

Kapitel 3

Ähnlichkeitsmodelle für räumliche Objekte

3. Ähnlichkeitsmodelle für räumliche Objekte

■ Übersicht

3.1 Ähnlichkeitsmodelle für räumliche Objekte – Einführung

3.4 Ähnlichkeitsmodelle

3.4.1 Formhistogramme

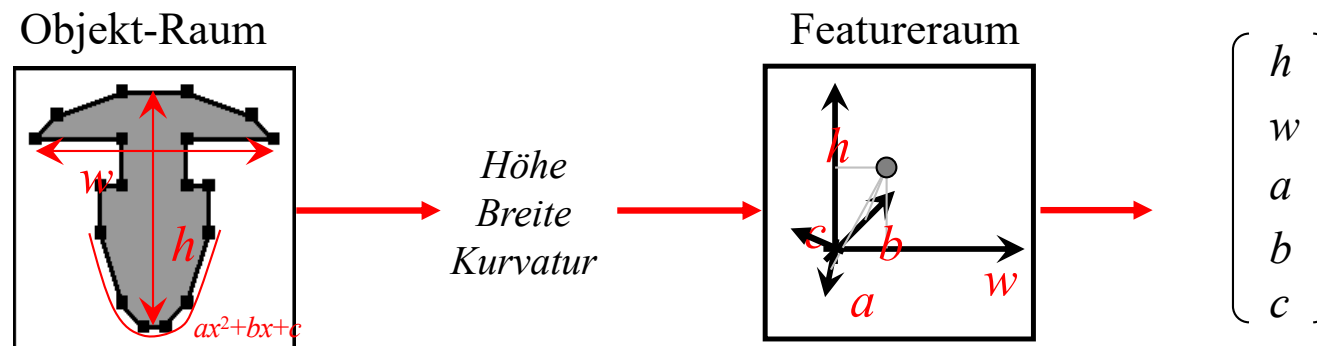
3.4.2 Erweiterte Formhistogramme

3.4.3 Überdeckungsmodell für 2D Objekte

3.4.4 Überdeckungsmodell für 3D Objekte

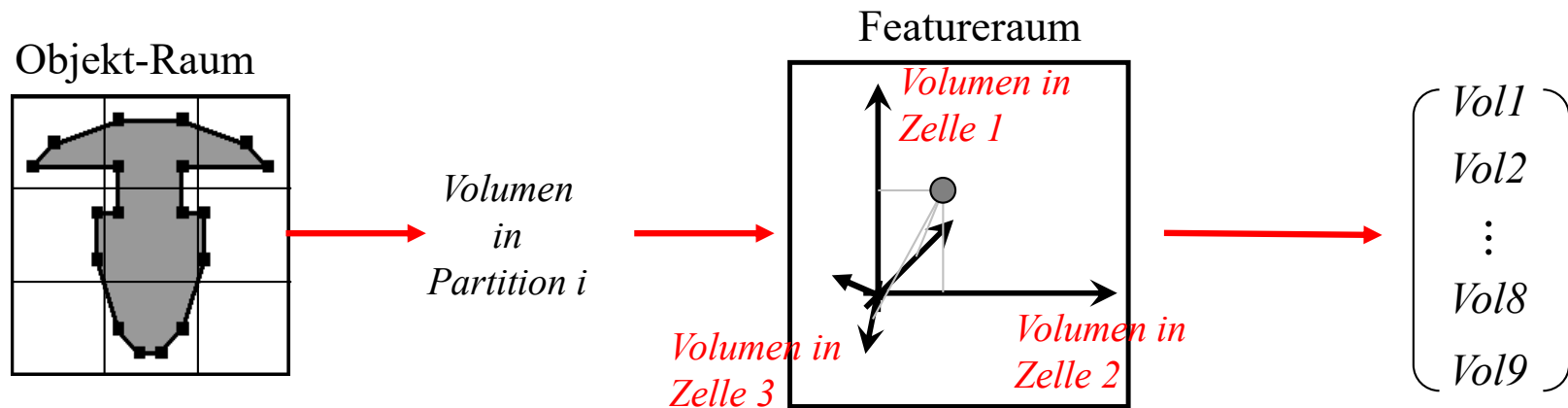
3.1 Einführung

- Feature Transformation für räumliche Objekte
 - Ziel: „gute“ Beschreibung der realen Objekte als Featurevektoren (metrisch oder besser: Euklidisch)
 - Ähnlichkeit im Objektraum \approx Ähnlichkeit im Featureraum
 - D.h. Merkmale sollten „sinnvoll“ / „aussagekräftig“ sein
 - Möglichkeit 1:
 - Extrahiere Merkmale für das gesamte Objekt



3.1 Einführung

- Möglichkeit 2:
 - Partitioniere Objektraum
 - **Objekt-spezifische Partitionierung:** das Objekt wird zerlegt, unabhängig davon, wie es im Datenraum liegt
 - **Datenraum-spezifische Partitionierung:** der Datenraum wird zerlegt, unabhängig davon, wie das Objekt darin liegt
 - Extrahiere Merkmale aus einzelnen Partitionen
