Seminar

Surface Reconstruction

„One Triangle at a Time“ -

Ein inkrementeller Algorithmus zur Oberflächenrekonstruktion

Vortrag im Seminar Surface Reconstruction von

Stefan Kleine
2kleine@informatik.uni-hamburg.de

Inhalt

Seminar Surface Reconstruction

Einleitung

Daniel Freedman

Lehrstuhl im Dept. Computer Science des
Rensselaer Politechnic Institute (RPI),
Troy, NY 12180, USA

Arbeitsgebiete

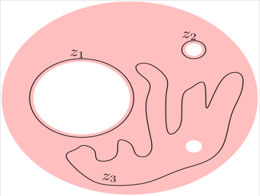
- Computer Vision

- Geometric Algorithms

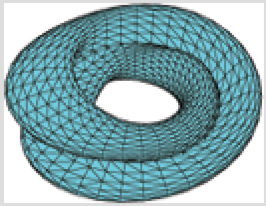
Einleitung

Daniel Freedman

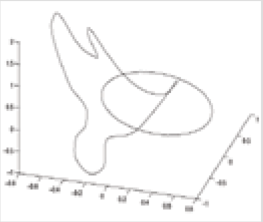
Forschungsprojekte im Bereich Geometrische Algorithmen



The Hardness of Finding the Optimal Homological Cycle

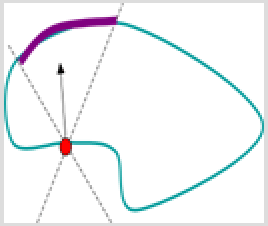


Incremental Surface Reconstruction



Finding Natural Generators for a Homology Group

Combinatorial Manifold Reconstruction



Provable Curve Reconstruction with a New Sampling Condition

Einleitung

Incremental Surface Reconstruction

SR klassisches Problem im Bereich Computational Geometry

Bekannte Verfahren benötigen in R^3 liegende Oberfläche

Vorgestellter inkrementeller Algorithmus erlaubt
Oberflächen beliebiger *Kodimension*

Dies erlaubt die Rekonstruktion von Oberflächen aus
der Klasse der **nicht-orientierbaren Oberflächen**

Andere Verfahren zur Oberflächenrekonstruktion

Boissant	„sculpting“ Delauny Triangulations
Hoppe et al.	a functions zero-level set = surface
Amenta and Bern	crust; power-crust („watertightness“)
Dey	co-cone idea
Boissant	erster inkrementeller Ansatz (1984)
Bernardini et al.	Ball-Pivoting Algorithmus
Gopi et al.	*computed normals* für inkrementelle Konstruktion der Oberfläche
.....	.....

Begriffe und Definitionen

Im Handout und/oder in diesem Vortrag beschriebene Begriffe:

(Prä-)Hilbertraum

Mannigfaltigkeit

Tangentialraum

simplicial complex

Dimensionsbegriff

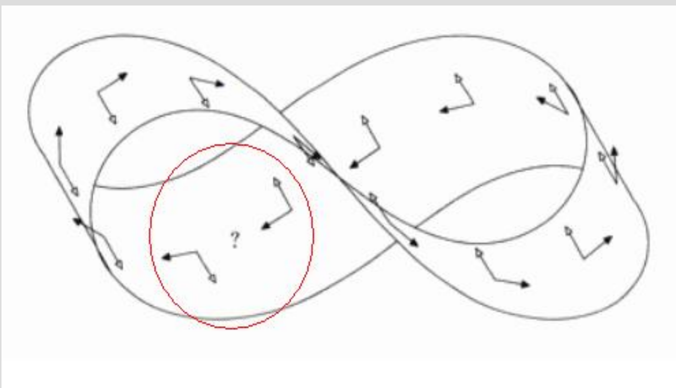
Codimension

Orthonormalbasis

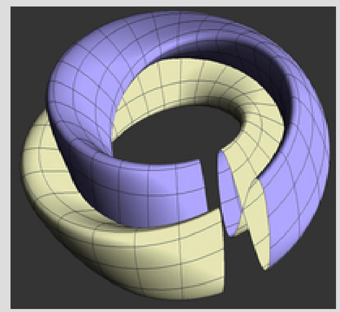
Orientierbarkeit

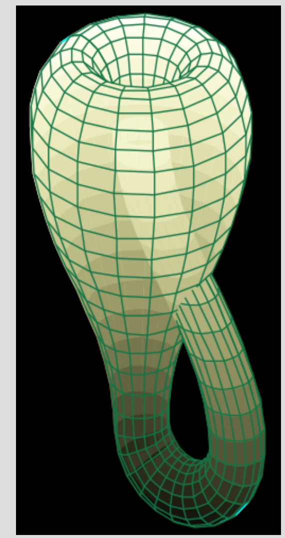
Homöomorphismus

Beispiele für nicht-orientierbare Flächen



Möbius-Band





Klein'sche Flasche

Eine Fläche F ist **orientierbar**, falls sie keinen Pfad enthält, auf dem sich die Orientierung umkehrt.

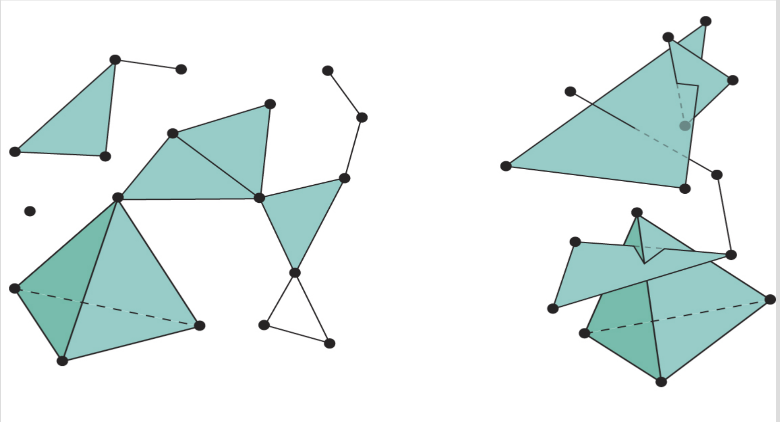
Andernfalls heisst F **nicht-orientierbar**.

Zum Begriff *simplicial complex*

Ein **simplicial complex** K ist eine Menge von Simplexen für die gilt:

1. Jedes *face* eines Simplex K liegt ebenfalls in K
2. Jede Intersektion zweier Simplexe $б_1$, $б_2$ aus K ist *face* von $б_1$ und $б_2$

Die *konvexe Hülle* einer nicht-leeren Teilmenge der n+1 Punkte, die ein n-Simplex definieren, nennt man **face** eines Simplex.



links: simplicial 3-complex

rechts: abstrakter simplicial complex

Begriff *(Prä-)Hilbertraum*

Ein **Prähilbertraum** ist ein normierter Vektorraum (über reellen oder komplexen Zahlen), d.h. dass mit Hilfe des Skalarprodukt eine Norm erzeugt werden kann. Ein normierter Raum heisst unitär, man in ihm ein Skalarprodukt einführen kann, das mit der Norm verknüpft ist.

Ein vollständiger, unitärer Raum heisst **Hilbertraum**, welcher eine Verallgemeinerung der euklidischen Geometrie auf einen unendlich-dimensionalen Raum darstellt.

