

# Entfernungen, Kräfte, Kollisionen

- Problemstellung
  - Abstands- und Kontaktinformation wozu?
- Objektmodellierung
- Überblick Algorithmen
  - Kollision: Ja oder nein?
  - Abstand: Wie weit (und in welche Richtung)
  - Kontakt: Eindringung wo und wie tief?
  - Kontakte bei nicht-konvexen Objekten

# Kontaktinformation wozu?

Computergrafik, Virtuelle Realität:

Sehr häufig Simulation realer, physikalischer Szenen und Vorgänge

Kontakte, Kräfte:

Nachbildung des physikalischen Prinzips „Starre Körper können sich nicht durchdringen“ in der Computergrafik  
(Physik-Engines: Havok, ODE, PhysX, Bullet, ...)

Abstände: Information des Benutzers, Grundlage für Strategien, ...

CAD, Robotik, Spiele,...

# Kontaktinformation wozu?

Beispiel Robotikanwendungen auf der ISS

Roboter müssen sich kollisionsfrei bewegen  $\Rightarrow$

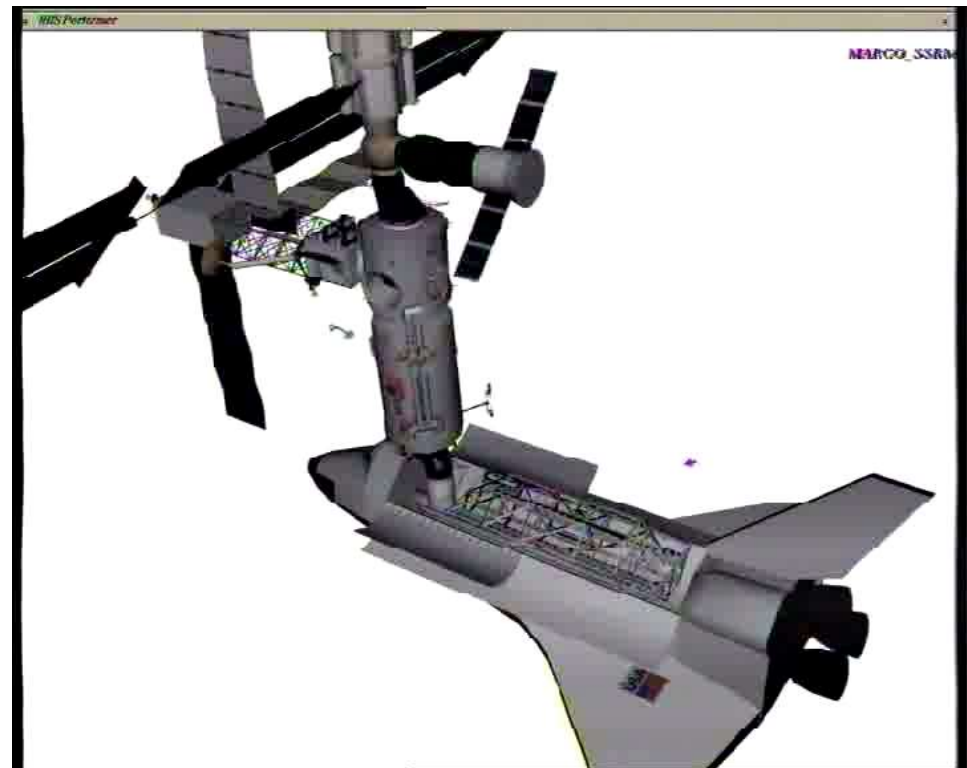
Planung kollisionsfreier Bahnen in VR  $\Rightarrow$

