung der Bahnkurve (die ja für $\gamma(s+Lh)$ dieselbe ist) berechnen wir die Bahnkurve so weit wie möglich.

• zur Berechnung von $F(\gamma(s+Lh))$ verwende den schon berechneten Wert $F(\gamma(s))$

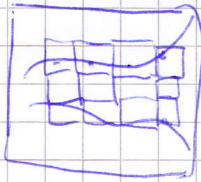
$$F(\gamma(s+Lh)) = F(\gamma(s)) - \frac{1}{2L+1} T(\gamma(s+Lh)) + \frac{1}{2L+1} T(\gamma(s+(L-1)h)) \quad (**)$$



Algorithmus (naïv)

Gegeben: Fläche M und (kantenhaare) Vektorfeld v
 gewünschte Auflösung des LIC-Bildes F

1. Für jeden Pixel von F berechne Richtung durch F (so weit wie möglich)
2. Von p startend berechne $F(p) = F(x|0)$ durch Berechnung der Summe $(*)$
 Anschließend laufe entlang x und berechne $F(x|i \cdot h)$ nach $(**)$
3. goto 1. und wähle nächsten "freien" Pixel aus.



Da Pixel am Rand geschaffen werden können, sollte man nicht beim ersten Hit einen Pixel als berechnet ansehen.

Algorithmus (Pseudocode)

for each pixel p

if numHits $(p) < H \approx 3-5$ then

compute x with $x(0)$ - center of p und Schrittweite h
 (also M Strahlen)

compute $F(x|0)$ [nach $(*)$]

for $i = 1, \dots, M$

~~$F(x|i \cdot h)$ [nach $(**)$]~~

~~if (numHits $(x(i \cdot h)) \geq H$~~

~~if (numHits $(x(i \cdot h)) \geq 1$ then~~

~~break~~

compute $F(x|i \cdot h)$ [nach $(**)$]

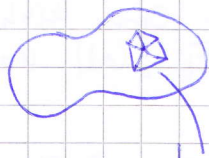
add $F(x|i \cdot h)$ zum Wert des Pixels $F(x|i \cdot h)$
 und numHits $(x|i \cdot h) ++$

for each pixel p

$F(p) / \text{numHits}(p)$

Erinnerung:

Simpliziale Fläche $S = (K, V)$
(index-basierte) Textur $T: S \rightarrow \mathbb{R}^2$



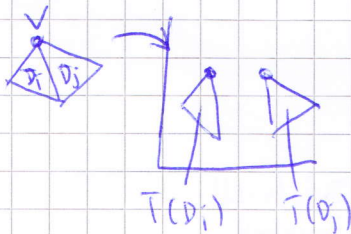
T ist Simplicialkomplex mit selben K und $T(v)$



Bei LIC würde es zu Verzerrungen führen, wenn man Nachbarschaft der Dreiecke in der Ebene findet.

Alternativ: element-basierte Texturen

Jeder Dreieck $t \in K$ erhält drei eigene Texturkoordinaten, für jeden seiner Eckenpunkte



Wenn T isometrisch ist, kann man stetigkeit an den kanten und keine Verzerrung

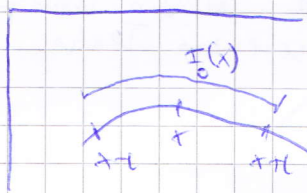
Vorlesung 13

Animiertes LIC-Bild

30.11.2010

Die weißen und schwarzen "Inseln" eines LIC-Bildes sollen sich mit der Zeit längs der Bahnkurve bewegen.

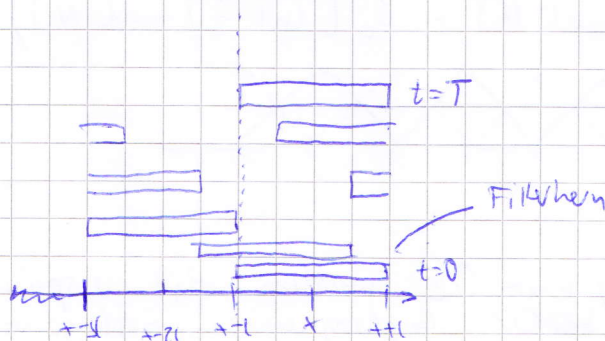
Idee: Verschiebe den Faltungskern



$$F_0(x) = F(x) = \int_{x-l}^{x+l} T(x(z)) \cdot h(z-x) dz$$

$$F_t(x) = \int_{(x-t)-l}^{(x-t)+l} T(x(z)) \cdot h(z-(x-t)) dz$$

Wir erhalten jetzt eine Animation $F_t: M \rightarrow \mathbb{R}$ mit $t \in [0, t_0]$. Für größeres t fehlt Information auf jenseits vom Rand.



Periodischer Filterkern mit Periode T , d.h. $F_t = F_0$

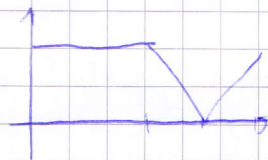
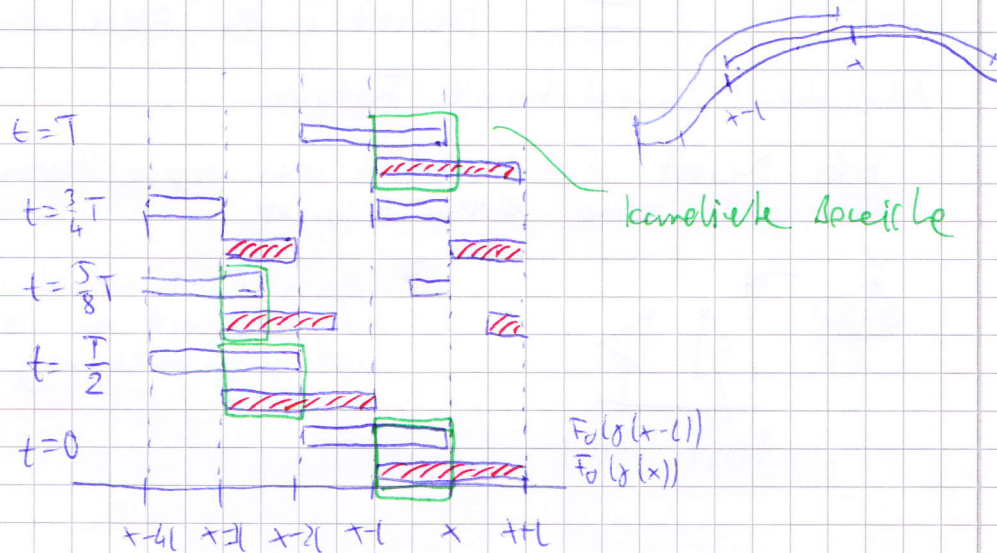
F_t ist stetige Funktion in t , da der Filterkern sich stetig mit t längs der Bahnkurve verschiebt

bzw. stetig wiederkehrt.

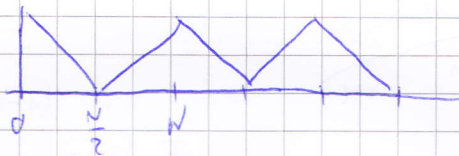
Fovealblinzproblem

Betrachte zwei ~~kurven~~^{Punkte} $y(x)$ und $y(x-L)$ auf einer Bahnkurve und die Korrelation der beiden Farbwerte $F_t(y(x))$ und $F_t(y(x-L))$.

Für $t=0$ sind $F_0(y(x_0))$ und $F_0(y(x-L))$ zu 50% korreliert



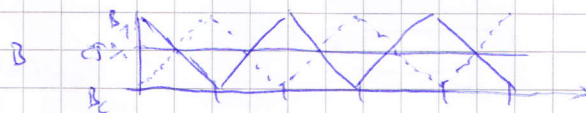
$B_1(n)$ ist eine Folge $n=1, \dots, N$ Bildern unserer LIC-Animation bzgl. unserer festen vorgegebenen Textur.



$B_2(n)$ ist eine zweite Bildfolge bzgl. einer zweiten Rauschtextur.



$$B(n) = \frac{1}{2} (B_1(n \bmod N) + B_2((n + \frac{N}{2}) \bmod N))$$



Durch die Mittelung mit $1/2$ werden die ursprünglichen Rauschfunktionen beider Bildfrequenzen orthogonal zu den Bahnkurven gemittelt, damit weniger Rauschen orthogonal zu Bahnkurven und d.h. weniger Kontrast

→ Gegenkennzeichnung durch Kontrastverstärkung

Einfachste Flächentexturen

1. Parametrisierte Flächen

$$F: \Omega = [0, u] \times [0, v] \rightarrow \mathbb{R}^3$$

(Texturen auf Wände und andere elementare Objekte)

$$T: \mathbb{R}^2 \rightarrow \text{PGS-Farbraum}$$

Die Farbe im Punkt $F(u, v)$ ist $cd(F(u, v)) = T(u, v)$

2. Für beliebige Flächen

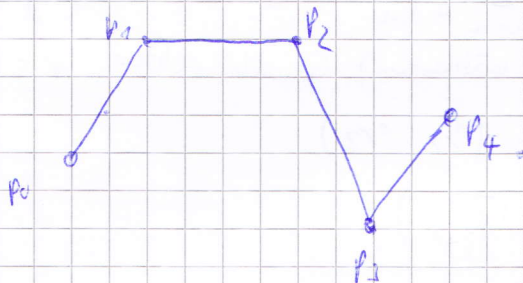
$$cd: \mathbb{R}^3 \rightarrow \mathbb{R} \quad \text{z.B.} \quad cd(x, y, z) = \sin x \sin y \sin z$$

z.B. Marmor

Aufgabe 14

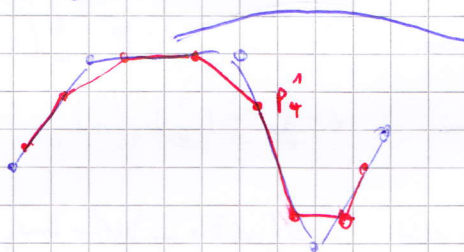
Subdivision-Verfahren

2.12.2010



1. Was soll überhaupt der Grenzwert sein?

Corner-Cutting Verfahren von Chaiken



im i -ten Schritt ist diese Kante auf $\frac{1}{2^i}$ der ursprünglichen Kante zentriert um den Mittelpunkt