 - Z.B. Volumen des Objekts in jeder Partition



3.4.1 Formhistogramme für 2D und 3D Objekte

4.1.1 Formhistogramme für 2D und 3D Objekte

[Ankerst,

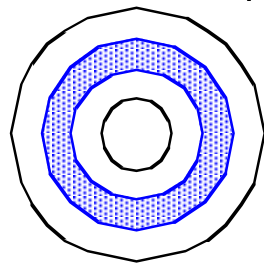
Kastenmüller, Kriegel, Seidl. Proc. Int. Symp. Large Spatial Databases (SSD), 1999]

□ Anwendung:

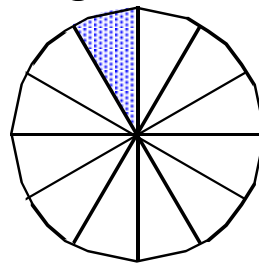
- Objekte sind Mengen von Oberflächenpunkten
- CAD-Bausteine, Moleküle, etc.

□ Grundidee: Formhistogramme (primär ausgelegt für 2D Raum)

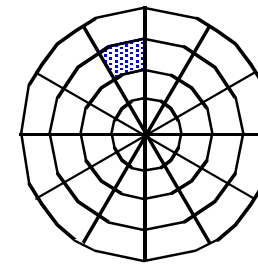
- Partitioniere den Objektraum (2D)
- Bestimme Anzahl von Oberflächenpunkten des Objekts pro Zelle (normiertes Histogramm; unabhängig von Punktdichte)
- Verschiedene Raumpartitionierungen



Schalenmodell



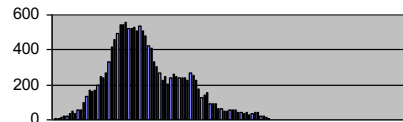
Sektorenmodell



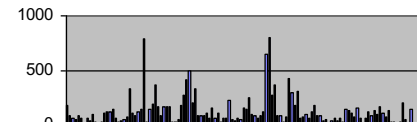
Kombiniertes Modell

3.4.1 Formhistogramme für 2D und 3D Objekte

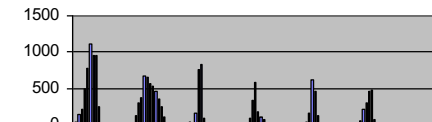
■ Beispiel: Protein-Oberfläche



Schalenmodell
(120 Schalen)



Sektorenmodell
(122 Sektoren)



Kombiniertes Modell
(20 Schalen, 6 Sektoren)