**Kollision, ja oder nein, Abstandsinformation**

Roboter müssen mit Kontakten zurechtkommen  $\Rightarrow$

Simulation von Kraftreglern in VR  $\Rightarrow$

**Kontaktinformation**



# Kontaktinformation wozu?

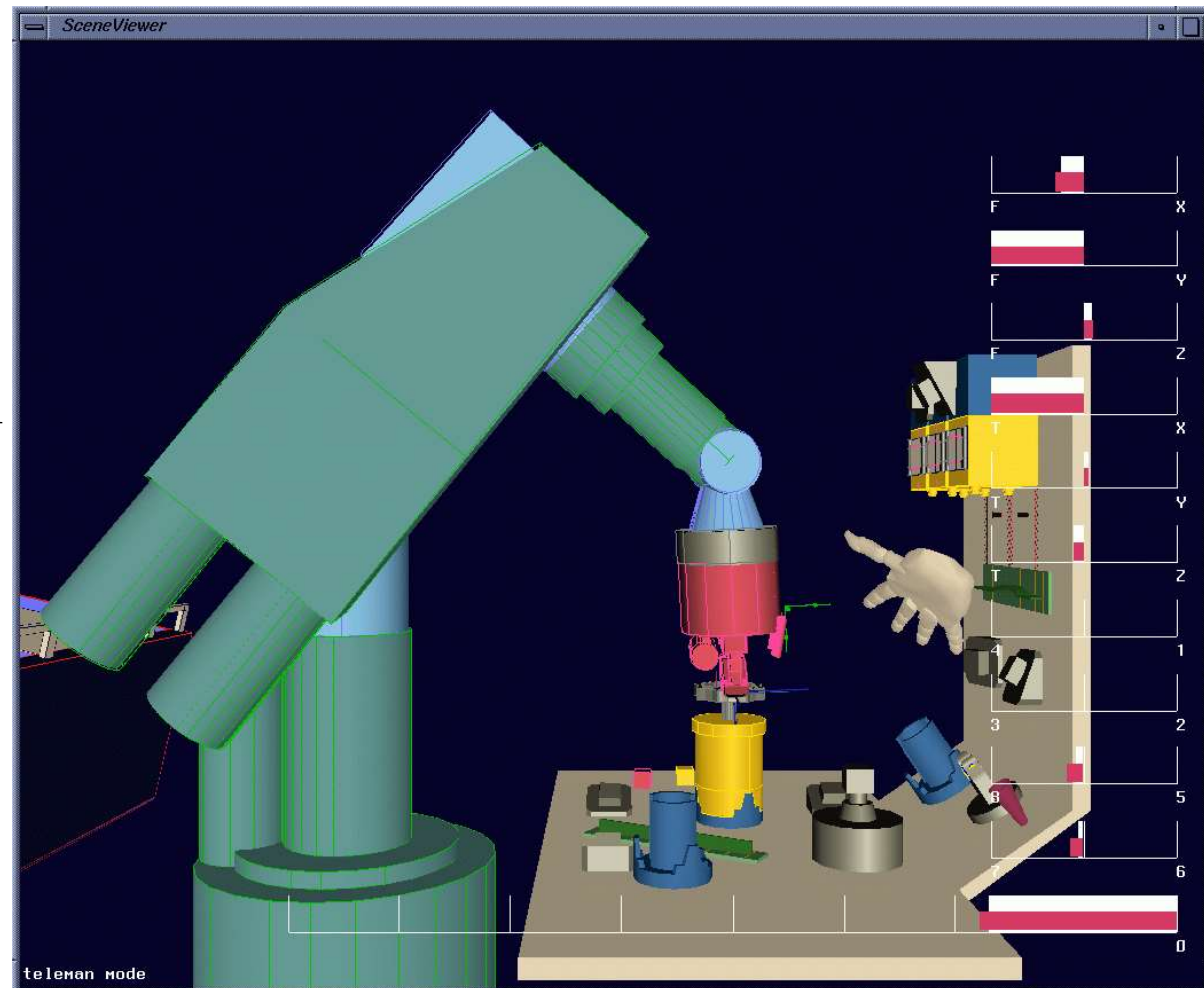
## Beispiel Telerobotik-Workstation

Simulation von

- Kraftsensoren
- Abstandssensoren

Bahnplanung

Bedienerinteraktion



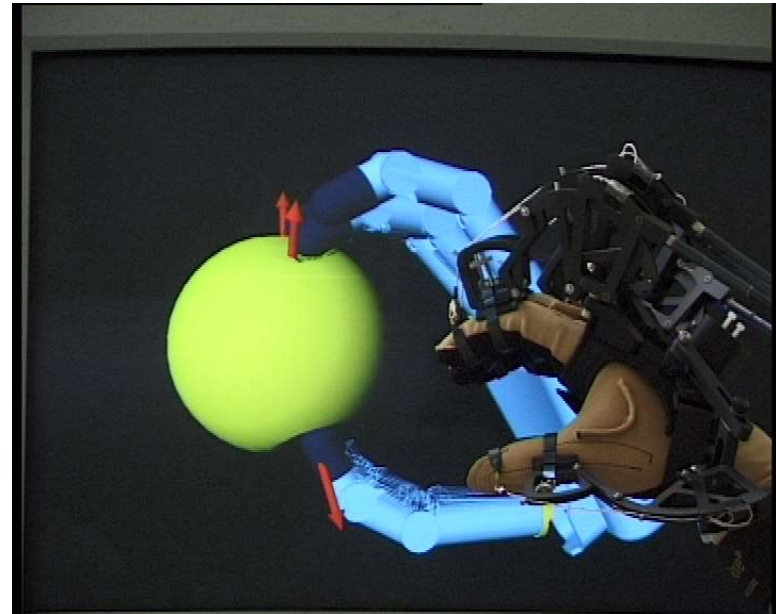
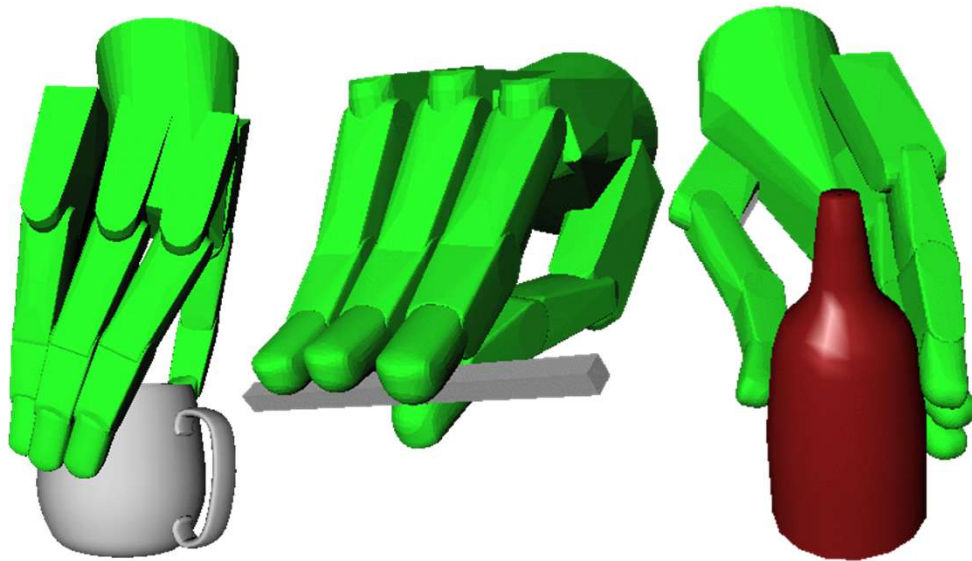
# Kontaktinformation wozu?

Beispiel Greifplanung, Interaktion in VR-Systemen

Planungssystem benötigt **Kontaktinformation**,

Rückkopplung von **Kräften** an den Benutzer,

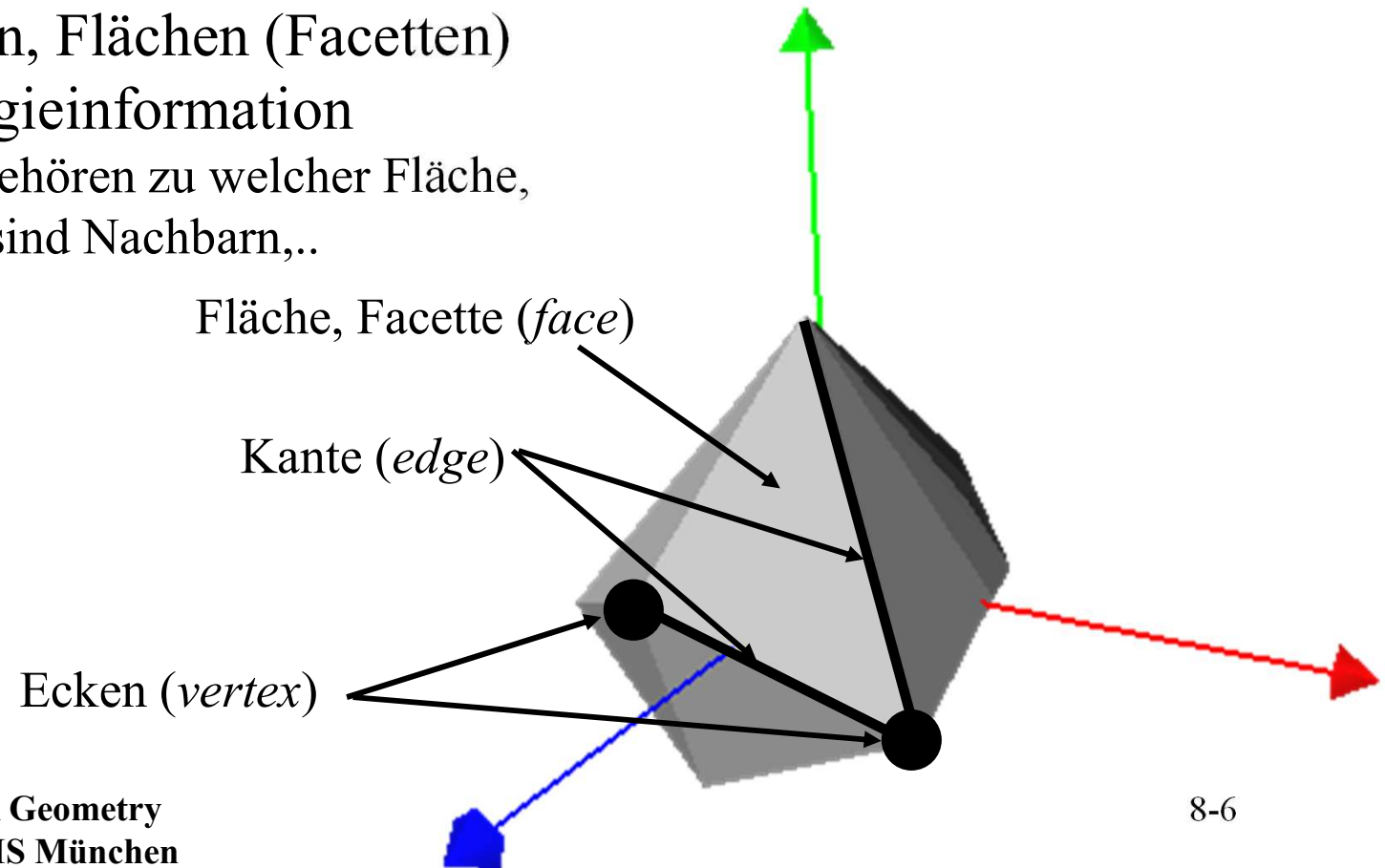
Objekte werden durch **Kräfte** bewegt



# Objektmodelle

Hier verwendete Objektmodellierung  
Oberflächendarstellung  
Polyedermodelle

- Ecken, Kanten, Flächen (Facetten)
- (ggf.) Topologieinformation  
welche Kanten gehören zu welcher Fläche,  
welche Flächen sind Nachbarn,...



# Klassifikation, Algorithmen und Objektmodelle

	Konvexe Körper	Allgemeine Körper
Distanz	Voronoi (GJK)	Hierarchie konvexer Körper, dann Voronoi (bzw.GJK)
Kollision, ja oder nein	s.o., oder Hüllkörper	Hüllkörper
Kontaktinfo	Minimale translatorische Distanz, basierend auf GJK	Kombination GJK, Hüllkörper, MDT



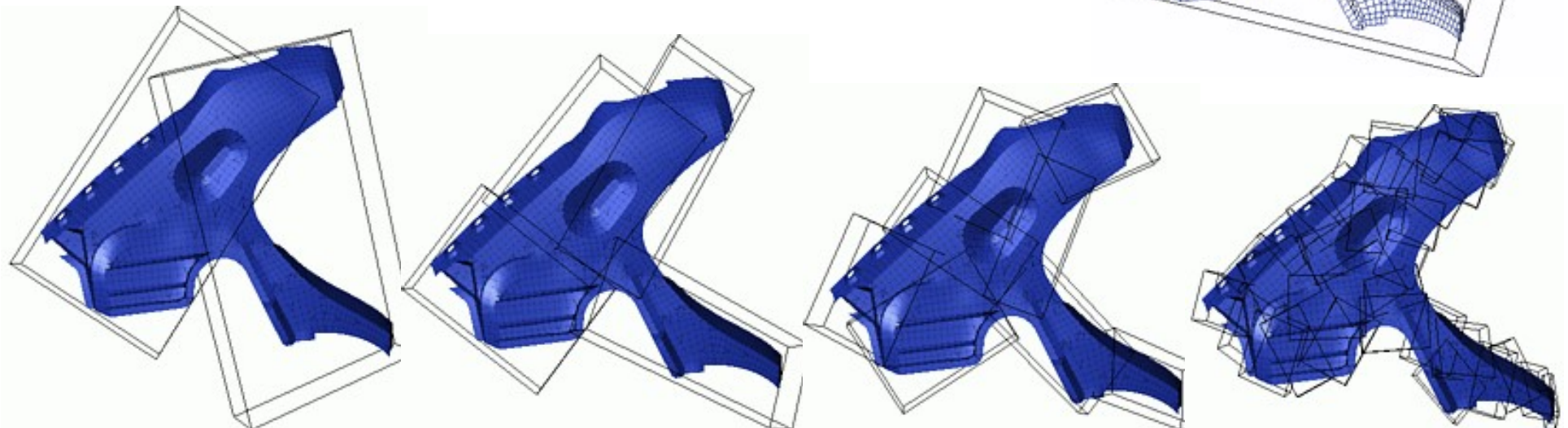
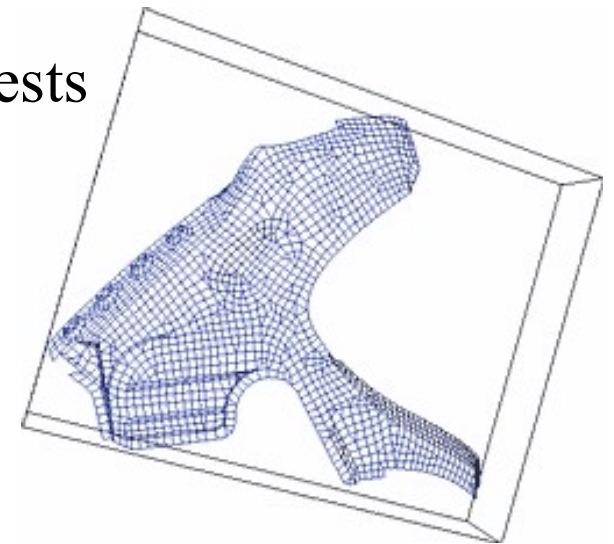
# Kollision, ja oder nein?