Die Kantenmittelpunkte liegen auf der Grenzkurve von jedem Polygon

2. Erhalten eine Sequenz von Polygonen P^i mit

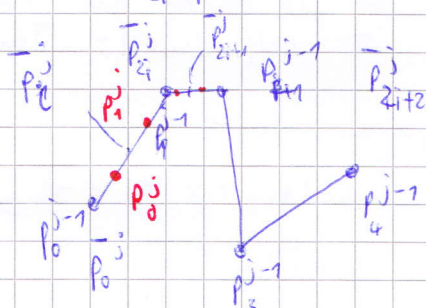
$$\lim_{i \rightarrow \infty} P^i = P^\infty \text{ "glatte" Grenzkurve}$$

Sei P^0 gegeben durch Punkte $\{p_0^0, p_1^0, \dots, p_n^0\}$.
Dann erfolgt ein Chaikin-Schritt in zwei Schritten:

1. Unterteilungsschritt: Neue Punkte werden auf den Kantenmitten eingeführt

$$\bar{p}_{2i}^j := p_i^{j-1}$$

$$\bar{p}_{2i+1}^j := \frac{1}{2} (p_i^{j-1} + p_{i+1}^{j-1})$$



2. Mittlungsschritt: Jeder Punkt $\bar{p}_i^j$ wird nun etwas verschoben, um Punkt p_i^j von Polygon P^j der Stufe j zu werden

allgemein: $\bar{p}_i^j \rightsquigarrow p_i^j := \sum_{k=-\infty}^{\infty} r_k \bar{p}_{i+k}^j$

mit einem (zu wählenden) Gewichtsvektor $r = (\dots, r_{-1}, r_0, r_1, \dots)$

Chaikin: $r = (r_0, r_1) = (1/2, 1/2)$

$$p_i^j = \frac{1}{2} (\bar{p}_i^j + \bar{p}_{i+1}^j)$$

Anforderung an den Gewichtsvektor: Summe r sollen $\sum_{n=-\infty}^{\infty} r_n = 1$ betragen.

andere Gewichtsvektoren: $r = (r_1, r_2, r_3) = \frac{1}{4} (1, 2, 1)$

bzw $r = \frac{1}{n} \left(\binom{n}{0}, \binom{n}{1}, \dots, \binom{n}{n} \right)$

Theorem: Das vorgestellte Subdivision-Verfahren mit r^* führt zu uniregulären B-Splines vom Grad $(n+1)$

Chaikin: quadratische B-Splines haben als Kontrollpolygon die ursprünglichen Polygone

Das Chaikin-Verfahren können wir alternativ interpretieren

1. Das gegebene Polygon wird in Sequenz von Bezier-Kontrollpolygone zerlegt

2. Im Grenzwert beschreibt P^∞ die Vereinigung

der entstehenden quadratischen ~~Polygone~~ ~~Segmente~~
Polygonsegmente

DLG-Schema (interpretierend) von (Dyn-levin, -Gregory)

$$p_i^j = \begin{cases} \bar{p}_i^j & i \text{ gerade} \\ \sum v_i \bar{p}_i^j & i \text{ ungerade} \end{cases}$$

$$r = (v_{-2}, \dots, v_2) = \frac{1}{16} (-2, 5, 10, 5, -2)$$

In jedem Schritt Subdivisions-Schritt werden die Punkte p_i^j vollständig und allein aus den Punkten $\{p_i^{j-1}\}$ erzeugt.

D.h. wir können die Subdivisionsverfahren auch ohne Einführung der Zwischenpunkte $\{p_i^j\}$ formulieren.

Bei Verwendung des Gewichtvektors

$$r = \frac{1}{4} (1, 2, 1) = (v_{-1}, v_0, v_1) \quad \text{erhalten wir}$$

$$p^j = \frac{1}{4} (\bar{p}_-^j + 2\bar{p}^j + \bar{p}_+^j)$$

$$= \frac{1}{4} \left(\frac{1}{2} (\bar{p}_-^{j-1} + \bar{p}_+^{j-1}) + 2\bar{p}^{j-1} + \frac{1}{2} (\bar{p}_-^{j-1} + \bar{p}_+^{j-1}) \right)$$

$$= \frac{1}{8} \bar{p}_-^{j-1} + \frac{1}{2} \bar{p}^{j-1} + \frac{1}{8} \bar{p}_+^{j-1}$$

Explizite Darstellung der $\{p_i^j\}$ aus $\{p_i^{j-1}\}$

Die direkte Berechnung ist bei DLG auch möglich.

Vorlesung
15
7.12.2010

Beobachtung: Für dieses Verfahren können wir allein durch Verwendung von drei benachbarten Kontrollpunkten eine unendliche Folge von Punkten generieren

$$p^0 \rightarrow p^j \rightarrow p^{j+1} \rightarrow \dots \rightarrow p^\infty \in \delta\text{-Spalte}$$

$$\begin{pmatrix} p_{-}^{j+1} \\ p_{+}^{j+1} \\ p_{+}^{j+1} \end{pmatrix} = \frac{1}{8} \underbrace{\begin{pmatrix} 4 & 4 & 0 \\ 1 & 6 & 1 \\ 0 & 4 & 4 \end{pmatrix}}_{\text{Subdivisionsmatrix } L} \begin{pmatrix} p_{-}^j \\ p_{+}^j \\ p_{+}^j \end{pmatrix}$$

Erlaubt aus den Punkten der Stufe $j-1$ direkt die Punkte der Stufe j zu erzeugen.

Der Grenzpunkt p^∞ der Folge $\{p^j\}$ lässt sich daher durch fortgesetzte Anwendung der Matrix L berechnen.

$$\lim_{j \rightarrow \infty} p^j = p^\infty$$

$$\underbrace{\lim_{k \rightarrow \infty} L^k}_{L^\infty} \begin{pmatrix} p_{-}^0 \\ p_{+}^0 \\ p_{+}^0 \end{pmatrix} = \begin{pmatrix} p_{-}^\infty \\ p_{+}^\infty \\ p_{+}^\infty \end{pmatrix} \quad \text{d.h. } p_{-}^\infty = p_{+}^\infty$$

Ziel: Wir wollen den Grenzwert p^∞ direkt aus den Anfangswerten $(p_{-}^0, p_{+}^0, p_{+}^0)$ berechnen können, damit wir gleich dem Punkt p^∞ auf der Grenzkurve kennen.

Dazu müssen wir L^∞ kennen.

Wir betrachten die rechtsseitigen Eigenvektoren $\{v_i\}$ von L mit Eigenwerten $\{\lambda_i\}$.

$$L v_i = \lambda_i v_i$$

Bei L einer $m \times m$ -Matrix ist $v \in \mathbb{R}^m$ und $i = 1, \dots, m$

Nummerierung nach der Größe der Beträge der Eigenwerte $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_m|$.

Damit die Konfiguration nicht beliebig groß wird, sollte gelten $|\lambda_2| < 1$.

Konstruiere

$$V = \begin{pmatrix} | & | \\ v_1 & v_m \\ | & | \end{pmatrix} \quad \text{enthält die Eigenvektoren als Spalten.}$$

$$A = V^{-1} \quad \text{Inverse zu } V, \text{ d.h. } id = V \cdot A$$

$$A = (a_{ij}) \quad \left(\begin{array}{c} \vdots \\ \uparrow \\ e_j^T \end{array} \right) = \left(\begin{array}{c|c} A & v_j^o \\ \hline -v_j^o & I \end{array} \right) \left(\begin{array}{c} \vdots \\ \uparrow \\ 0 \end{array} \right)$$

$\downarrow$
Spalte

$$e_i = \sum_{j=1}^m a_{ji} \cdot v_j$$

Subdivision k -mal anwenden

$$\begin{aligned} L^k \begin{pmatrix} p_1 \\ \vdots \\ p_m \end{pmatrix} &= L^k \sum_{i=1}^m e_i p_i \\ &= \sum_{i=1}^m L^k \left(\sum_{j=1}^m a_{ji} \cdot v_j \right) p_i \\ &= \sum_{i=1}^m \sum_{j=1}^m a_{ji} L^k v_j \cdot p_i \\ &= \sum_{i=1}^m \sum_{j=1}^m a_{ji} \lambda_j^k v_j \cdot p_i \end{aligned}$$

Annahme: $\lambda_1 = 1$ $|\lambda_k| < 1$ für $k \geq 2$

$$\Rightarrow \lambda_n^{\text{da}} = \begin{cases} 1 & n=1 \\ 0 & n \geq 2 \end{cases}$$

Damit = $\sum_{i=1}^m a_i v_i \cdot p_i$

$$\lim_{m \rightarrow \infty} L^k \begin{pmatrix} p_1 \\ \vdots \\ p_m \end{pmatrix} = \sum_{i=1}^m a_{ji} \cdot v_i \cdot p_i = \begin{pmatrix} \sum_{i=1}^m a_{1i} \cdot v_i \\ \vdots \\ \sum_{i=1}^m a_{ji} \cdot v_i \end{pmatrix} = \begin{pmatrix} v_1^{\text{ex}} \\ \vdots \\ v_m^{\text{ex}} \end{pmatrix}$$

Rei Zeit
ist eine
gewählte
Summe von
12 bis 14

Beispiel: $L = \frac{1}{8} \begin{pmatrix} 4 & 4 & 0 \\ 1 & 6 & 1 \\ 0 & 4 & 4 \end{pmatrix}$

$$v_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad \lambda_1 = 7$$

$$V_2 = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} \quad \alpha_2 = \frac{1}{2}$$

$$V_3 = \begin{pmatrix} 2 \\ -1 \\ 2 \end{pmatrix} \quad \rho_3 = \frac{1}{4}$$

$$V = \begin{pmatrix} 1 & -1 & 2 \\ 1 & 0 & -7 \\ 1 & 1 & 2 \end{pmatrix}$$

$$A = \frac{1}{6} \begin{pmatrix} 1 & 4 & 1 \\ -3 & 0 & 3 \\ 1 & -2 & 1 \end{pmatrix}$$

$$L \propto \begin{pmatrix} p_- \\ p \\ p_+ \end{pmatrix} = \frac{1}{6} \begin{pmatrix} (p_- + 4p + 7p_+) \cdot v_m \\ (p_- + 4p + 7p_+) \cdot v_{q2} \\ (p_- + 4p + 7p_+) \cdot v_{q3} \end{pmatrix}$$

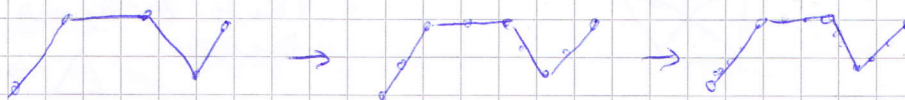
$$p^{aa} = \frac{1}{5} (1 p_- + 4 p_0 + 1 p_+)$$

Bemerkung:

1. Damit in $L^\infty \begin{pmatrix} p_1 \\ \vdots \\ p_n \end{pmatrix}$ alle Komponenten aus p^∞ folgen,
muss gelten $V_{ni} = V_{nn} = 1$. Damit ist p^∞ dieij
aus der ersten Zeile von $A = V^{-1}$ bestimmt.