■ Histogramm-Definition modell-spezifisch

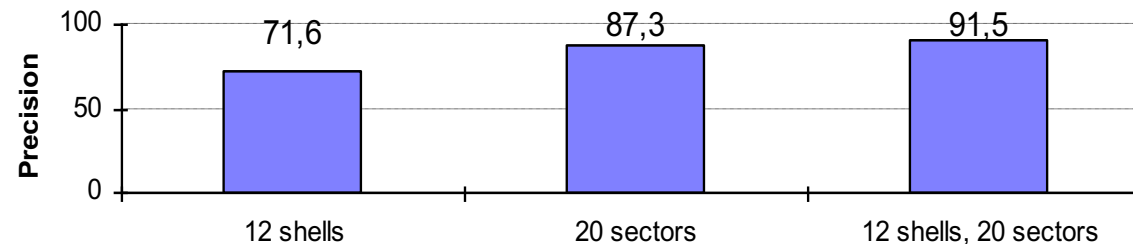
- **Schalenmodell**: Definiere die Bins über den Abstand zum Mittelpunkt, d.h. Anzahl der Punkte auf der jeweiligen Schale.
- **Sektorenmodell**: Anzahl der Punkte im jeweiligen Sektor.
- **Kombiniertes Modell**: Synthese aus Schalen- und Sektorenmodell

■ Invarianten

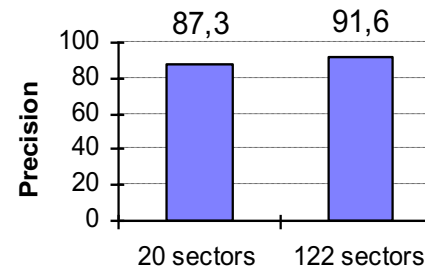
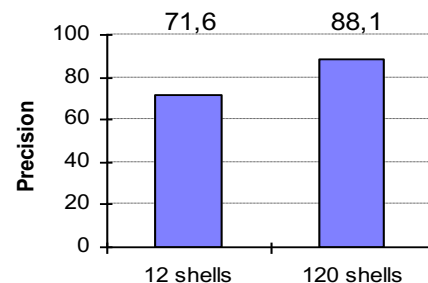
- **Rotationsinvarianz** beim Schalenmodell
- **Skalierungsinvarianz** beim Sektorenmodell

3.4.1 Formhistogramme für 2D und 3D Objekte

- Ähnlichkeitsmatrix $A = [a_{i,j}]$ enthält die Ähnlichkeit von Einträgen in den Zellen i und j der Raumpartitionierung
- Eintrag $a_{i,j}$ aus Abstand $d_{i,j}$ der Zellen i,j berechnen: $a_{i,j} = e^{-\sigma(d_{i,j}/d_{\max})^2}$
- Verwende z.B. Euklidische Distanz als Abstand $d_{i,j}$
- Experimentelle Untersuchung zur Wahl der Partitionierung
 - Datenbank mit Protein-Molekülen, K-NN (k=1) Anfragen mit jedem einzelnen Protein, Precision als Gütemaß



- Experimentelle Untersuchung: Granularität der Partitionierung



3.4.2 Erweiterungen der Formhistogramme

□ Erweiterung für Voxelisierte Objekte (3D Raum wird gut unterstützt)

[Kriegel, Kröger, Mashael, Pfeifle, Pötke, Seidl. Proc. Int. Conf. Database Systems for Advanced Applications (DASFAA), 2003]