## „Primitiver“ Ansatz

Zwei Objekte kollidieren, wenn zwei Facetten sich schneiden  $\Rightarrow$   
jede Facette gegen jede auf Kollision testen

- quadratische Komplexität, hohe Anzahl von Tests  
(z.B.  $1000 \times 1000 = 1000000$  !!!)
- Test Facette/Facette teuer

Lösung: Hierarchien von Hüllkörpern





# Kollision, ja oder nein?

Welche Hüllkörper ???

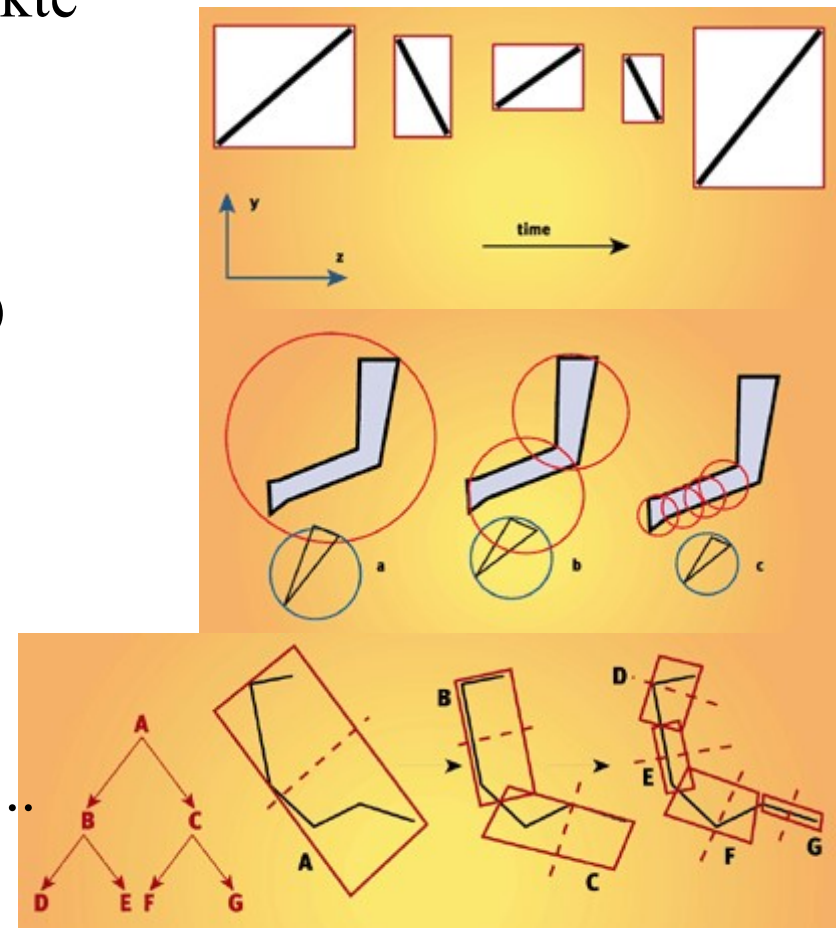
## Generelle Kriterien

- Geschwindigkeit des Kollisionstests zwischen ihnen
- Anpassbarkeit des Volumens an Objekte

## Typen von Hüllkörpern

- Achsparallele Quader (AABB)
  - Geschwindigkeit +
  - Anpassbarkeit -- (nicht rotationsinvariant)
- Kugeln
  - Geschwindigkeit ++
  - Anpassbarkeit -
- Orientierte Quader (OBB)
  - Geschwindigkeit -
  - Anpassbarkeit +
- Ellipsoide, „Swept Sphere Volumes“...

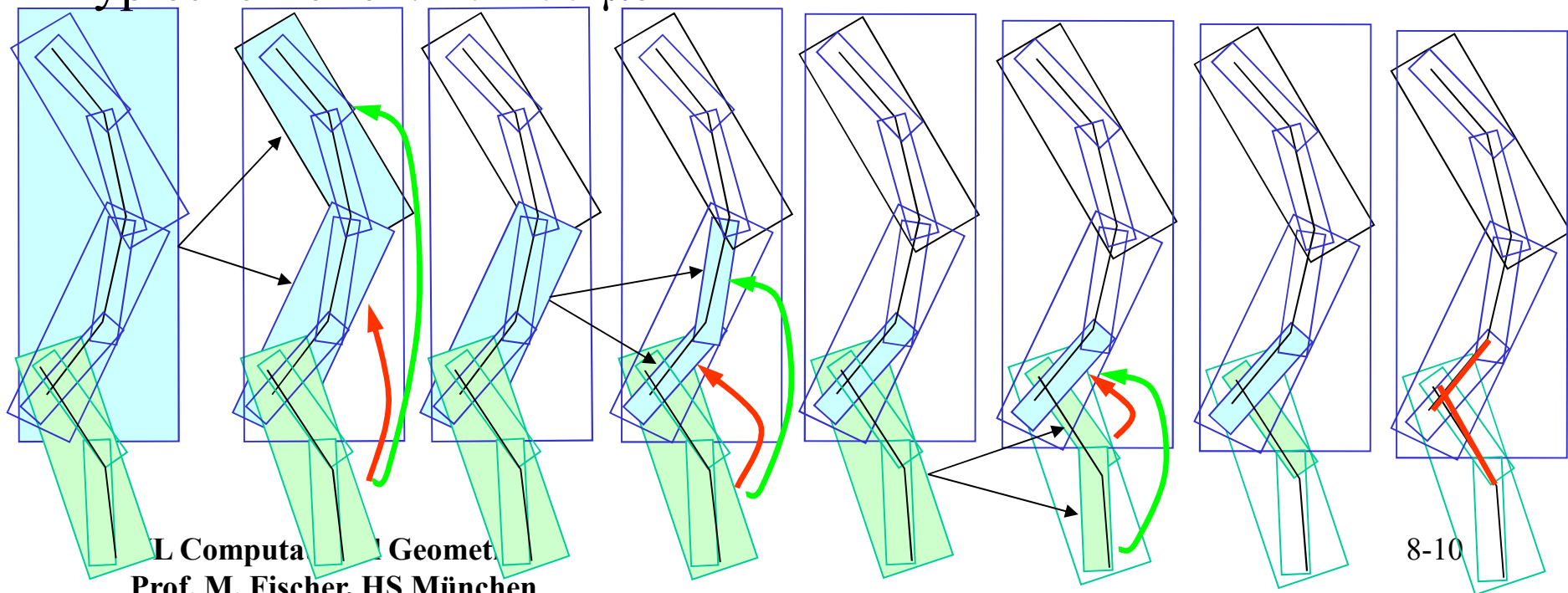
VL Computational Geometry  
Prof. M. Fischer, HS München



# Hüllkörperhierarchien

## Kollisionstest mit Hüllkörperhierarchien

- Beginne mit umfassendsten HK beider Objekte
- Wenn HK nicht kollidieren, kollidieren Objekte nicht
- Wenn HK kollidieren, teste einen HK mit den Kindern des anderen
- Wenn beide HK Facette als Kinder haben, teste Facetten
- typische Zeiten: 10-100  $\mu$ s