Algorithmus – Einleitung

Besonderheit gegenüber anderen gängigen Verfahren:

Bezug auf *Mannigfaltigkeiten (manifolds)* und nicht auf den zugrundeliegenden (topologischen) **Raum**

Inkrementelle Eigenschaft: bei jeder Iteration wird der Fläche ein einzelnes Dreieck hinzugefügt

→ Oberfläche muss nicht in R^3 liegen, sondern darf in einem beliebigen *Hilbertraum D* eingebettet sein

→ Rekonstruktion von Oberflächen ohne Begrenzung (u.a. *nicht-orientierbare* Oberflächen)

Ziele des inkrementellen Algorithmus

Gegeben: Menge von (unorganisierten) Punktdaten einer (unbekannten) Oberfläche

Ziel: Konstruktion einer zweiten Oberfläche, die *homöomorph* zur ersten ist

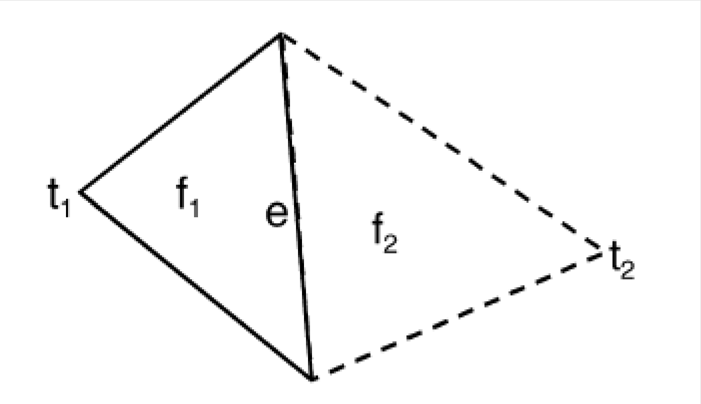
Teilziel: geometrisch möglichst gute Annäherung an die originale Oberfläche

Algorithmus – Inkrementelle Rekonstruktion

Aufbau der Fläche durch Hinzufügen einzelner Simplexe

Auswahl der *dangling edge e* des aktuellen Simplex:
die Kante genau einer Fläche f_1

Ziel ist es, die zweite Fläche f_2 zu finden



Algorithmus abstrakt (1/2)

```
M = {xi}i=1 bis n

RECONSTRUCT(M)
  INITIALIZE(M)
  do
    b = barycenter(e)
    FILTERING
    PROJECTION
    dmin = inf
    for t2 ∈ Γ
       f2_pot = e U {t2}
       if ||xt2 - b|| < dmin
          if !INTERSECTION
             dmin = ||xt2 - b||
             f2 = f2_pot
    K = K U {s: s C f2}
    draw(e, f1, t1)
  until there is no e
return K
```

1. Initialisierung

2. Filterung

3. Projektion

4. Prüfung und Auswahl von f_2

Algorithmus abstrakt (2/2)

Details des Hauptalgorithmus

RECONSTRUCT($\{x_i\}_{i=1}^n$)
  $(e,\ f_1,\ t_1,\ K) =$ INITIALIZE($\{x_i\}_{i=1}^n$)
  do
    $b = \text{barycentre}(e)$
    $(\Gamma,\ L) =$ FILTER($\{x_i\}_{i=1}^n, t_1, b, e, K$)
    $\{u_i\}_{i \in \Gamma} =$ PROJECT($\{x_i\}_{i=1}^n, e, t_1, b, \Gamma$)
    $d_{min} = \infty$
    for all $t_2 \in \Gamma$
      $f_2^{pot} = e \cup \{t_2\}$
      if $\|x_{t_2} - b\| < d_{min}$
        if INTERSECT($f_2^{pot}, L, \{u_j\}_{j \in \Gamma}$) $=$ False
          $d_{min} = \|x_{t_2} - b\|$
          $f_2 = f_2^{pot}$
    $K = K \cup \{\sigma : \sigma \subset f_2\}$
    draw $(e,\ f_1,\ t_1)$ from K with #cofaces$(e) = 1$
  until there is no e
  return K

t_1, t_2	vertices	0-simplex
e	dangling edge	1-simplex
f_1, f_2	faces	2-simplices
K		(current) simplicial complex
Γ	Teilmenge samples	simplicial complex
L	subcomplex in Nachbarschaft zu e	simplicial complex