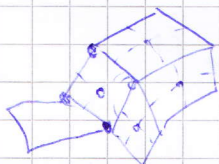
Subdivision von Flächen

Lineare Subdivision



$$v = (-010-)$$

1. Bilineare Subdivision (für Vierecke)

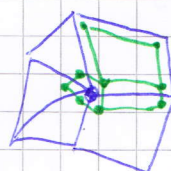


Unterteilung:

- Kantenmitten
- Schwerpunkt des Vierecks

2. Catmull-Clark (Ähnungen)

3. Doo-Sabin



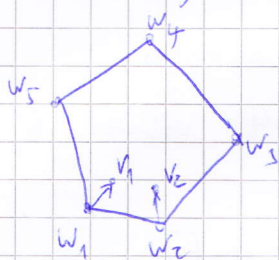
Valenz eines Punktes ist die Anzahl der ausgehenden Kanten.

Unterteilung: Jeder Eckpunkt erzeugt d Zwischenpunkte, je einen für die d angrenzenden Flächen.
Fläche mit k Seiten bleibt k Fläche, Kante erzeugt 4-Fläche, Vertex erzeugt d -Fläche

Wo liegen nun die neuen Punkte?

Mittelungsschritt:

$$v_i = \sum_{j=1}^k a_{ij} w_j$$



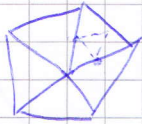
$$a_{ij} = \begin{cases} (k+5)/4k & i=j \\ \frac{3+2\cos(\frac{2\pi(i-j)}{k})}{4k} & i \neq j \end{cases}$$

Position der neuen Punkte einer Fläche f hängt nur von den bisherigen Eckpunkten von f ab.

Beliebige Zellkomplexe mit Dreiecken, Vierecken usw.

Nachteil: Die Anzahl der Flächen geht sehr stark wachsen.

4. Loop-Verfahren für Dreiecksnetze



- Kanten mit Hintereinander
- 1 zu 4 Zerlegung jedes Dreiecks



Mittelungsschritt:

Sei $\bar{p}$ ein Zwickelpunkt (d.h. Kanten mit alle bisherigen Verts und $\bar{p}_1, \dots, \bar{p}_k$ seine direkten Nachbarn, dann wird p ausgerechnet durch

$$p = \frac{1}{\alpha(k)+k} (\alpha(k)\bar{p} + \sum_{i=1}^k \bar{p}_i)$$

$$\alpha(k) = \frac{k(1-\beta(k))}{\beta(k)}$$

$$\beta(k) = \frac{5}{4} - \frac{1}{32} \left(3 + 2 \cos \frac{2\pi}{k} \right)^2$$

Butterfly-Schema (DLG Automaten)

Das erste interpolierende Verfahren nach dem Bilinearieren

- Man nimmt die Kantenmittelpunkte und macht 1 zu 4 Zerlegung

Mittelung: a) Kantenmitte p

sofern beide Endpunkte der Kante Valenz 6 haben

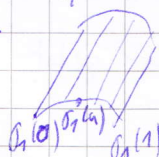
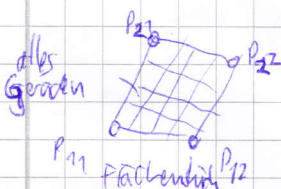
a) Die Eckpunkte bleiben erhalten

$\bar{p}$ liegt auf Kante, wobei eine Ecke Valenz 6 und andere Ecke Valenz m .

Durch neue Gewichtete

$$s_j = \frac{1}{m} \left(\frac{1}{4} + \cos \left(\frac{2\pi j}{m} \right) \right) + \frac{1}{2} \cos \frac{6\pi j}{m}$$

Coxs Patch: Gegeben eine Randkurve aus 4 Segmenten.
Gesucht: interpolierendes Flächenstück



$$F: [0,1] \times [0,1] \rightarrow \mathbb{R}^3$$

$$F_1(u,v) = (1-v)x_1(u) + vx_2(u)$$

interpoliert x_1, x_2 und beide Randgeraden

$$F(u,v) = (1-v) \left((1-u)p_{11} + up_{12} \right) + v \left((1-u)p_{21} + up_{22} \right)$$



Coxs-Patch interpoliert Randkurven

(Nachteil: Tangentialer Verlauf längs des Randes ist nicht unter Kontrolle d.h. 2 benachbarte Coxs-Patches haben oft keine gemeinsame Tangentialen)

und damit Normale längs des Randes

Messen auf / von Kurven und Flächen

14.12.20

Polygonzug $P = (p_0, p_1, p_2)$ $p_i \in \mathbb{R}^d$

Länge $(P) = \sum_{i=1}^n |p_i - p_{i-1}|$ Summe der Kantenlängen

Lemma Wenn $c: I \rightarrow \mathbb{R}^d$ eine glatte Kurve und $(P_i^k)_{k \in \mathbb{N}}$ eine Folge von Polygonen mit interpolierenden

$$\lim_{k \rightarrow \infty} \max_{i \in \{1, n\}} |p_i^k - p_{i-1}^k| = 0$$

dann konvergiert $\text{Länge}(P_i^k) \xrightarrow{k \rightarrow \infty} \text{Länge}(c)$

Simpliziale Fläche S :

$$\text{Area}(S) := \sum_{\substack{t \text{ Dreiecke} \\ \text{von } S}} \text{area}(t)$$

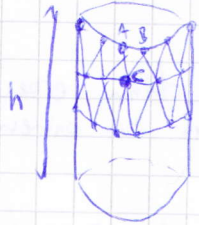
Frage: $\{S^k\}_{k \in \mathbb{N}}$ eine simpliziale Familie von simplizialen Flächen, die eine glatte Fläche $F: \Omega \rightarrow \mathbb{R}^d$ interpolieren.
Gilt:

$$\lim_{k \rightarrow \infty} \text{area}(S^k) = \text{area } F, \text{ wenn } \lim_{k \rightarrow \infty} \max_{\substack{t \text{ Dreieck} \\ \text{von } S^k}} \text{Umkreisradius}(t) = 0$$



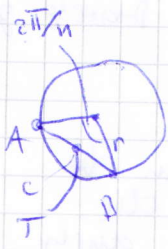
~~Lemma~~ ~~Umlaufzeit~~

Lehrsatz von Schwarz $L(m, n)$



m Punkte auf jedem Kreis, die Kreise sind äquidistant
 m Punkte ~~auf~~ in vertikaler Richtung

$$\text{area}(\text{Zylinder}) = 2\pi r \cdot h$$



$$\text{Länge der Kante } T: T = (A, B) = 2r \cdot \sin \frac{\pi}{n}$$

Zwischenrechnung: Wir betrachten die Grenzwerte

$$\lim_{n \rightarrow \infty} 2n \cdot \sin \frac{\pi}{n} = 2\pi$$

$2 \sin \frac{\pi}{n}$ ist Länge von T für Einheitskreis. n solcher Kanten T beschreiben n -Gon. Für $n \rightarrow \infty$ approximieren die n -Gone den Kreis und nach Lemma konvergieren die Gesamtlängen der n -Gone gegen den Kreisumfang.

Es gilt ~~also~~ $\cos \alpha = \cos^2 \frac{\alpha}{2} - \sin^2 \frac{\alpha}{2}$ (but Formelsammlung).
Damit gilt

$$\lim_{n \rightarrow \infty} 1 - \cos \frac{\pi}{n} = \lim_{n \rightarrow \infty} 1 - \cos^2 \frac{\pi}{2n} + \sin^2 \frac{\pi}{2n} = \lim_{n \rightarrow \infty} 2 \sin^2 \frac{\pi}{2n} \\ = \lim_{n \rightarrow \infty} 2 \frac{\pi^2}{n^2}$$

$c = (r, \frac{h}{n})$ in x, z -Koordinaten

$$\text{Höhe } h_c = \text{Abstand}(c, (\cos \frac{\pi}{n}, 0)) = \sqrt{\frac{h^2}{n^2} + r^2 (1 - \cos^2 \frac{\pi}{n})} \\ = \left| \begin{pmatrix} r \\ \frac{h}{n} \end{pmatrix} - \begin{pmatrix} r \cos \frac{\pi}{n} \\ 0 \end{pmatrix} \right|$$

$$\lim_{m, n \rightarrow \infty} \text{area}(L(m, n)) \stackrel{m, n \rightarrow \infty}{=} 2(m-1)n \cdot r \sin \frac{\pi}{n} \sqrt{\frac{h^2}{n^2} + r^2 (1 - \cos^2 \frac{\pi}{n})} \\ = 2\pi r \sqrt{\frac{h^2 (m-1)^2}{n^2} + (m-1)^2 r^2 (1 - \cos^2 \frac{\pi}{n})} \\ \stackrel{m, n \rightarrow \infty}{=} \lim_{m, n \rightarrow \infty} 2\pi r \sqrt{h^2 + r^2 \pi^4 \cdot \left(\frac{(m-1)^2}{n^2}\right)}$$

Der Grenzwert hängt ab, wie m, n relativ zueinander gegen ∞ streben.

In Abhängigkeit vom Grenzwert $\frac{m-1}{n^2}$ erhalten wir

$$\frac{m-1}{n^2} \rightarrow \begin{cases} 0 \Rightarrow \text{area}(L(m, n)) \rightarrow \text{area}(\text{Zylinder}) \\ > 0 \text{ bzw. beliebig groß} \Rightarrow \lim \text{area}(L(m, n)) \text{ kann} \\ \text{jeden beliebig großen Grenzwert annehmen} \end{cases}$$

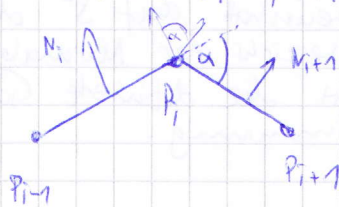
Wenn n nicht schnell genug wächst im Vergleich zu m , dann erzeugt das wachsende m immer mehr Faltungen, die dann zu einem beliebig großen Flächeninhalt führen.