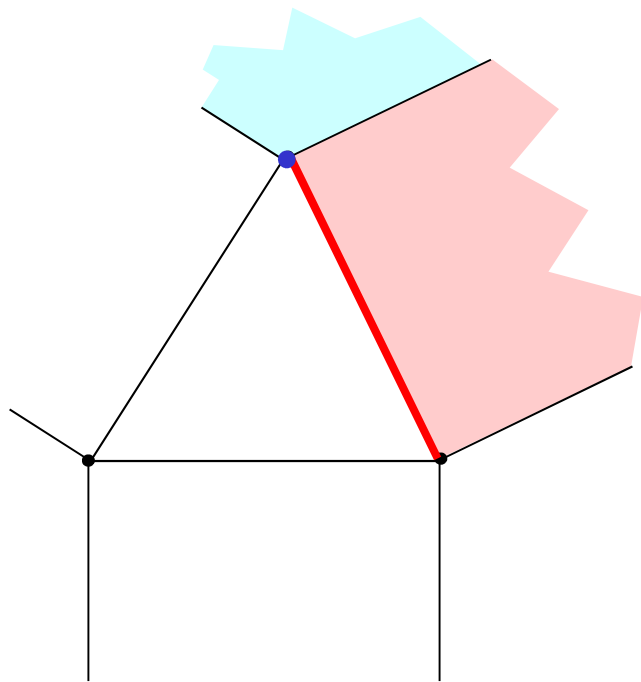


# Klassifikation, Algorithmen und Objektmodelle

	Konvexe Körper	Allgemeine Körper
Distanz	Voronoi (GJK)	Hierarchie konvexer Körper, dann Voronoi (bzw.GJK)
Kollision, ja oder nein	s.o., oder Hüllkörper	Hüllkörper
Kontaktinfo	Minimale translatorische Distanz, basierend auf GJK	Kombination GJK, Hüllkörper, MDT

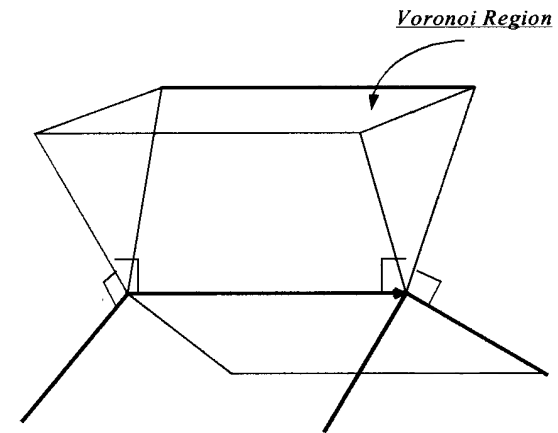
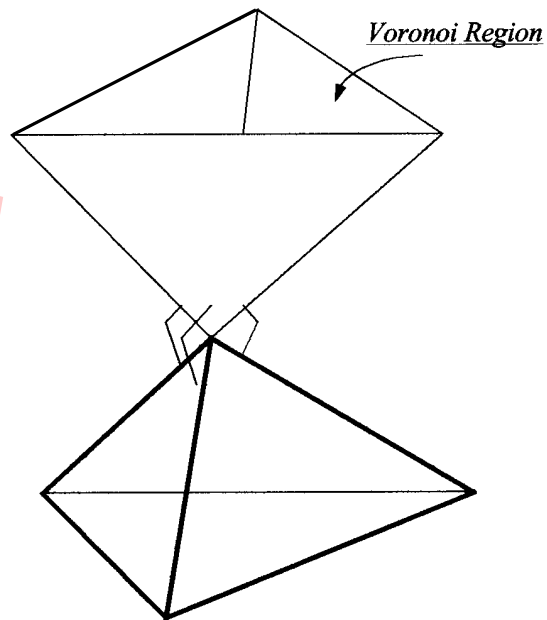
# Distanzberechnung mit Voronoi-Diagrammen

## Voronoi-Diagramme



**2D**

Primitive: Ecken, Kanten, Flächen



**3D**

# Distanzberechnung mit Voronoi-Diagrammen

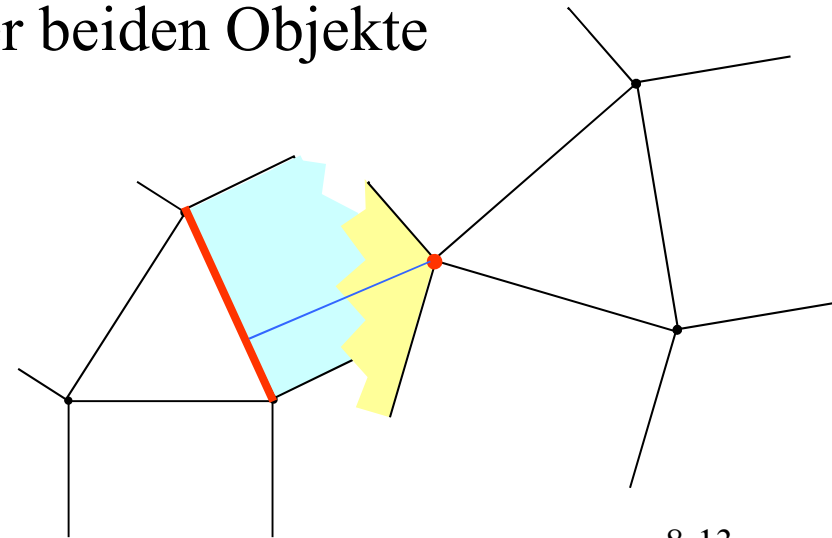
## Algorithmus von Lin, Canny (1988)

Lemma: Gegeben zwei konvexe Objekte.

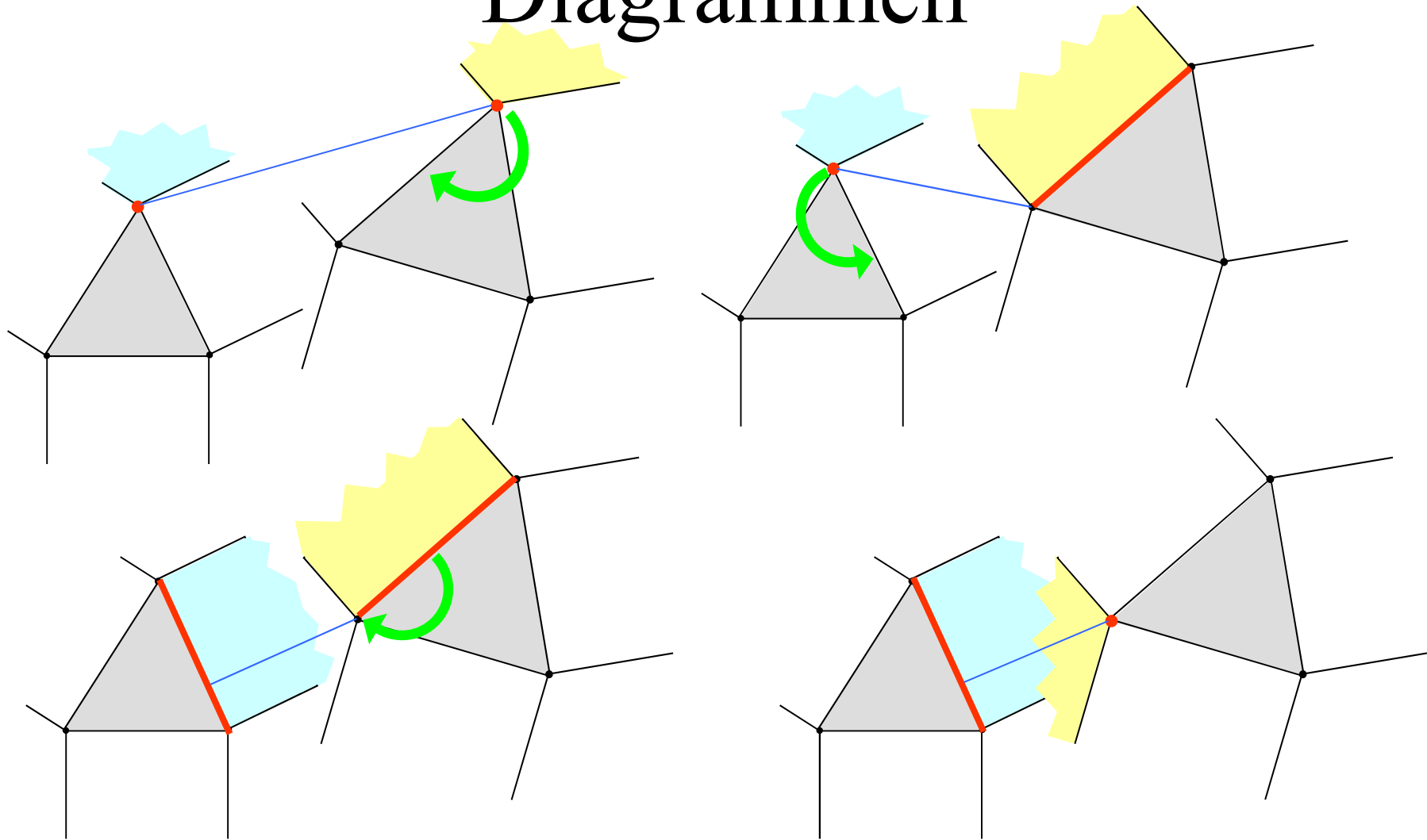
Wenn die nächstgelegenen Punkte zweier Primitive (Punkt, Kante, Fläche) jeweils in der Voronoi-Region des anderen Primitivs liegen, so sind diese Punkte die nächstgelegenen Punkte der beiden Objekte

Algorithmus:

Starte mit zwei Punkten  
„Wandere“ auf dem Objekt entlang,  
bis obige Situation eintritt  
Liefere die beiden Punkte (bzw.  
deren Abstand) als Ergebnis



# Distanzberechnung mit Voronoi-Diagrammen



# Distanzberechnung mit Voronoi-Diagrammen

## Anmerkungen

Sehr effizientes Verfahren !!!