Seminar Surface Reconstruction

Algorithmus - 1. Initialisierung

$\textsc{Initialize}(\{x_i\}_{i=1}^{n})$
$\quad q = 1$
$\quad r = \operatorname{argmin}_{j \neq q} \|x_j - x_q\|$
$\quad J = \{j \neq q, r:\ \text{angle}(x_j - x_q, x_r - x_q) \geqslant \pi/4\}$
$\quad s = \operatorname{argmin}_{j \in J} \|x_j - x_q\|$
$\quad f = \{q, r, s\}$
$\quad e = \{r, s\}$
$\quad t_1 = s$
$\quad K = K \cup \{\sigma : \sigma \subset f_1\}$
$\quad \text{return } (e, f_1, t_1, K)$

Auswahl *initial simplex* und *dangling edge e*

- **1. Knoten:** zufälliger Knoten q

- **2. Knoten:** der zu q nächste Knoten des Simplex

- **3. Knoten:** der zu q zweitnächste Knoten des Simplex, so dass die Vektoren x_r-x_q und x_s-x_q einen Winkel >= Pi/4 bilden

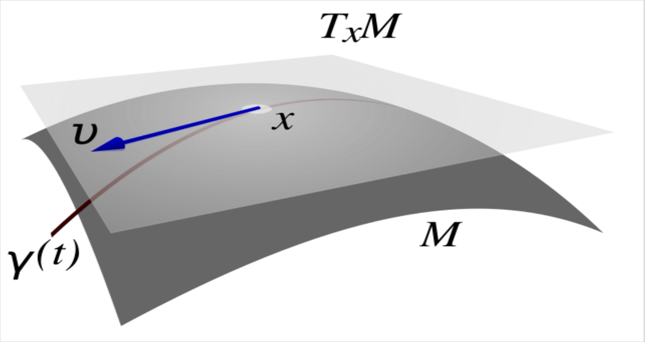
→ wir erhalten (e, f1, f2, K) K ist *simplicial complex*

Algorithmus - 2. Filterung (1/3)

Wie kann die Auswahl von Dreiecken verhindert werden, die zu einer Selbst-Intersektion der Fläche führen?

Ansatz: Alle Berechnungen werden lokal in einer Fläche durchgeführt, die benachbarte *Tangentialräume* der *manifolds* approximiert

Ein *Tangentialraum* T_xM ist ein Vektorraum, der in einem Punkt x ein differenzierbares *manifold* M 'berührt'



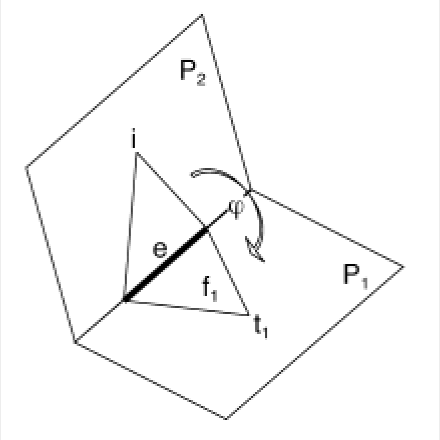
Algorithmus - 2. Filterung (2/3)

Auffinden der Punkte in Nachbarschaft von e

statt Abstandskriterium: *tangentialraum*-basierte Methode:

Approximation der Tangentialräume von Punkten nahe der *Kante e* der Fläche P_1, die durch *e's coface* f_1 verläuft

Knoten i und Kante e bilden P_2, wenn Winkel



zwischen P_1 und P_2 > Pi-ε mit ε = Pi/4

<u>Motivation für Filterung:</u>

War die Oberfläche M glatt und gut gesampelt, existieren mehrere Punkte i in Nachbarschaft zur Kante e

Algorithmus - 2. Filterung (3/3)

sind die Punkte gefiltert, ist es leicht die Simplexe zu filtern:

K = *initialial simplicial complex*
Γ = Menge der gefilterten Knoten, dann ist
L = { σ E K: *vert*{σ} C Γ } der gefilterte Subcomplex