Bemerkung: Wenn $m, n \rightarrow \infty$, dann gehen die Umhänge der Spirale gegen 0.

Satz: Wenn zusätzlich die Flächennormalen der Laken gegen die Flächennormalen der Grenzfläche konvergieren (also die Faltung mehr konvergenz gehalten wird), dann erzielt man auch Konvergenz der Flächeninhalte.

Krümmungslegende

Sei $P = (p_0, \dots, p_i, p_{i+1}, \dots)$ ein Polygon.



Definition: Der äußere Winkel α wird als Krümmung des Polygons an der Stelle p_i bezeichnet.

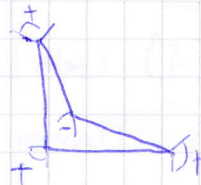
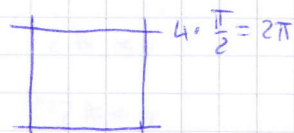
Lemma: α ist auch der Winkel der Normalenvektoren der beiden an p_i anliegenden Kanten

Lemma: Sei P ein geschlossenes Polygon. Dann gilt für die Gesamtkrümmung

$$K(P) = \sum_{i=0}^{n-1} K(p_i) = 2\pi$$

Beweis: Normale macht eine volle 2π Drehung.

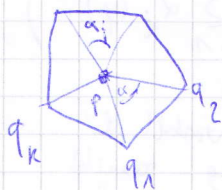
- einfach geschlossen: Polygon hat keine Selbstüberschneidungen
- muss nicht notwendig konvex sein, wenn man mit negativen Drehwinkel arbeitet



16.12.2016

Gauß-Krümmung von simplizialen Flächen

Vorlesung 18



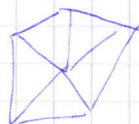
α Winkel bei p der angrenzenden Dreiecke "vertex angle"

$$K(p) = 2\pi - \sum_{i=1}^k \alpha_i$$

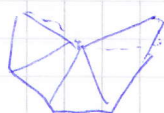
"Winkeldefizit"



$K > 0$ Kegelsituation "sphärische Ecke"



$K = 0$ "flach"



$K < 0$ sattelartig "hyperbolische" Ecke

Lemma: Wenn $K(p) = 0$, dann muss $\text{star}(p)$ nicht notwendig in $\mathbb{R}^3$ flach sein, allerdings lässt sich $\text{star}(p)$ isometrisch in die Ebene auffallen.

Hintergrund der Definition



g
Gauß-
Abbildung



Flächeninhalt auf S^2 der
Drehsektor der Normalen ist
exakt die definierte Gauß-
Krümmung.

Beweis: Mit Hilfe von sphärischer Trigonometrie ist der sphärische
Flächeninhalt des Polygons auf der Kugel $= 2\pi - \sum \alpha_i$.

Satz (diskrete Version von Gauß-Bonnet): Sei S eine geschlossene simpliciale
Fläche, dann gilt für die totale Gaußkrümmung

$$K(S) = 2\pi \chi(S)$$

Definition: Totale Gaußkrümmung von S

$$K(S) = \sum_{\substack{p \text{ Vertex} \\ \text{von } S}} K(p)$$

$$S^{(0)} = \{0\text{-Simplizes von } S\} \text{ (Vertex)}$$

$$v = \# S^{(0)}$$

$$S^{(1)} = \{1\text{-Simplizes von } S\} \text{ (Kanten)}$$

$$e = \# S^{(1)}$$

$$f = \# S^{(2)}$$

$$S^{(i)} = \text{Menge der } i\text{-Simplizes}$$

Beweis von Satz: $K(S) = \sum_{p \in S^{(0)}} K(p) = \sum_{p \in S^{(0)}} \left(2\pi - \sum_{i=1}^k \alpha_i(p) \right)$

jeder Vertexwinkel der
Triangulierung taucht
genau einmal auf

$$= 2\pi \cdot v - \pi \cdot f \quad 1+0$$

$$= 2\pi \cdot v - 2\pi f - 2\pi e$$

$$\text{mit } 3f - 2e = 0$$

$$= 2\pi \chi(S)$$

Die Gesamtkrümmung $K(S)$ hängt ausschließlich von der
Topologie der Fläche ab.



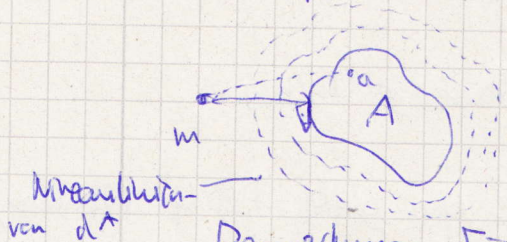
$$K(p) = 360^\circ - 3 \cdot 90^\circ = \frac{\pi}{2}$$

$$K(\text{cube}) = 8 \cdot \frac{\pi}{2} = 4\pi$$

Abstand zwischen Punkt Mengen (in $\mathbb{R}^d$)

Definition: Sei $A \subset \mathbb{R}^d$ nicht leer, $m \in \mathbb{R}^d$. Dann definieren wir den Abstand

$$d(m, A) := \inf_{a \in A} d(m, a)$$



Beimerkung: Für eine feste Menge A liefert dieser Abstand eine Abstandsfunktion

$$d^A: \mathbb{R}^d \rightarrow \mathbb{R} \\ m \mapsto d^A(m) := d(m, A)$$

Definition: Eine ϵ -Umgebung von A

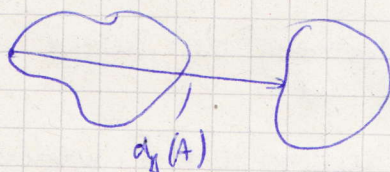
$$A_\epsilon := \{m \in \mathbb{R}^d \mid d^A(m) < \epsilon\}$$

Definition: Seien A und B zwei Mengen in $\mathbb{R}^d$, dann

$$d(B, A) := \inf_{b \in B} \left(\inf_{a \in A} d(b, a) \right) = d(A, B)$$

Kleinsten Abstand zwischen Punktgruppen von A und B .
Nachteil: kein Maß für weiter weg entfernten Punkte

Definition: ~~$d_B(A)$~~ $d_B(A) = \sup_{b \in B} (d(b, A))$



Abstand noch nicht
symmetrisch
"einseitige Hausdorff Abstand"

Definition: $d^H(A, B) = \max(d_B(A), d_A(B))$

Verfeinerungsverfahren (Refinement)

4-12-11

Wir wollen: Subdivisionsverfahren $\Rightarrow$ globale Verfahren,
die die gesamte Geometrie verändern

Vorlesung
19

Jetzt: lokale Verfahren: - 1-4 Split
- Kravars Bisektionsalgorithmus

Gegeben: Simpliciale Fläche M_h , die eine glatte Fläche (Funktion) approximiert.

Ziel: Verfeinere M_h soweit, dass irgendein Fehler kleiner wird.

Wir studieren jetzt Möglichkeiten, wie ein Dreiecksnetz lokal verfeinert werden kann.

1-4 Split mit Rot-Grün-Abschluss

Die zu verfeinernden Dreiecke werden wie gezeigt in vier gleichgroße Dreiecke unterteilt



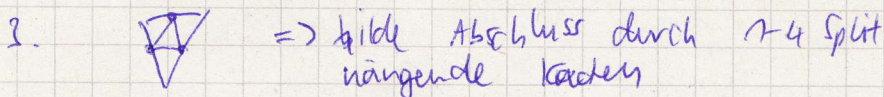
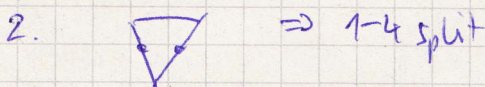
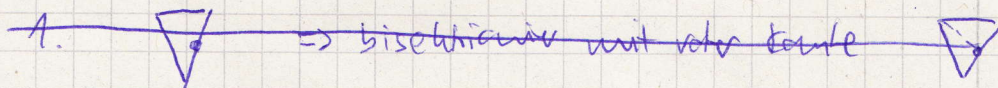
hängender
Knoten

Dies ist ~~so~~
kein Simplicialkomplex

1) Sei $S \subset M_h^{(2)}$ eine Teilmenge der Dreiecke von M_h , die als zu verfeinern markiert sind.

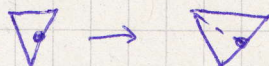
2) Die zu verfeinernden Dreiecke $t \in S$ werden wie gezeigt in vier gleichgroße Dreiecke unterteilt.

3) Wir erhalten Dreiecke mit



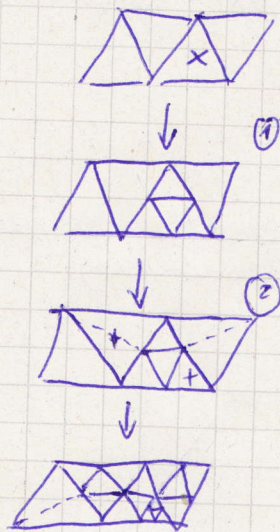
4) Wenn Sei $S = \{\text{Dreiecke mit hängenden Knoten}\}$. Wenn $S \neq \emptyset$
goto 3. oder 3

5) Jetzt sind nur noch Dreiecke mit max. 1 hängenden Knoten,
die dann mit roter Kante bisektiert werden



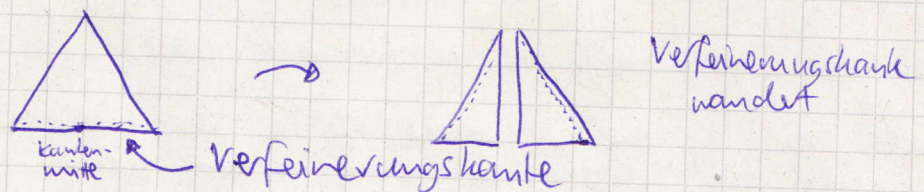
Wenn in einem Nachfolgenden Schritt M_n nochmals verfeinert werden soll, dann werden in einem zusätzlichen 0-ten Schritt alle roten kanten herausgenommen.