(fast)  $O(1)$ , typische Zeiten  $< 10 \mu s$ , falls

- inkrementelle Bewegung
- Startwerte: nächstgelegenen Punkte der letzten Berechnung

## Konkurrenzverfahren

GJK (Gilbert, Johnson, Keerthi, 1991)

verwendet (implizit) Minkowski-Summe, Quadratic Programming



# Klassifikation, Algorithmen und Objektmodelle

	Konvexe Körper	Allgemeine Körper
Distanz	Voronoi (GJK)	Hierarchie konvexer Körper, dann Voronoi (bzw.GJK)
Kollision, ja oder nein	s.o., oder Hüllkörper	Hüllkörper
Kontaktinfo	Minimale translatorische Distanz, basierend auf GJK	Kombination GJK, Hüllkörper, MDT

# Kontaktinformation für konvexe Objekte

## Berechnung von Kräften

- Dynamiksimulation, Reaktion auf Kräfte
- Federgesetz:  $F = k \cdot \Delta x$  ( $\Delta x$  : Auslenkung, hier Eindringungstiefe)

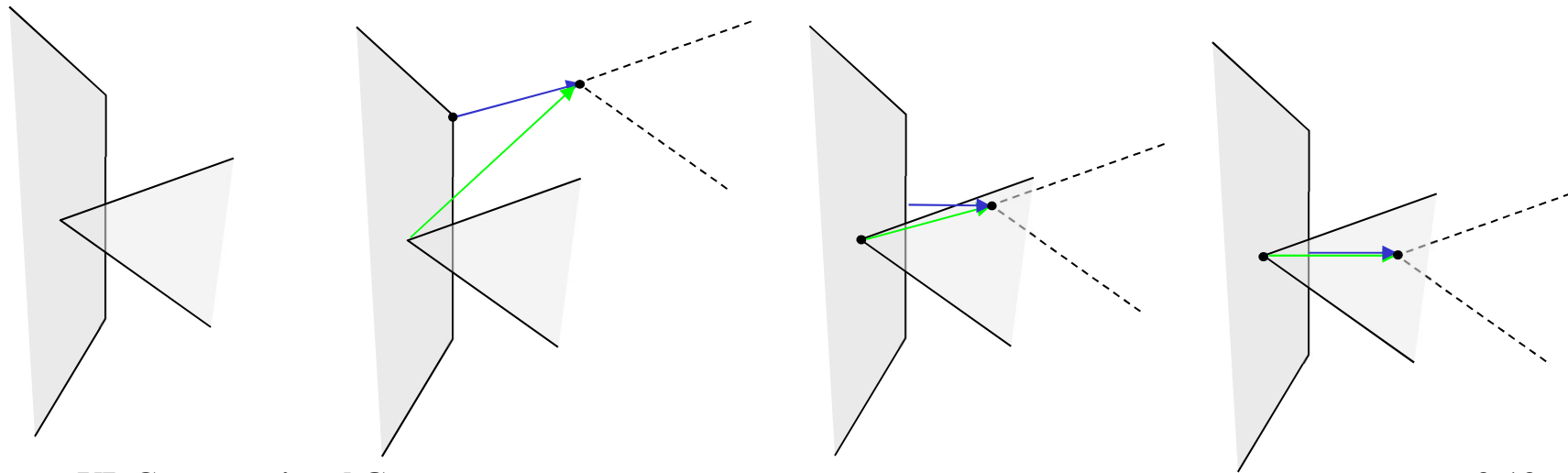
## Minimale translatorische Distanz (MTD)

Minimale Verschiebung entlang der besten von allen Richtungen, um kollidierende Objekte zu trennen

# Kontaktinformation für konvexe Objekte

## Algorithmus (iterativ)

- Verschieben in **gegebene Richtung**  $\vec{v}$  (initial heuristisch)
- Bestimmen des **Vektors**  $\vec{d}$  **zwischen den beiden Nahpunkten**
- Fertig, wenn  $\vec{v}$  und  $\vec{d}$  parallel (MTD: Differenz der beiden Vektoren)
- Neuer Verschiebungsvektor  $\vec{v}'$  :  $\vec{v}' \parallel \vec{d}$ ,  $|\vec{v}'| = (|\vec{v}| - |\vec{d}|) \cdot (1 + \varepsilon)$



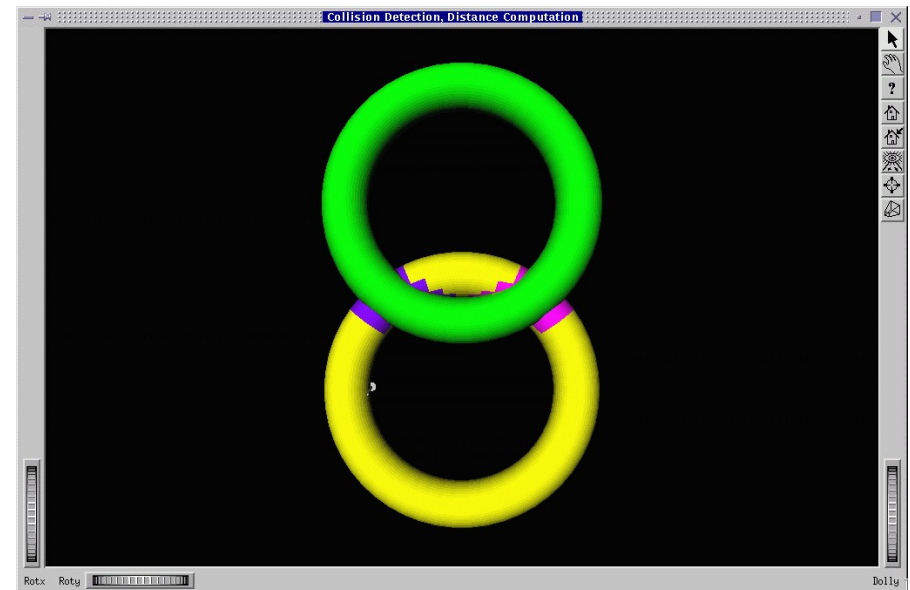
# Klassifikation, Algorithmen und Objektmodelle

	Konvexe Körper	Allgemeine Körper
Distanz	Voronoi (GJK)	Hierarchie konvexer Körper, dann Voronoi (bzw.GJK)
Kollision, ja oder nein	s.o., oder Hüllkörper	Hüllkörper
Kontaktinfo	Minimale translatorische Distanz, basierend auf GJK	Kombination GJK, Hüllkörper, MDT