→ L enthält genau die Simplexe, die den 'Winkeltest' passiert haben

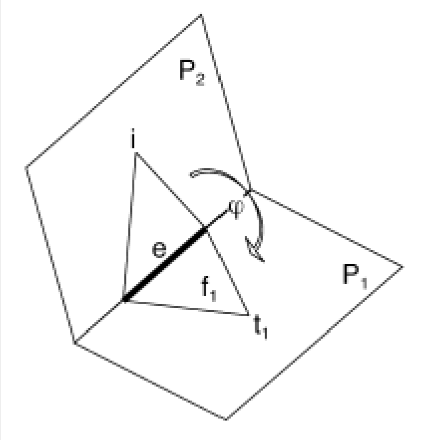
Algorithmus - 3. Projektion

Ziel ist das **Auffinden von f_2** für *dangling edge e* zur inkrementellen Triangulation (nächster Abschnitt)

Projektion der Punkte in Γ und Simplexe in L:

wir behalten *abstract simplicial (sub)complex* und erhalten neuen *geometric simplicial complex*

→ alle Simplexe schneiden Fläche in „natürlicher“ Weise oder gar nicht



Algorithmus - 4. Inkrementelle Triangulation

Hauptziel des Algorithmus: passende Auswahl von f_2 für Kante e

Finden eines Knoten t_2 aus Γ so, dass f_2 2-Simplex aus e und t_2:

Sei L' = L U { σ: σ C f_2 }

Mappen aller Knoten i in L' auf korresponierenden Punkt u_i in einer Ebene G'

Schlüsselkriterium für potentielle f_2:

G' muss *geometric simplicial complex* sein, d.h.

f2 darf die bereits existierenden Simplexe in der Nachbarschaft von e nur in „natürlicher“ Weise schneiden

Algorithmus - 4. Inkrementelle Triangulation

Frage: Existiert coface f_2 immer?

Theorem: Enthält L mindestens einen Knoten, der nicht auf der Linie durch e liegt, dann existiert ein zweites *coface* f_2, das L hinzugefügt werden kann und L bleibt ein *simplicial complex*.

→ Da das Theorem beweisbar ist, existiert immer ein f_2

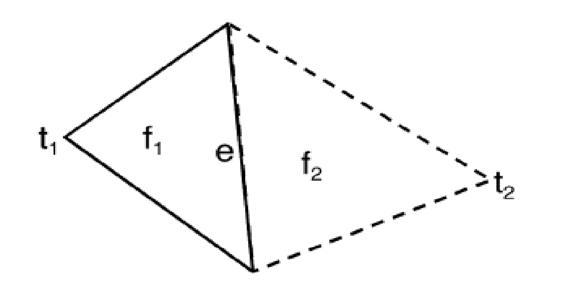
Algorithmus - 4. Inkrementelle Triangulation

Auswahl eines geeigneten f_2:

nicht alle potentiellen f_2 sind „geometisch schön“

verschiedene Kriterien zur Auswahl denkbar,

hier: kleinster Abstand von t_2 zum Mittelpunkt von e



Algorithmus – Komplexität (1/2)

F = Anzahl der *Flächen* (faces) einer Oberfläche
E = Anzahl der *Kanten* (edges)
n = Anzahl der *sample points*

Analyse

(a) Hauptschleife wird genau F mal durchlaufen

(b) Komplexität der Schleife in FILTER ist O(n)
Berechnung von L ist O(n + E + F)

(c) Komplexität von PROJECT ist O(|Γ|)

(d) Komplexität von INTERSECT ist O(|L|)