Ergebnis: Anwendung von 0. - 5. führt zu 1-4



Rivara-Refinierungsverfahren

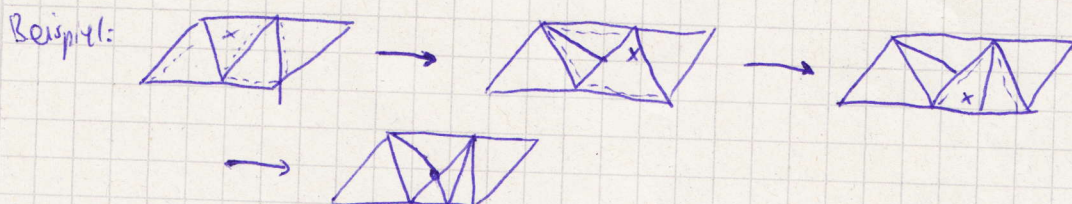
Gegeben: Simpliciale Fläche M_n . Zusätzlich wird in jedem Dreieck eine kante als ausgezeichnet markiert als sogenannte Verfeinerungskante.



Regel: Jedes Dreieck wird durch Disektion der Verfeinerungskante aus dem Schema unterteilt.

• nach 2 Schritten: alle Kantenmitten eingeholt

• nach 3 Schritten: Wir erhalten eine 1-4 Schritt



Rivara-Algorithmus: Gegeben eine simpliciale Fläche M_n , Sei der jedes Dreieck eine ausgezeichnete Verfeinerungskante besitzt. Ferner sei S eine Menge von einander verfeinerten Dreiecken

- 1) Jedes Dreieck $t \in S$ wird gemäß Rivara-Schnitt unterteilt



- 2) Sei S die Menge der Dreiecke mit hängenden Enden

- 3) Wenn $S \neq \emptyset$ goto 1

Unter ungünstigen Umständen kann die einfache Auswertung des Rivara-Verfahrens zu einer globalen Unterteilung führen.

Satz: Der Rivara-Algorithmus terminiert spätestens nachdem alle Dreiecke zweimal verfeinert wurden.

Beweis: Um das anfangs selektierte Gebiet S wird im Verfahren am Rand Verfeinerungskanten aufgesaut, die einen Wiedereintritt in ein schon verfeinertes Gebiet verhindern.

In der Praxis bleibt das Verfahren recht lokal.

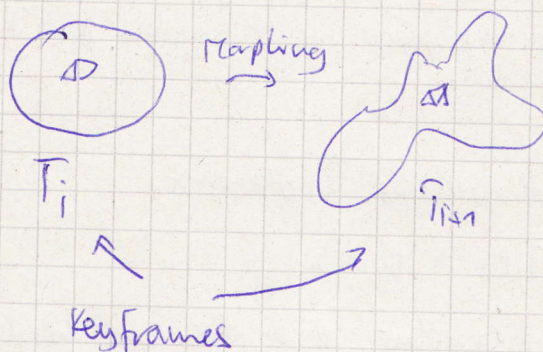
Satz: Sei M_n eine geschlossene Fläche ohne Rand. Dann kann man alle Dreiecke zu benachbarten Paaren verknüpfen und so die gemeinsame Kante die Verfeinerungskante beide Dreiecke liegen

th. Sci Vis
6.1.11

Vorlesung 20

Animation von simplizialen Flächen $F_i, i \in \{1, \dots, N\}$
Kontinuierliche Interpolation zwischen T_i und T_{i+1}

$$T(t) = (1-t)T_i + tT_{i+1}$$



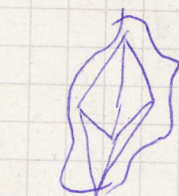
Interpolation dieser Art ist klar, wenn alle T_i eine paarweise simplizial isomorphe Triangulierung besitzen.
Der kombinatorische Simplizialkomplex ist identisch für alle T_i und die geometrische Position der Punkte ist unterschiedlich.

$$T_i = (K, P_i) \Rightarrow T_t = (K, (1-t)P_i + tP_{i+1}) \quad (*)$$

Problem hierbei: keine lokalen Verfeinerungen erlaubt.

Idee: Verwende eine hierarchische Geometriedarstellung für jede T_i .

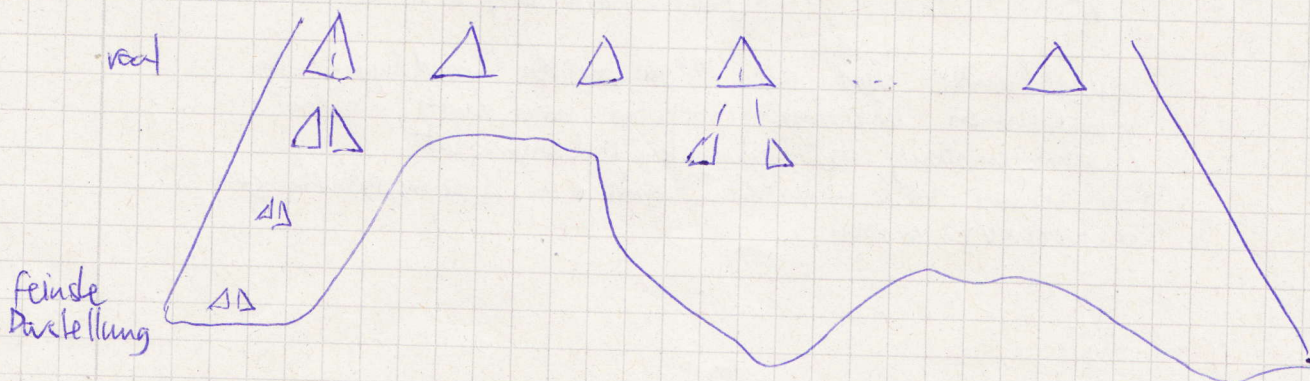
Hierarchie H:



1) H hat auf dem obersten Level "root level" eine Menge von "root triangles", die eine simpliziale Fläche beschreiben. Das root level entspricht der größten Darstellung.

2) Durch lokale Verfeinerung werden weitere Level erzeugt.

3) Die Blätter der Hierarchie formen die feinste Auflösung



Interpolationsbedingung:

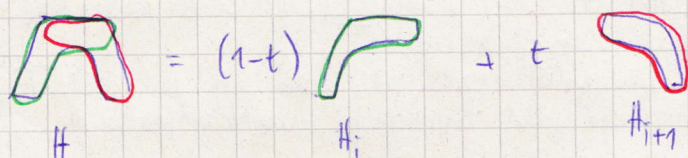
und die Verteilung der Verfeinerungskanten im Leaf-Level ist ~~essentiell~~ ^{isotopisch}

- A) Die Root-levels der T_i sind nach (*) simplizial isomorph.
D.h. auf dem größten Niveau kann mit (*) ganz einfach interpoliert werden.
- B) Jede Keyframehierarchie H_i wurde mit Hilfe von Rimba lokal verfeinert.

Dann existiert eine kontinuierliche Interpolation

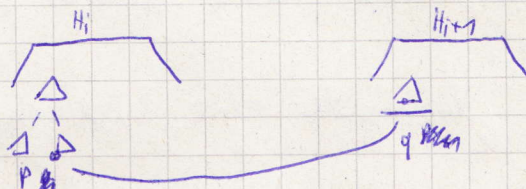
$$H(t) = (1-t) H_i + t H_{i+1}$$

Und zweitens die Hierarchie H ist die einfache Vereinigung der Bäume $H_i + H_{i+1}$

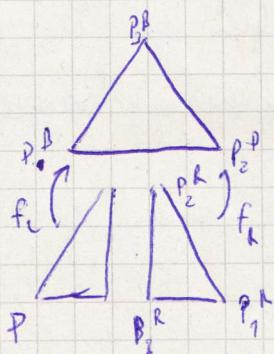


Warum besteht H nur aus der einfachen Vereinigung der Bäume H_i und H_{i+1} ?

Spezialfall (einfach):



Der Punkt p ~~hat~~ in H_i hat in H_{i+1} einen eindeutigen Korrespondenzpunkt q



Lineare Abbildung

$$f_L \begin{pmatrix} b_1^L \\ b_2^L \\ b_3^L \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1/2 & 0 & 1 \\ 1/2 & 0 & 0 \end{pmatrix} \begin{pmatrix} b_1^L \\ b_2^L \\ b_3^L \end{pmatrix} = \begin{pmatrix} b_2^L \\ b_1^L + b_3^L \\ b_2^L \end{pmatrix}$$

$$f_R \begin{pmatrix} b_1^R \\ b_2^R \\ b_3^R \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 1/2 & 0 & 0 \\ 1/2 & 1 & 0 \end{pmatrix} \begin{pmatrix} b_1^R \\ b_2^R \\ b_3^R \end{pmatrix}$$

Neue Punkte ~~sind~~ ^{wird} die Langzeithierarchien koordinaten im gemeinsamen überbereich beachtet, und durch diesen Zusammenhang der Langzeithierarchien koordinaten konstruiert
Für tiefere Verfeinerungen müssen Übergangsabbildungen konstruiert werden

Sei V

Interpolation von adaptiv verfeinerten Geometrien

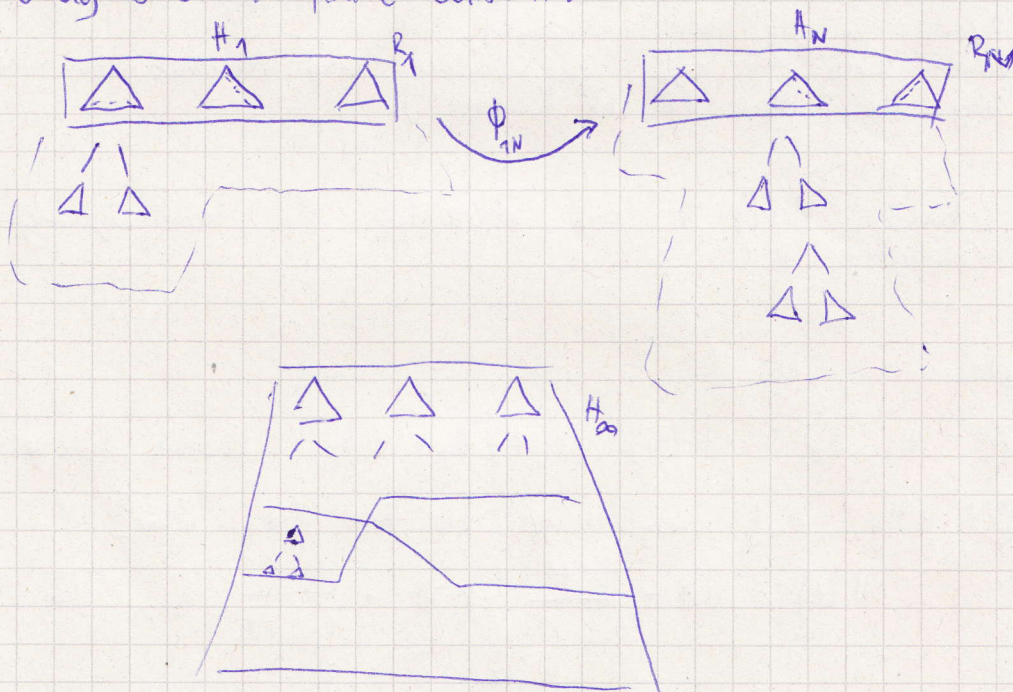
13.1.2011

Interpolationsbedingungen: Sei $\mathcal{H} = \{\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_N\}$ eine Familie von lokal verfeinerten simplizialen Flächen mit Verfeinerungshierarchie:

Vorlesung 21

- 1) Die jeweiligen Flächen $R_1, \dots, R_N$ der "Ratendreiecke" sind V simplicial isomorph mit Bijektionen $\Phi_{ij}: R_i \rightarrow R_j$
- 2) In jeder Ratfläche R_i hat jedes Dreieck eine ausgezeichnete Verfeinerungskante und die Auswahlen sind kompatibel mit Φ_{ij}
- 3) Jede Hierarchie wurde mit Rivara erzeugt

Theorem 1: Wenn eine Familie von Hierarchien $\mathcal{H} = \{\mathcal{H}_1, \mathcal{H}_2, \dots\}$ die Interpolationsbedingungen erfüllt, dann ist jede Hierarchie ein Teil der unendlichen ~~Hierarchie~~ $\mathcal{H}_\infty$ die durch unendliche Verfeinerung einer Ratfläche entsteht.



Theorem: Sei $\mathcal{H}$ wie oben und $\{b_1, \dots, b_N\}$ eine Familie von stetigen (Gewichts)-Funktionen $b_i: [0, 1] \rightarrow \mathbb{R}$ dann

$$\mathcal{H}(t) = \sum_{i=1}^N b_i(t) \mathcal{H}_i$$

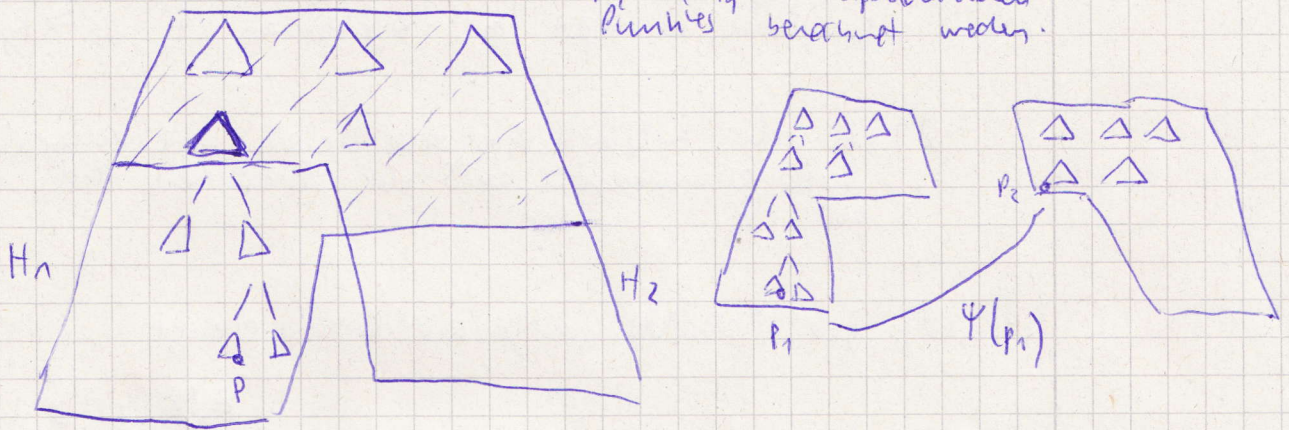
eine sich kontinuierlich verändernde Hierarchie, insbesondere ist die Menge der Blätter von $\mathcal{H}(t)$ eine sich stetig verändernde simpliziale Fläche.

Beweis: ObdA behalte zwei Hierarchien $\mathcal{H}_1$ und $\mathcal{H}_2$

- a) im gemeinsamen Bereich lässt sich der Isomorphismus Φ_{12} der Ratenelemente zu einem Isomorphismus der gemeinsamen Bereiche kanonisch ausdehnen, insbesondere besteht dadurch

eine Bijektion der Vertices.

- b) In Bereichen wo (obdA) H_1 weiter referiert ist als H_2 , dort kann durch die Übergangsabbildung zu jedem Punkt von H_2 die korrespondierende Position in H_1 im gemeinsamen Ellipsoid berechnet werden, und damit die gemeinsame Position p_2 der zu p_1 in H_1 korrespondierenden Punkte berechnet werden.



- c) Damit haben wir für alle Vertices aus H_1 einen korrespondierenden Punkt in H_2 und umgekehrt. Punkt muss kein Vertex sein, sondern kann innerhalb von Dreiecken liegen (auf den "Rivara Punkten").

Bemerkung: $H(t)$ ist als Hierarchie die Vereinigung aller beteiligten H_i .

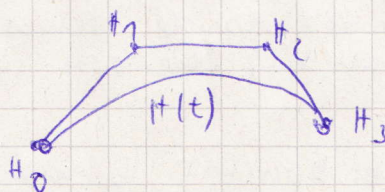
Beispiele:

- ① H_1, H_2 $H(t) = (1-t)H_1 + tH_2$ lineare Interpolation
- ② $\{H_1, \dots, H_n\}$ Keyframes zur Zeit $\{t_1, \dots, t_n\}$

$$H(t) = \left(1 - \frac{t-t_i}{t_{i+1}-t_i}\right) \cdot H_i + \frac{t-t_i}{t_{i+1}-t_i} H_{i+1} \quad \text{für } t \in [t_i, t_{i+1}]$$
- ③ Polynomiale Interpolation $\rightarrow$ siehe Übung
- ④ Bezier-Interpolation: Interpretiere alle H_i als Kontrollpunkte, d.h. Kontrollhierarchien

$$H(t) = \sum_{i=0}^n B_i^n(t) H_i \quad t \in [0, 1]$$

Bernstein polynome



Sei Vis

Dünne Schalen

Vorlesung 22

Formraum

18.1.2011

- polyedrische Flächen in $\mathbb{R}^3$
- variere die Abstände der Ecken im Raum, aber behalte die Konnektivität bei
- Die Menge aller so generierten "Flächen" bildet Konfigurationsraum, den wir den Formraum $\mathcal{N}$ nennen

$$\mathcal{N} = \mathbb{R}^{3n}$$

Funktion $E: \mathcal{N} \rightarrow \mathbb{R}$ die Energie

Beispiel Federenergie: $E_{\text{Feder}}(x) = \frac{1}{2} \sum e l_e^2$

Entspricht der potentiellen Energie eines Systems (Feder / Federn mit verschiedenen Ruhelängen)

Aufgabe: Berechne die Ableitungen / Gradienten

Probleme: • Berechne Minimum der Energien unter Nebenbedingungen

• Berechnung (Lösung) einer Bewegungsgleichung

$$\ddot{x}(t) = D(x(t)) + \text{grad}_{x(t)} E = F_{\text{Feder}}(x(t))$$

Dünne Dünne Schalen

Die Energie für dünne Schalen misst die Deformation einer Fläche von einem Referenzstand. Zwei Teile:

- Dehnungsenergie - Änderung der Metrik
- Biegeenergie - Änderung der Krümmung

$$E_D(x) = \lambda E_D(x) + \mu E_D(x)$$

(Materialkonstanten)

$$E_D(x) = E_L(x) + E_A(x)$$

$$E_L(x) = \frac{1}{2} \sum_i \frac{1}{A_i} \left(l_i - \hat{l}_i \right)^2$$

$$E_A(x) = \frac{1}{2} \sum_i \frac{1}{A_i} \left(A - \hat{A} \right)^2$$

Die Dehnungsenergie ist invariant unter Isometrien (z.B. Papier rollen, Wälzende Fahne usw.)

$$E_D(x) = \frac{1}{2} \sum_i \frac{l_i^2}{A_i} \left(\theta_i - \hat{\theta}_i \right)^2$$

Gesamtenergie ist invariant bei starren Transformationen.

Sci Vii

25.1.2011

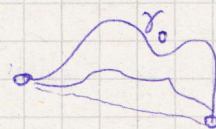
Vorlesung 23

$$\gamma_0: J = [a, b] \rightarrow \mathbb{R}^2 \quad s \in \mathbb{R}$$

$\gamma(a)$ und $\gamma(b)$ fest

Finde eine Kurve γ , die die Energie

$$E(\gamma) = \int |\gamma'(t)| dt \quad \text{Länge von } \gamma \text{ minimiert}$$

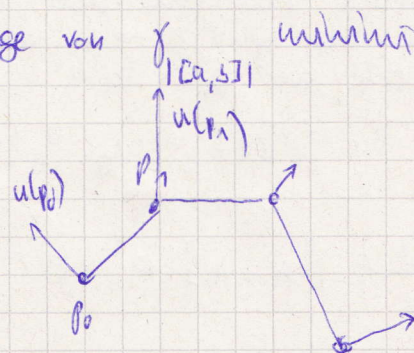


Finite Elemente

Sei γ eine polyedrische Kurve

Stückweise Funktion auf γ_i

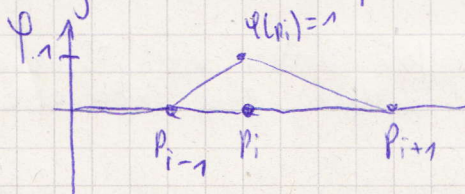
$$u: \underbrace{\text{spur } \gamma}_{\text{Punktmeng.}} \rightarrow \mathbb{R}$$



$u(p_i)$ abtragen auf die Normalen von p_i

stückweise linear (auf jeder Kante) gegeben durch Funktionswert $u(p) \in \mathbb{R}$ an allen Eckpunkten von γ

Zu jedem Punkt p konstruieren wir eine Hütchenfunktion



$$\varphi_i: \mathbb{R} \rightarrow \mathbb{R}$$

$$\varphi_i(p_i) = 1$$

$$\varphi_i(p_j) = 0 \quad \forall j \neq i$$

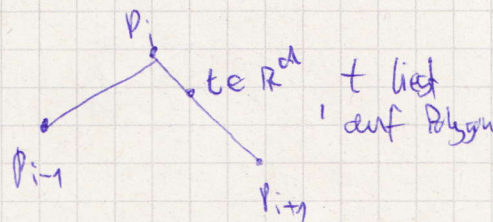
$$\text{bzw. } \varphi_i(p_j) = \delta_{ij}$$

$$\varphi(t) = \begin{cases} \frac{p_{i+1} - t}{p_{i+1} - p_i} & t \in [p_i, p_{i+1}] \\ \frac{t - p_{i-1}}{p_i - p_{i-1}} & t \in [p_{i-1}, p_i] \\ = 0 & t \notin [p_{i-1}, p_{i+1}] \end{cases}$$

$$p_i \in \mathbb{R}^d$$

$$\varphi(t) = \begin{cases} \frac{|p_{i+1} - t|}{|p_{i+1} - p_i|} \\ \frac{|t - p_{i-1}|}{|p_i - p_{i-1}|} \\ 0 \end{cases}$$

sonst



Für jeden Punkt p_i existiert eine Funktion $\varphi_i: \text{spur } \gamma \rightarrow \mathbb{R}$.