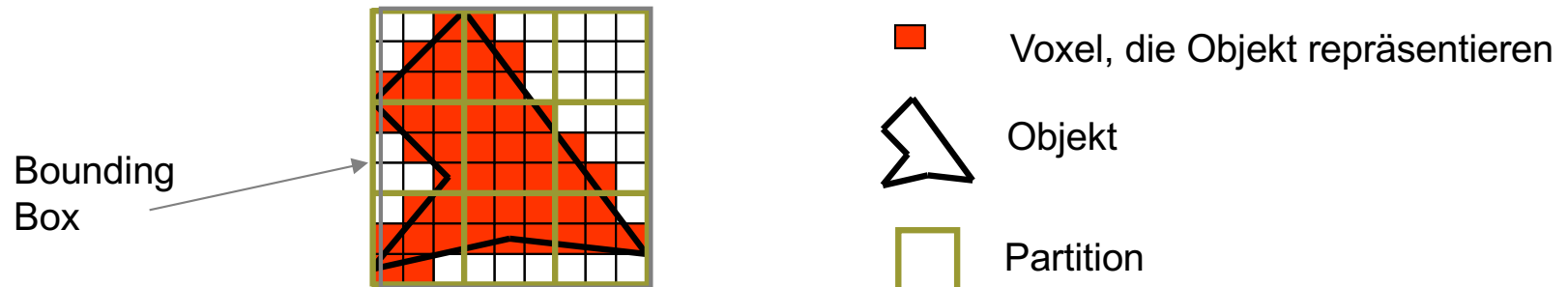
■ Partitionierung

□ Kugelförmige Partitionierung ist bei Voxelmengen nicht sinnvoll

- Voxel können auf Schnittfläche von zwei oder mehr Partitionen liegen
- Bildung von Sektoren im 3D-Raum nicht so einfach realisierbar
- Zu welchen Partitionen sollen diese Voxel hinzugezählt werden?
- Sollen Voxel zu mehreren Partitionen hinzu gezählt werden?

□ Daher: würfelförmige Partitionierung der Bounding Box eines Objekts

- Jedes Voxel liegt in genau einer Zelle
- ACHTUNG: Partitionierung nicht mehr rotationsinvariant



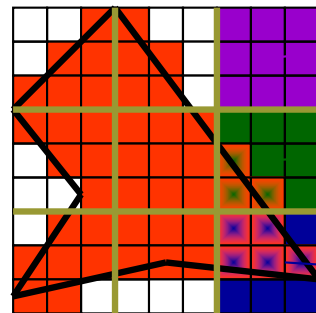
3.4.2 Erweiterungen der Formhistogramme

■ Räumliche Features

□ Volumen Modell (ursprünglicher Ansatz)

- Anzahl der Objekt-Voxel pro Partition
- Normiert mit der Kapazität einer Partition (# aller Voxel pro Zelle)

Kapazität einer Partition: 9 Voxel



$$\begin{pmatrix} \vdots \\ 0/9 \\ \vdots \\ 3/9 \\ \vdots \\ 5/9 \\ \vdots \end{pmatrix}$$

□ Bewertung:

- Einfaches Modell
- Erweiterung: extrahiere andere Features, die die Form des Objekts innerhalb der Zellen beschreibt („Shape Descriptor“):

Solid Angle Wert: beschreibt die Konvexität/Konkavität

Eigenwerte der Hauptachsen: beschreiben die Varianz entlang der Hauptachsen (Objektausrichtung)

3.4.2 Erweiterungen der Formhistogramme

□ Solid Angle Modell

- Voxelisierte Referenzsphäre S_c um Zentrums-Voxel c
- Berechnen für jedes Oberflächen-Voxel v von Objekt o den SA-Wert:

$$SA(v) = \frac{|S_v \cap V^o|}{|S_v|}$$

V^o = Voxelmenge, die Objekt o repräsentiert

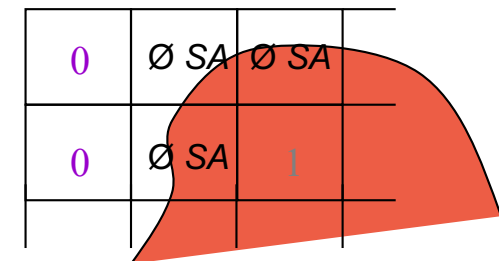
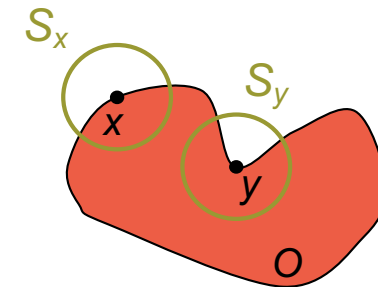
- Berechne für jede Zelle z ein Feature $f(z)$:
 $f(z) = 0$ falls z keine Objekt-Voxel enthält