(e) 'innere Schleife' (for all t_2 in L): O(|Γ|*|L|)

```
M = {xi}i=1 bis n

RECONSTRUCT(M)
  INITIALIZE(M)
  do
    b = barycenter(e)
    FILTERING
    PROJECTION
    dmin = inf
    for t2 ∈ Γ
       f2_pot = e U {t2}
       if ||xt2 - b|| < dmin
          if !INTERSECTION
             dmin = ||xt2 - b||
             f2 = f2_pot
    K = K U {s: s ⊂ f2}
    draw(e, f1, t1)
  until there is no e
return K
```

Algorithmus – Komplexität (2/2)

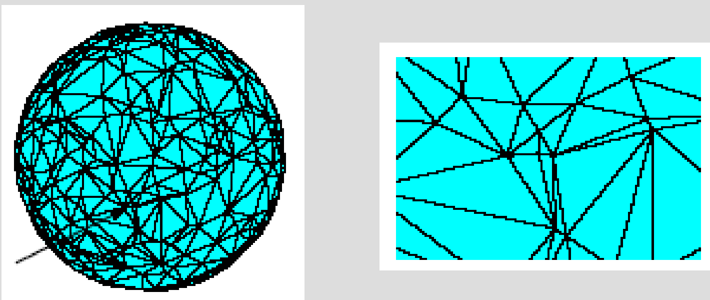
Analyse

Aus (a)-(e) ergibt sich mit einigen Umformungen und der Tatsache,dass K eine Mannigfaltigkeit darstellt, die Komplexität

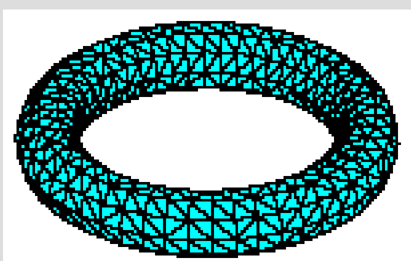
$$O(\ n^2 + n * |\Gamma| * |L|\ )$$

→ Exponent von *n* abhängig von der Dimension *D* des Raums, in dem die Fläche eingebettet ist

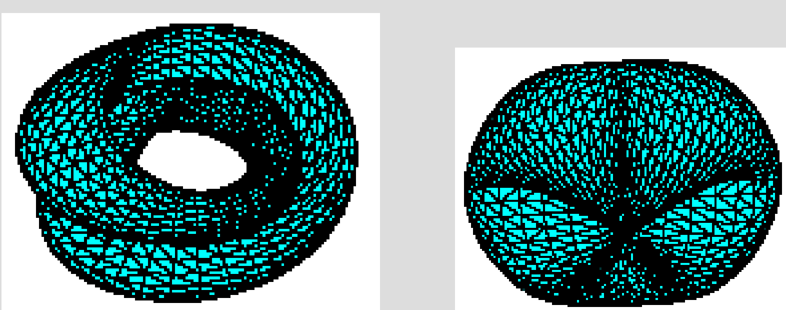
Algorithmus - Ergebnisse



(a) reconstruction of a sphere, highly non-uniformed sampling



(b) korrekte Ergebnisse auf Oberflächen mit nicht-sphärischer Topologie: Beispiel Torus



(c) Varianten der Klein'schen Flasche als Beispiele für nicht-orientierbare Flächen (keine Einbettung in R^3 möglich): Einbettung hier in R^5

Seminar Surface Reconstruction

Literatur

Daniel Freedman	An incremental algorithm for reconstruction of surfaces of arbitrary codimension. Computational Geometry: Theory and Applications, 36(2):106-116, 2007. Surface reconstruction, one triangle at a time. In Proceedings of the Sixteenth Canadian Conference of Computational Geometry (CCCG), pages 15-19, 2004. http://www.cs.rpi.edu/~freedd/
H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, W. Stuetzle	Surface Reconstruction from unorganized points. In Proc. SIGGRAPH '92, pages 71-78, 1992.
Klaus Jänich	Lineare Algebra, 10. Auflage, Springer
Bronstein u.a.	Taschenbuch der Mathematik, Kapitel 4 zu Lineare Algebra

Seminar Surface Reconstruction

Danke für die Aufmerksamkeit!