# Kontaktinformation für nichtkonvexe Objekte

## Algorithmus

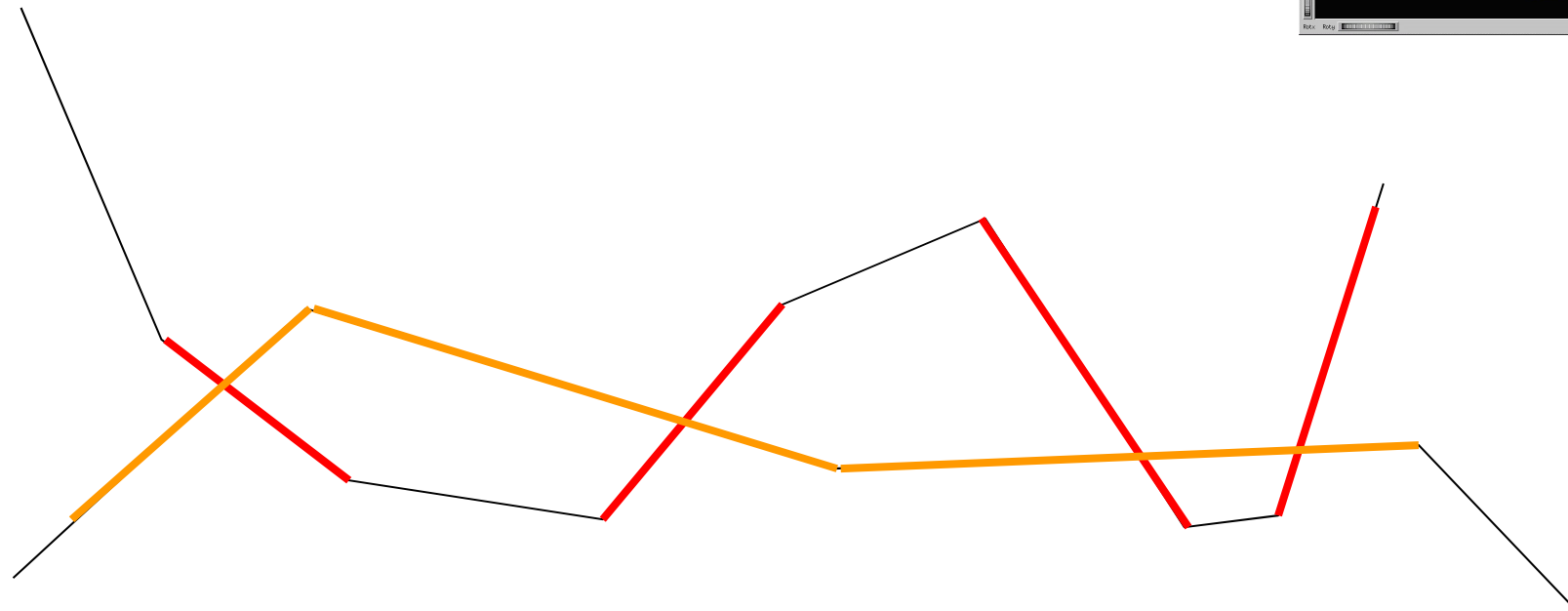
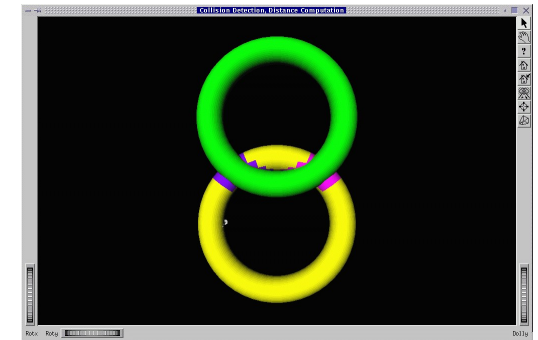
- Bestimmen aller kollidierender Facettenpaare
- Trennen in einzelne Kontaktregionen
- Vervollständigen der „Kontaktringe“
- Bestimmen der MTD pro Kontaktregion



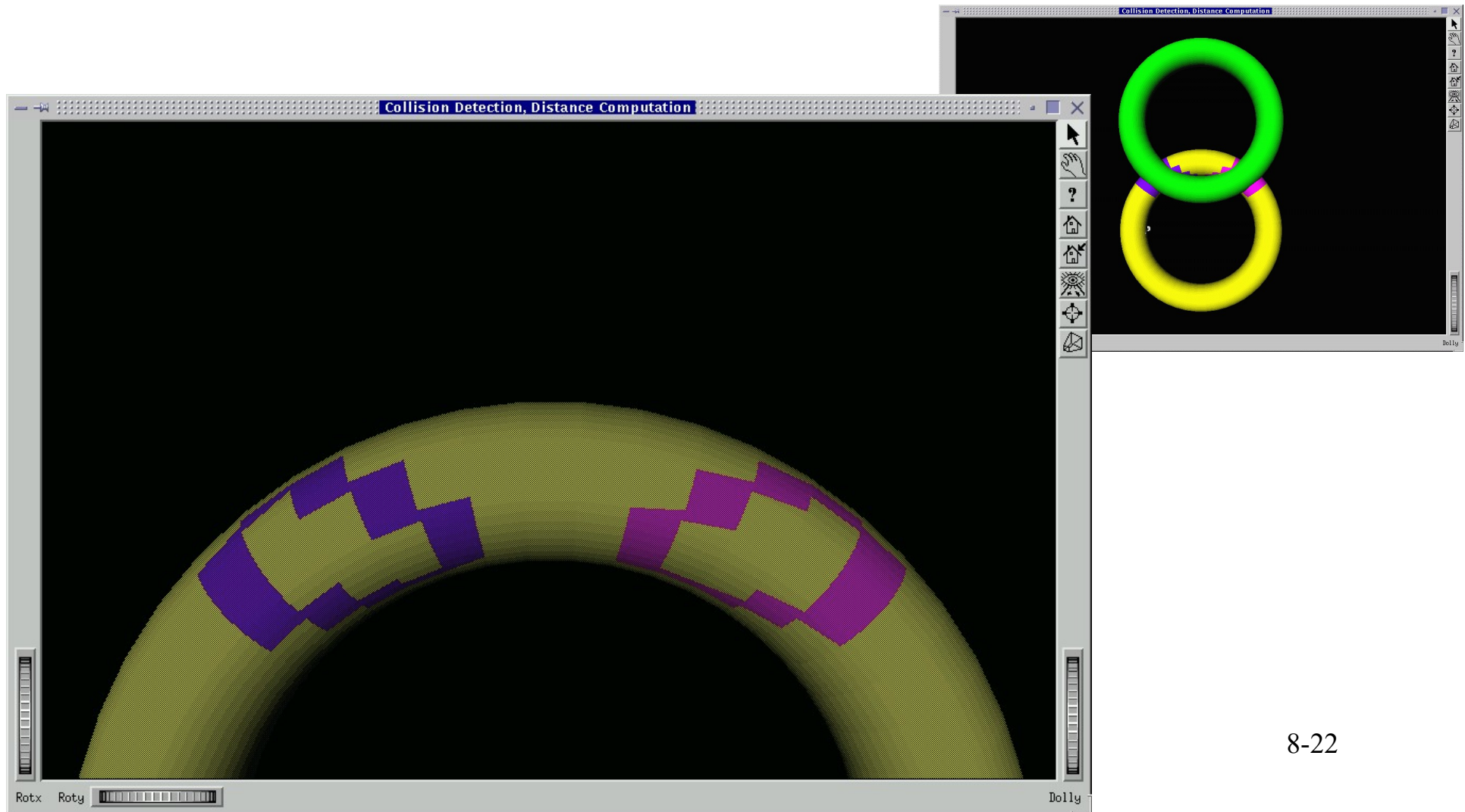
# Kontaktinformation für nichtkonvexe Objekte

Bestimmen kollidierender Facettenpaare

- Standardverfahren mit Hüllkörperhierarchien



# Kontaktinformation für nichtkonvexe Objekte





# Kontaktinformation für nichtkonvexe Objekte

Trennen in einzelne Kontaktregionen

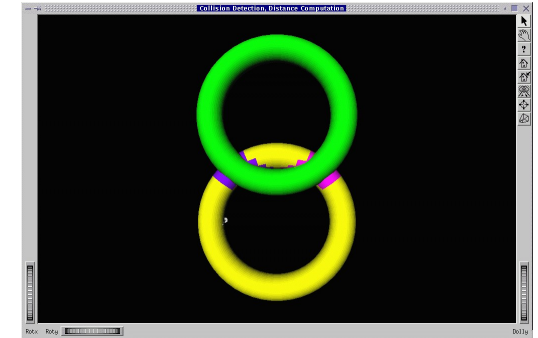
- Mittels Topologieinformation (*Winged Edge*)

Für alle kollidierenden Facettenpaare:

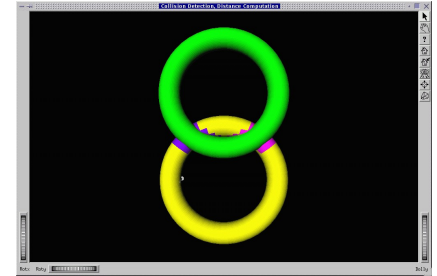
Mit keinem Kontaktring benachbart → in neuen Kontaktring

Mit einem Kontaktring benachbart → in Kontaktring hinzufügen

Mit mehreren Kontaktringen benachbart → diese verschmelzen und Facettenpaar zu verschmolzenem Kontaktring hinzufügen