Bei n Punkten hat man $\{\varphi_0, \varphi_1, \dots, \varphi_n\}$ Funktionen.

Aus den Funktionen $\{\varphi_0, \dots, \varphi_n\}$ können wir jede stückweise lineare Funktion über $\text{spur } \gamma$ beschreiben.

Sei $u: \text{spur } \gamma \rightarrow \mathbb{R}$ stückweise linear mit Funktionswerten $u_i \in \mathbb{R}$ an p_i . Dann

$$u(t) = u_0 \varphi_0(t) + \dots + u_n \varphi_n(t) = \sum_{i=0}^n u_i \cdot \varphi_i(t)$$

$$t \in [p_0; p_1] \Rightarrow u(t) = u_0 \varphi_0(t) + u_1 \varphi_1(t) + \underbrace{0 + \dots + 0}_0$$

$$= u_0 \frac{|p_1 - t|}{|p_1 - p_0|} + u_1 \frac{|t - p_0|}{|p_1 - p_0|}$$

$$= \frac{u_0 |p_1 - t| + u_1 |t - p_0|}{|p_1 - p_0|}$$

$$= u_0(1-s) + u_1 \cdot s$$

$$\frac{|t - p_0|}{|p_1 - p_0|} = s$$

$$\frac{|p_1 - t|}{|p_1 - p_0|} = 1 - s$$

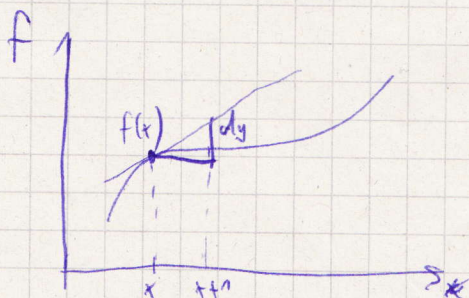
γ ein Polygon mit $n+1$ Punkten. Wir haben $n+1$ Funktionswerte $u_0, \dots, u_n \in \mathbb{R}$. Diese bestimmen eindeutig eine stückweise lineare Funktion

$$u: \text{spur } \gamma \rightarrow \mathbb{R}$$

$$u(t) = \sum_{i=0}^n u_i \cdot \varphi_i(t)$$

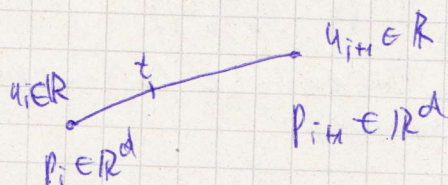
$$u \text{ ist identisch } 1 \ (u \equiv 1) \Rightarrow u(t) = \sum_{i=0}^n \varphi_i(t)$$

$$\begin{aligned} \text{Länge } \gamma &= \int_{\gamma} 1 \, dt = \sum_{i=1}^n \int_{p_{i-1}}^{p_i} 1 \, dt = \sum_{i=1}^n \int_0^1 |p_i - p_{i-1}| \, dt \\ &= \sum_{i=1}^n |p_i - p_{i-1}| \underbrace{\int_0^1 dt}_1 = \text{Gesamtlänge aller Kanten} \end{aligned}$$

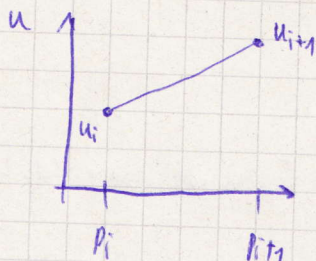


$f'(x)$ ist Höhe des Steigungsdreiecks der Tangente auf $f(x)$ mit Basis 1.

Betrachte die Kurve γ



Was ist die Ableitung von u auf der Kante p_i, p_{i+1} ?



$$u'(t) = \frac{u_{i+1} - u_i}{|p_{i+1} - p_i|}$$

Werttrage Konzept auf simpliziale Flächen.

Raum der stückweise linearen, stetigen Funktionen.
Sei M_h eine simpliziale Fläche mit Punkten $\{p_i\} \in \mathbb{R}^d$

$$u : M_h \rightarrow \mathbb{R}$$

$$S_h := \{ u : M_h \rightarrow \mathbb{R} \mid u|_{\Delta} \text{ linear auf jedem Dreieck, } u \in C(M) \text{ stetig auf } M \}$$

Vektorraum der stückweise linearen Funktionen hat Dimension $\#$ Eckpunkte, da aus jedem Eckpunkt die Funktionswert frei gewählt werden kann.
Die Basis ist $\{\varphi_0, \dots, \varphi_n\}$.

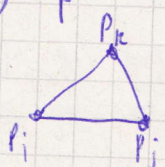
Satz: Jede Funktion $u \in S_h$ hat eine eindeutige Darstellung

$$u(p) = \sum_{i=0}^n u_i \cdot \varphi_i(p)$$

wo $u_i = u(p_i)$ der Wert von u an der Stelle p_i ist

$$1) \text{ Erwartung } u(p_i) = u_i$$

$$2) p \in \Delta(p_i, p_j, p_k)$$



$$u(p) = u_i \varphi_i(p) + u_j \varphi_j(p) + u_k \varphi_k(p)$$

$$\varphi_i + \varphi_j + \varphi_k \equiv 1 \text{ da linear und an } \Delta(p_i, p_j, p_k) \text{ drei Ecken 1.}$$

Sei $p \in \Delta(p_i, p_j, p_k)$

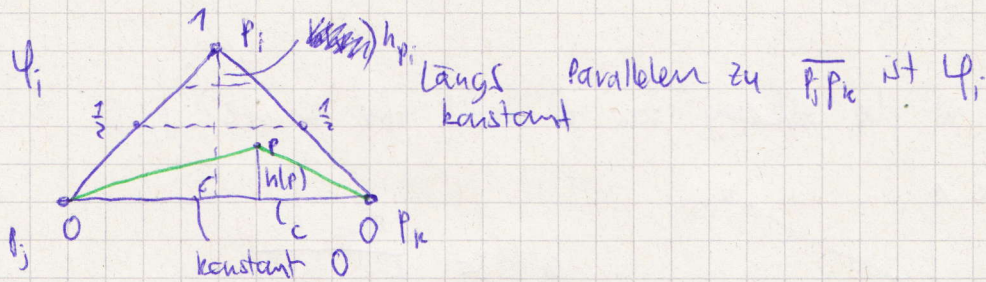
Sci Vis

Vorlesung
24

27.1.2021

$$\Rightarrow u(p) = \sum_{i=0}^n u_i \quad \varphi_i(p) = u_i \varphi_i(p) + u_j \varphi_j(p) + u_k \varphi_k(p)$$

hiermit kann $u(p)$ an allen Stellen $p \in \Delta$ ausgewertet werden, wenn wir $\varphi_i, \varphi_j, \varphi_k$ kennen.



Sei $h(p)$ die Höhe von p über der Grundlinie, d.h. $0 \leq h(p) \leq h_{p_i}$

$$(x) \quad \varphi_i(p) = \frac{h(p)}{h_{p_i}} = \frac{\frac{1}{2} h(p) \cdot |c|}{\frac{1}{2} h_{p_i} \cdot |c|} = \frac{\text{area}(p, p_j, p_k)}{\text{area } \Delta}$$

Damit einfache Formel um φ_i auszuwerten.

$$\varphi(p) = \varphi_i(p) + \varphi_j(p) + \varphi_k(p)$$

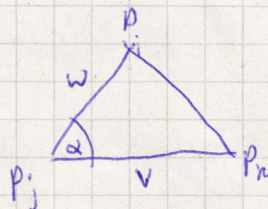
$$= \frac{\text{area}(p, p_j, p_k) + \text{area}(p, p_i, p_k) + \text{area}(p, p_i, p_j)}{\text{area } \Delta} = 1$$

Definition: Die Funktionswerte $(\varphi_i(p), \varphi_j(p), \varphi_k(p))$ heißen baryzentrische Koordinaten von p im Dreieck $\Delta(p_i, p_j, p_k)$.