$f(z) = 1$ falls z nur Voxel aus dem Inneren des Objekts enthält

$$f(z) = \frac{1}{m} \sum_{i=1}^m SA(v_j) \text{ falls } z \text{ } m \text{ Oberflächen-Voxel } v_i \text{ des}$$

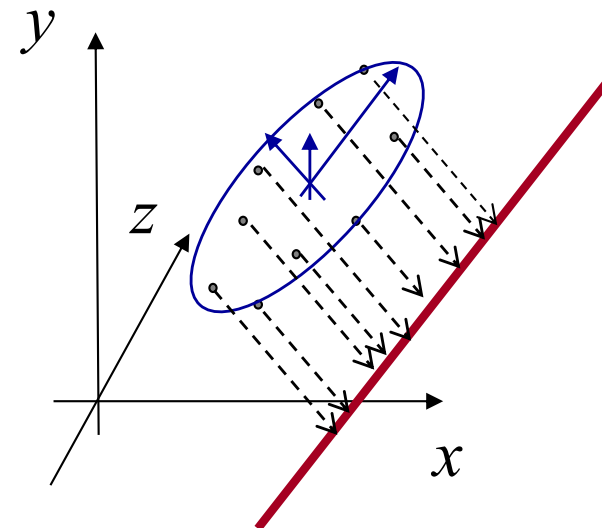
Objekts enthält

(durchschnittlicher SA-Wert aller Oberflächenvoxel in z)



3.4.2 Erweiterungen der Formhistogramme

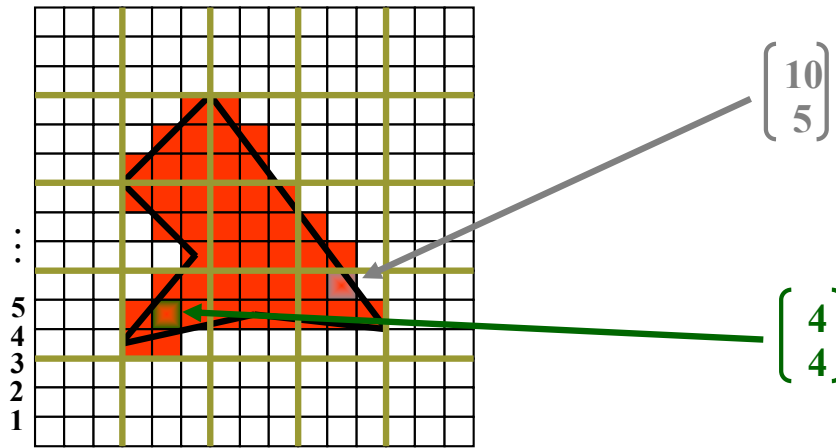
- Eigenwert Modell
 - Hauptachsenanalyse (PCA)
 - Eigenvektoren und Eigenwerte spannen das minimal umgebende Ellipsoid einer Punktmenge auf
 - Eigenvektoren repräsentieren die Hauptachsen (Hauptausdehnungen) der Punktmenge; stehen senkrecht aufeinander
 - Eigenwerte modellieren die Streuung der Punktmenge entlang dieser Hauptachsen
 - Extrahiere diese Streuung der Voxelmenge innerhalb einer Zelle als Feature