# Kontaktinformation für nichtkonvexe Objekte



Vervollständigen der „Kontaktringe“

**Für alle** Facetten eines Kontaktrings:

Suche Kante, die ganz im Inneren des Partnerobjekts

liegt **und** die eine Nachbarfacette hat, die nicht im Kontaktring liegt

Diese Nachbarfacette auf den Stack

**Solange** Stack nicht leer:

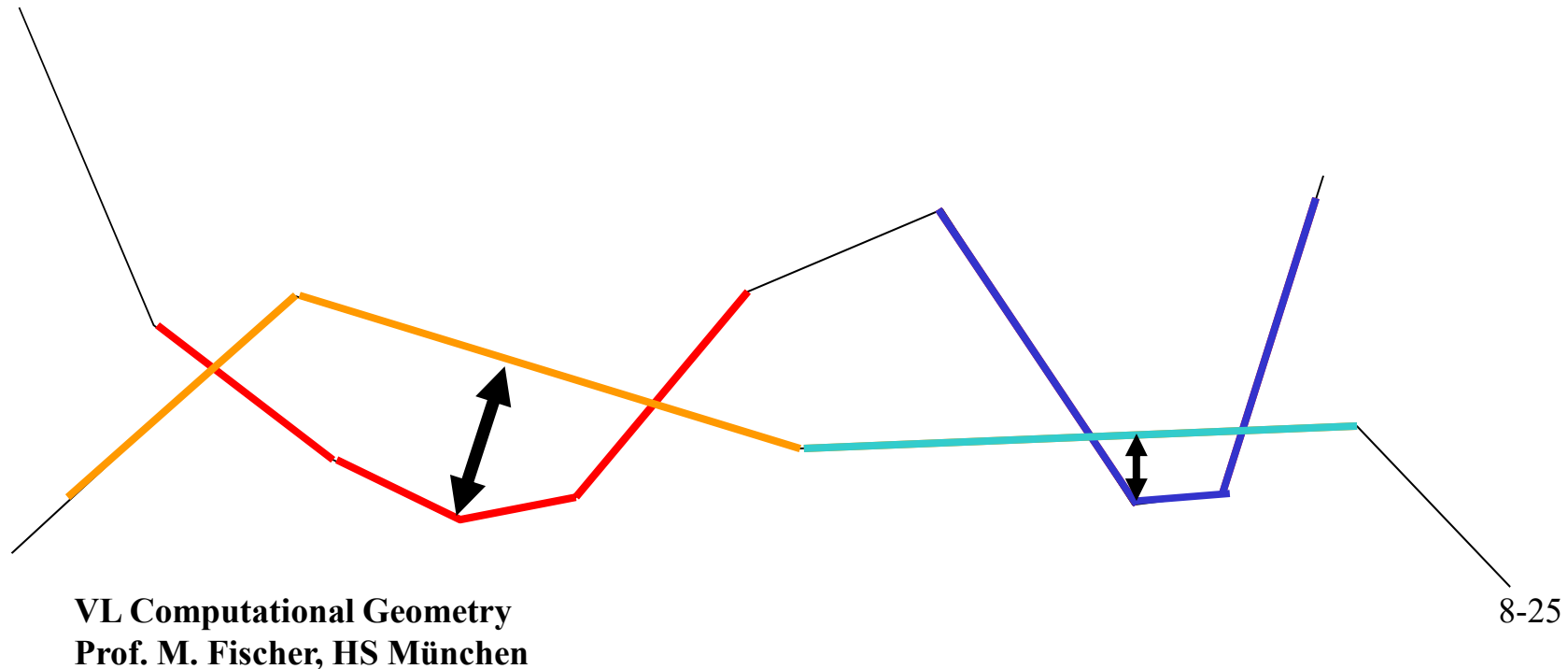
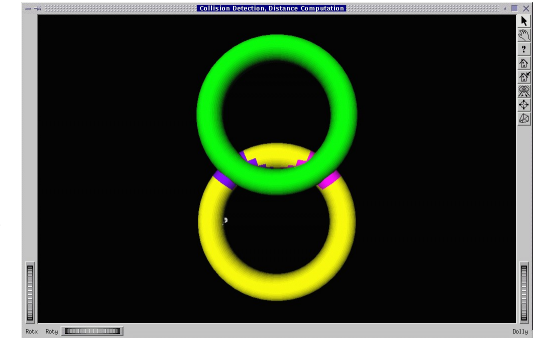
Oberste Facette vom Stack nehmen **und** zum Kontaktring hinzufügen

**Für alle** Kanten der obersten Facette:

**Wenn** Nachbarfacette nicht im Kontaktring → auf den Stack legen

# Kontaktinformation für nichtkonvexe Objekte

Bestimmen der MTD pro Kontaktregion  
Mittels zuvor beschriebenem, iterativen Verfahren

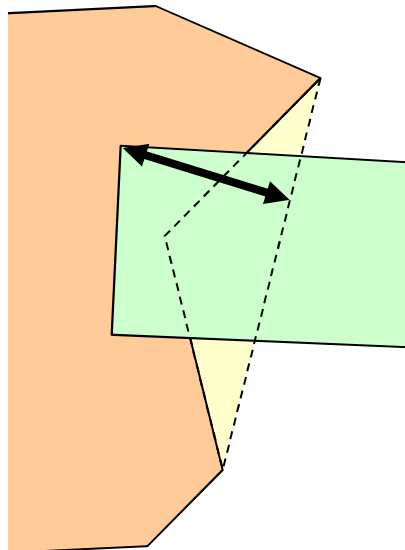


# Kontaktinformation für nichtkonvexe Objekte

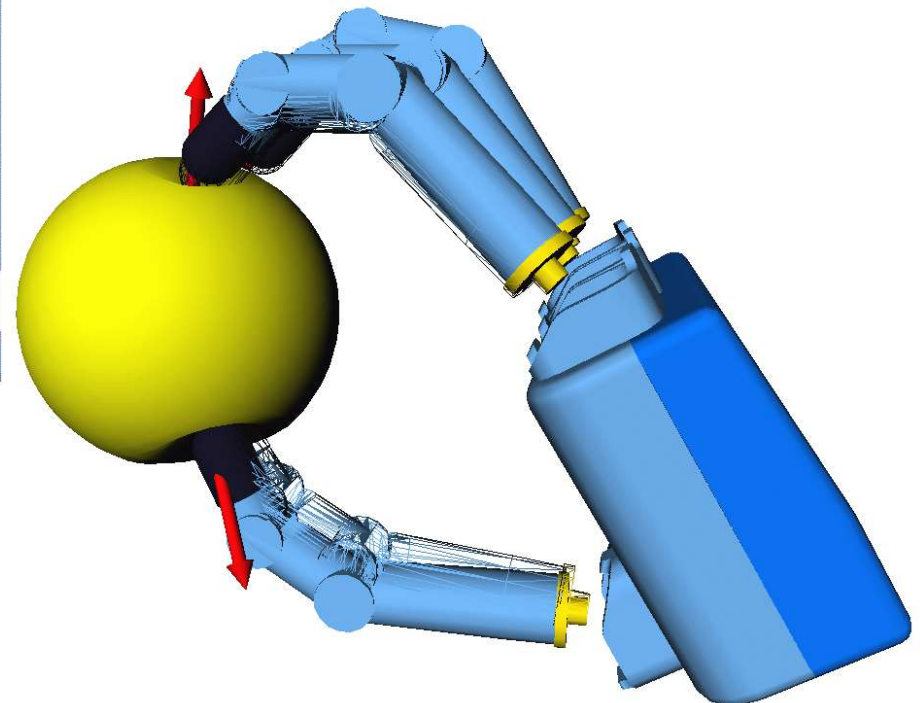
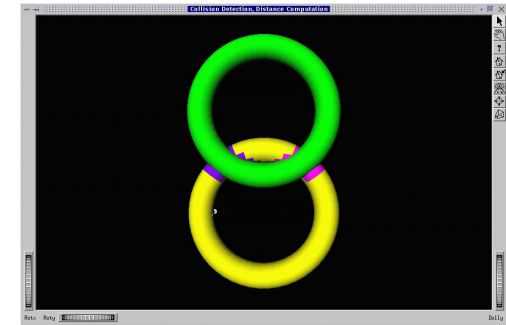
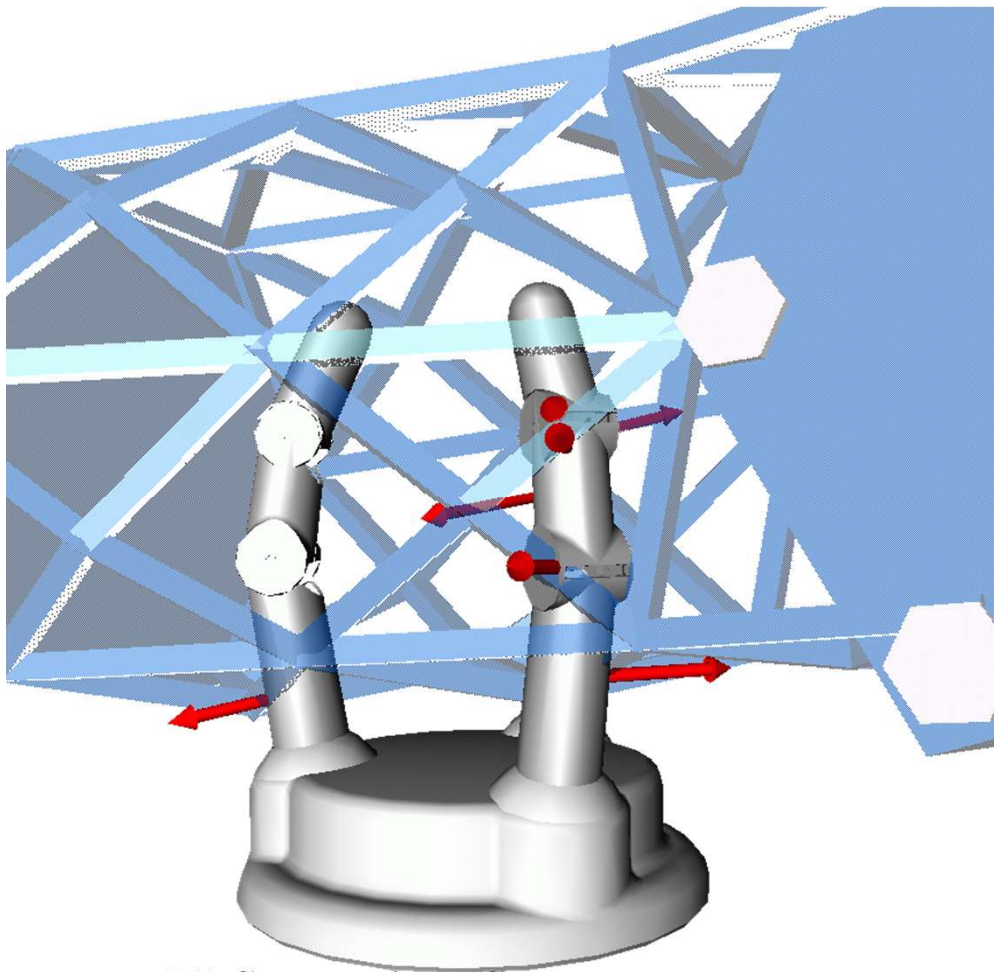
## Probleme