- Koordinaten addieren sich zu 1
- sind positiv (≥ 0) im Dreieck
- lassen sich aus Formel (x) für das p berechnen

Schwerpunkt von Δ : $S(\Delta) = \frac{p_i + p_j + p_k}{3} =: p \Rightarrow$ baryzentrische Koordinaten $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$

Flächeninhalt :



$$\text{area } \Delta = \frac{1}{2} |(p_k - p_j) \times (p_i - p_j)| = \frac{1}{2} \sin \alpha \cdot |v| \cdot |w|$$

$$\langle v, w \rangle = \cos \alpha \cdot |v| \cdot |w|$$

Differenzieren

$$\varphi: \mathbb{R}^2 \rightarrow \mathbb{R}$$

$$\frac{d}{dx} \varphi = \varphi_x$$

$$\frac{d}{dy} \varphi = \varphi_y$$

$$\text{grad } \varphi = (\varphi_x, \varphi_y) \in \mathbb{R}^2$$

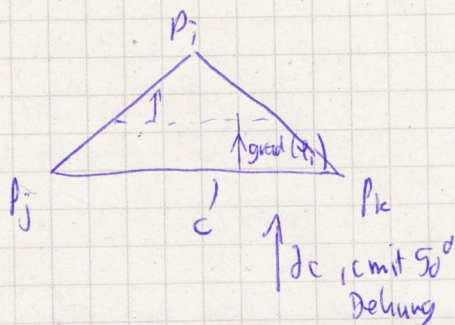
Vektor im Definitionsbereich

Eigenschaften:

- zeigt in Richtung der maximalen Funktionsänderung, d.h. $\text{grad } \varphi$ steht senkrecht auf Niveaulinien $\varphi = \text{konstant}$

- $|\text{grad } \varphi|$ misst Steigung, wenn man 1 in die Richtung von $\text{grad } \varphi$

Anwendung: Berechne $\text{grad } \varphi_i$ im Dreieck $\Delta(p_i, p_j, p_k)$



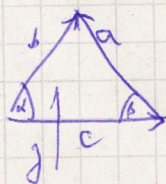
1. $\text{grad } \varphi_i$ senkrecht zur Grundlinie $p_j p_k$ und zeigt parallel zur Höhe

2. $|\text{grad } \varphi_i| = \frac{|\text{grad } \varphi_i|}{1} = \frac{1}{h_{p_i}}$
 $\Rightarrow |\text{grad } \varphi_i| = \frac{1}{h_{p_i}}$

$$\Rightarrow \text{grad } \varphi_i = \frac{1}{h_{p_i}} \frac{j_c}{|c|} \leftarrow \text{Einheitsvektor in Richtung der Höhe}$$

$$= \frac{1}{2 \text{area } \Delta} j_c = \frac{1}{2 \text{area } \Delta} (\cot \alpha \vec{a} + \cot \beta \vec{b})$$

Bemerkung:



Es gilt $j_c = \cot \alpha \vec{a} + \cot \beta \vec{b}$

Damit ist $\text{grad } \varphi_i$ bestimmt aus Größen des Dreiecks.

Zurück zu $\sum_{i=1}^n u_i \varphi_i(p)$

$$\nabla u(p) = \sum_{i=1}^n u_i \cdot \nabla \varphi_i(p)$$

da Gradient in die Summe gezogen werden kann und u_i konstant.

$$p \in \Delta(p_i, p_j, p_k) \quad \vec{a} = u_i \nabla \varphi_i(p) + u_j \nabla \varphi_j(p) + u_k \nabla \varphi_k(p)$$

$$= \frac{1}{2 \text{area } \Delta} (u_i j_{c_i} + u_j j_{c_j} + u_k j_{c_k})$$

∇u ist Vektorfeld auf M_n , dass auf jedem Dreieck durch einen konstanten Vektor repräsentiert wird. Vektorfeld ist notwendig un stetig längs Kanten von M_n .

Vorlesung 25
1.2.2011

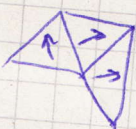
Formel: $\nabla u(p) = \sum_{i=0}^n u_i \nabla \varphi_i(p)$

$$\nabla \varphi_i|_T = \frac{1}{2 \text{area}(T)} \int_C$$

(Raum der PL-Funktionen)
stückweise linear

Raum der tangentialen Vektorfelder:

$$\Lambda^1 = \left\{ v \mid v|_T \text{ konstanter tangentialer Vektor auf jedem Dreieck } T \right\}$$

$$v \in \Lambda^1 =$$


Bemerkung: Wenn $u \in S_n(M_n) \Rightarrow \nabla u \in \Lambda^1(M_n)$

Frage: $v \in \Lambda^1$, gibt es eine Funktion $u \in S_n$ mit $\nabla u = v$?

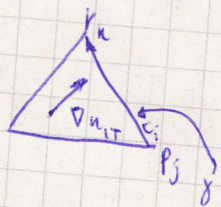
Potenzialfunktion Gradientenfeld

$$u(p) = \sum_{i=0}^n u_i \cdot \varphi_i(p)$$

Darstellung eines Dreiecks

$$= \frac{1}{2 \text{area}} \int_C (u_i - u_k) - \int_C (u_j - u_k)$$

Durch geeignete Wahl von $(u_i - u_k)$ und $(u_j - u_k)$ kann jeder Vektor in $\mathbb{R}^2$ dargestellt werden.
Damit finde ich zu jedem Vektor $v \in \Lambda^1(T)$ auf einem Dreieck T eine lineare Funktion $u \in S_n(T)$ mit $\nabla u = v$.



$$\begin{aligned} \langle C_i, \nabla u|_T \rangle &= \int_C \nabla u|_T = \int_0^1 \langle \nabla u|_T, \gamma'(t) \rangle dt \\ &= \langle \nabla u|_T, C_i \rangle \cdot \int_0^1 dt = a(p_k) - u(p_j) \end{aligned}$$

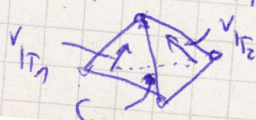
constant

$$\begin{aligned} \gamma(0) &= p_j \\ \gamma(1) &= p_k \\ \gamma(t) &= p_j + t(p_k - p_j) \\ \gamma'(t) &= p_k - p_j = C_i \end{aligned}$$

Bemerkung: Skalarprodukt eines Gradienten ∇u die Änderung der Funktion u längs C ist genau.

Korollar:

Satz: Ein Vektorfeld $v \in \Lambda^1(M_n)$ ist ein Gradientenfeld $\nabla u = v$ zu einer $u \in S_n(M_n)$ $\Leftrightarrow$ für jedes $C \in M_n$ gilt für v



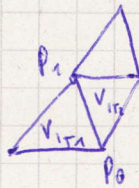
$$\begin{aligned} a) \langle C, v|_C \rangle &= \langle C, v|_{T_2} \rangle \quad (*) \\ b) \text{rot } v|_C &= 0 \end{aligned}$$

Beweis: $\Rightarrow$ schon gezeigt vorher also

$\Leftarrow$ Gegeben ein Vektorfeld $v \in \mathcal{X}^1(M_h)$ mit Eigenschaft (*) an jeder Kante erfüllt.

Konstruiere nun ein Potenzial $u \in S_h$ (mit $\nabla u = v$)

Beginne mit $u(p_0) = 0$ (oder beliebige Konstante)



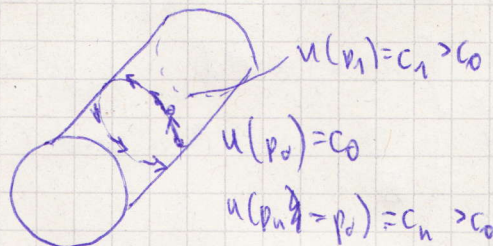
$$u(p_1) = u(p_0) + \langle \overline{p_1 p_0}, v_{12} \rangle$$

oder T_2 wegen Bedingung (*)

Auf diese Weise kann u an jedem Punkt p_i eindeutig bestimmt werden.

(In kurz: $u(p) = u(p_0) + \int_{p_0}^p v$)

Vorsicht:



v glattes Vektorfeld in $\mathbb{R}^2$ - $v \in \nabla u$ $u: \mathbb{R}^2 \rightarrow \mathbb{R} \Leftrightarrow \text{rot } v = 0$

Definition: Sei v ein Vektorfeld. Dann heißt

$$\text{rot } v|_c = \langle c, v_{12} \rangle - \langle c, v_{21} \rangle$$

die Rotation von v an Kante c .

Direkt-Energie auf $u \in S_h$

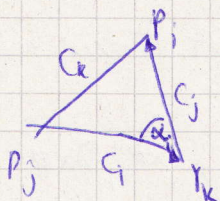
$$E_D(M_h) = \frac{1}{2} \int_{M_h} |\nabla u|^2$$

$E_D \gg 0$ für stark variablen in u (Kanten)
 $E_D = 0$ für u ist konstante Funktion

$$\frac{1}{2} \int_{M_h} |\nabla u|^2 = \frac{1}{2} \sum_{T \in \mathcal{T}_h} \int_T |\nabla u|^2 = \frac{1}{2} \sum_T \int_T \left\langle \sum_{i=1}^2 y_i \nabla \varphi_i, \sum_{j=1}^2 y_j \nabla \varphi_j \right\rangle$$

$$= \frac{1}{2} \sum_T \sum_{i,j=1}^2 y_i y_j \int_T \langle \nabla \varphi_i, \nabla \varphi_j \rangle$$

Zwischenrechnung: $\langle \nabla \varphi_i, \nabla \varphi_j \rangle = \frac{1}{(2\text{area})^2} \langle \vec{c}_i, \vec{c}_j \rangle$



$$= \frac{1}{(2\text{area})^2} \langle \vec{c}_i, \vec{c}_j \rangle$$

$$= \frac{1}{2\text{area}} \frac{\cos \alpha_{ij} |\vec{c}_i| |\vec{c}_j|}{\sin \alpha_{ij} |\vec{c}_i| |\vec{c}_j|}$$

$$= \frac{1}{2\text{area}} \cot \alpha_{ij}$$

$$\langle \vec{c}_i, \vec{c}_j \rangle = \cos \alpha_{ij} |\vec{c}_i| |\vec{c}_j|$$

$$\text{area} \sin \alpha_{ij} = \sin \alpha_{ij} |\vec{c}_i| |\vec{c}_j|$$

$$\star \int \langle \nabla \varphi_i, \nabla \varphi_j \rangle = \frac{1}{2} \cot \alpha_{ij}$$

$$E_D(u) = \frac{1}{4} \sum_i \sum_{j=1}^3 a_i u_j \cot \alpha_{ij}$$

$\star$ linear Wert

(für M_n ohne Rand)

Matrixschreibweise: Sei $u = (u_0, u_1, \dots, u_n)$. Sei S eine $(n+1) \times (n+1)$ Matrix mit

$$S_{ij} = \begin{cases} \cot \alpha_{ij} + \cot \beta_{ij} & p_i, p_j \text{ benachbart mit } i \neq j \\ 0 & \text{sonst} \end{cases}$$

$$S_{ii} = - \sum_{j \text{ benachbart}} S_{ij}$$

Dann gilt $E_D(u) = \frac{1}{4} u^T S u$

ist durch Kantenentwertung eines a_{ij} von Nachbardreieck entstanden.