3.4.2 Erweiterungen der Formhistogramme

- Modelliere jedes Voxel v des Objekts o als Vektor

$$\vec{v}^o = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$



V_i^o = Voxelmenge in Zelle i , die Objekt o repräsentiert

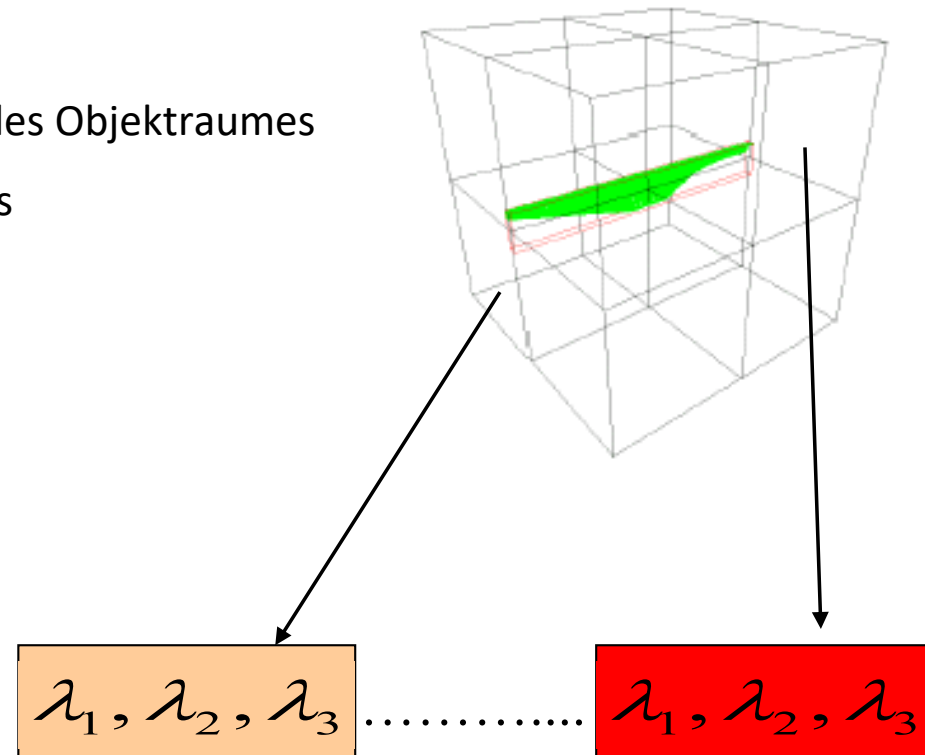
$$\text{Cov}_i^o = \frac{1}{|V_i^o| - 1} \begin{pmatrix} \sum_{j=1}^{|V_i^o|} x_j^2 & \sum_{j=1}^{|V_i^o|} x_j y_j & \sum_{j=1}^{|V_i^o|} x_j z_j \\ \sum_{j=1}^{|V_i^o|} x_j y_j & \sum_{j=1}^{|V_i^o|} y_j^2 & \sum_{j=1}^{|V_i^o|} y_j z_j \\ \sum_{j=1}^{|V_i^o|} x_j z_j & \sum_{j=1}^{|V_i^o|} y_j z_j & \sum_{j=1}^{|V_i^o|} z_j^2 \end{pmatrix}$$

- PCA: Kovarianzmatrix von Voxeln des Objekts o in Zelle i

3.4.2 Erweiterungen der Formhistogramme

- Bestimmung der Eigenwerte $\lambda_1, \dots, \lambda_d$ der Kovarianzmatrix
- Für 3D Objekte 3 Eigenwerte
- Allgemein: d = Dimensionalität des Objektraumes
- Pro Zelle z : extrahiere d Features

$$f(z) = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_d \end{pmatrix}$$



- Vergleich: Bei k Zellen

- Volumen Modell: d -dimensionales Objekt entspricht k -dimensionalem Vektor
- Solid Angle Modell: d -dimensionales Objekt entspricht k -dimensionalem Vektor
- Eigenwert Modell: d -dimensionales Objekt entspricht $(d \cdot k)$ -dimensionalem Vektor