- „Trennen“ und „Vervollständigen“ haben quadratische Komplexität
- Konkave Kontaktregionen werden durch konvexe Hülle approximiert

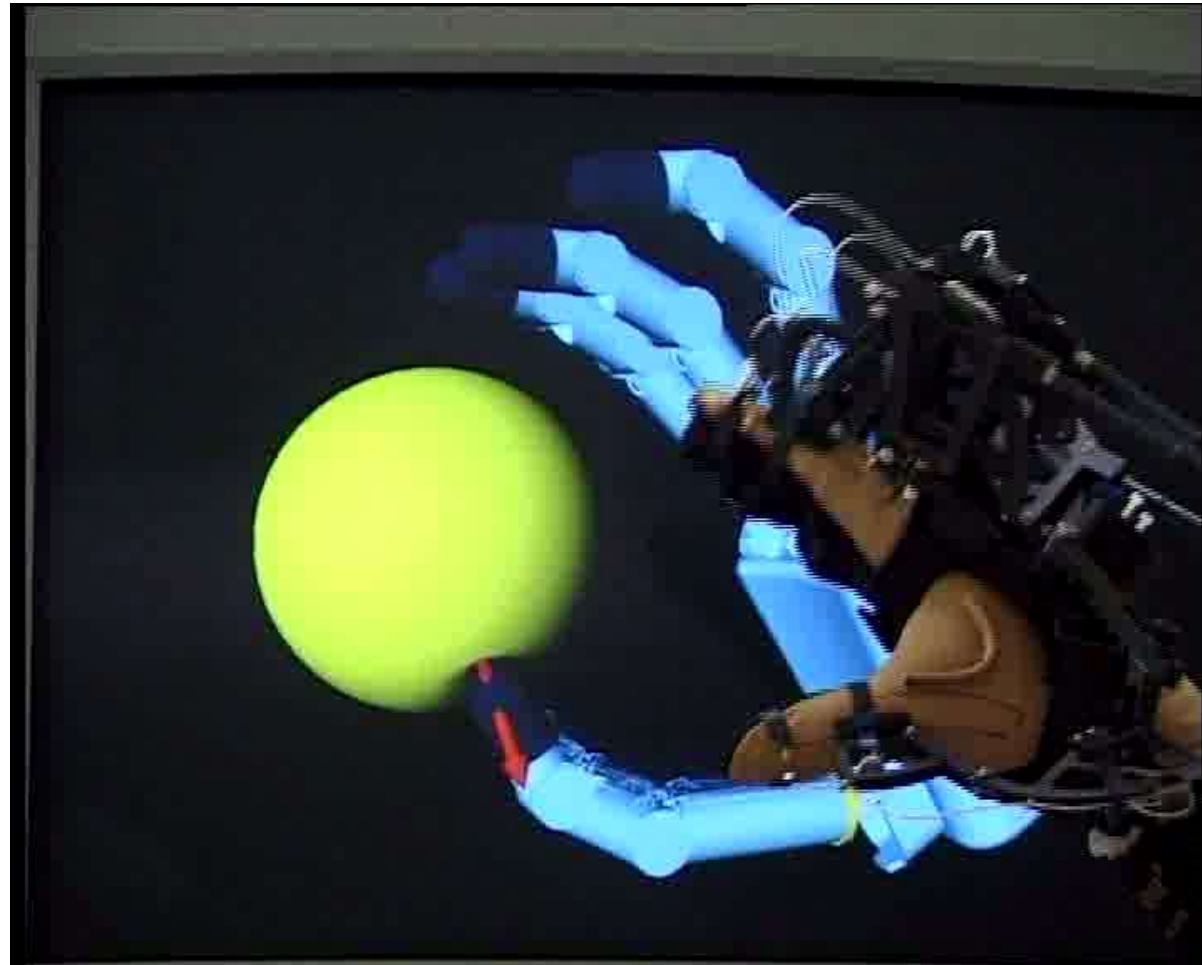
Abhilfe: Schrittweite so wählen, daß Kontaktregionen klein bleiben



# Kontaktinformation für nichtkonvexe Objekte

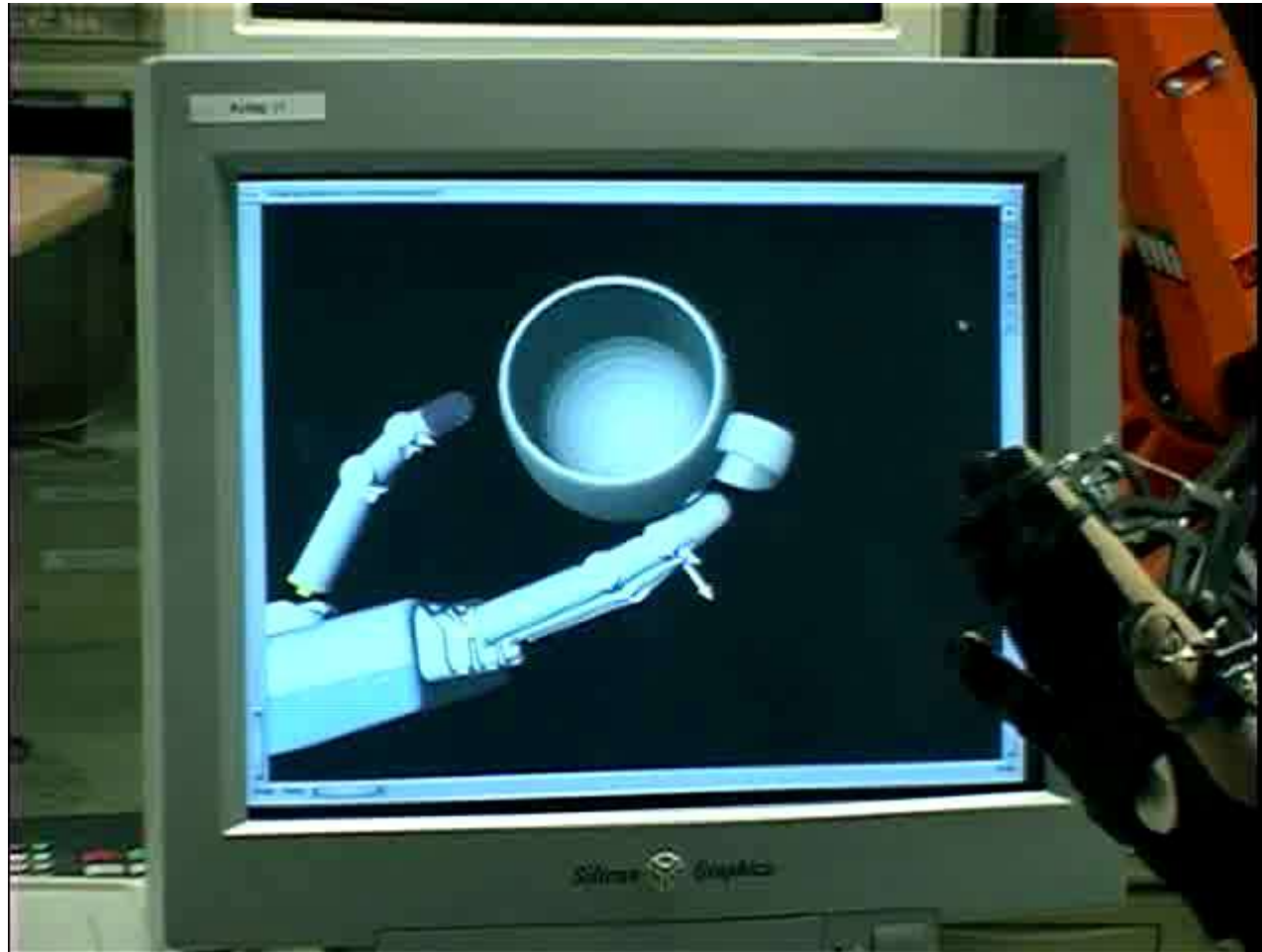


# Kontaktinformation für nichtkonvexe Objekte





# Kontaktinformation für nichtkonvexe Objekte





# Klassifikation, Algorithmen und Objektmodelle

Fertig...