

2.4 Geometrische Ähnlichkeitsmodelle

Einführung und Übersicht

Formen können in Pixelbildern vorliegen, aber auch durch Vektorgraphiken beschrieben werden (z.B. Polygone oder allgemeine Punktmengen).

- Ähnlichkeitsmodelle für Polygone (2D)
 - Approximation von Formen durch größensortierte Rechtecke [Jag 91]
 - Ähnlichkeit von Kantenzügen [GM 93] [MG 93] [MG 95]
 - Partielle Ähnlichkeitssuche per Fourier-Transformation [BKK 97a] [BK 97]
 - Angular Profile, LWL-Codierung (Länge-Winkel-Länge) [BMH 92] [BK 97]
 - Section Coding [BKK 97b] [BK 97]
- Vektorgrafikbasierte Ähnlichkeitsmodelle für 3D-Objekte
 - Formhistogramme [AKKS 99]
 - Approximationsbasierte Ähnlichkeit für Oberflächensegmente [KSS 97] [KS 98]
 - Geometrisches Hashing [LW 88]
 - Iterative Closest Points (ICP) [BM 92] [Zha 94]
 - Algebraische Moment-Invarianten [TC 91]

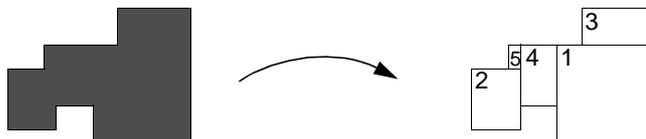
Wir betrachten im weiteren eine Auswahl aus diesen Techniken.

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

2.4.1 Überdeckungsmodell für 2D-Formen

[Jag 91] Jagadish H. V.: *A Retrieval Technique for Similar Shapes*. Proc. ACM Int. Conf. on Management of Data (SIGMOD) 1991, 208-217.

- Grundidee
 - Ähnlichkeitsmodell für 2D-Formen, hier: leicht erweiterbar auf 3D-Formen.
 - Distanzfunktion: Flächeninhalt der symmetrischen Differenz zweier Formen.
 - Hier: Translations- und skalierungsinvariant, nicht jedoch rotationsinvariant.
 - Vorgehen: Repräsentation der Formen durch rechteckige Überdeckungen.
 - Speicherung der Rechtecksflächenmaßzahlen z.B. der Größe nach geordnet.



- Objektmodell
 - Formen sind als konturierte Objekte gegeben (d.h. Polygone).
 - Extraktion von Formen aus Grauwertbildern möglich, solange klare Konturen bestimmt werden können (Probleme z.B. bei teilweise verdeckten Objekten).
 - Die Polygone müssen nicht konvex sein (d.h. Einbuchtungen möglich).

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

Rechtecksüberdeckungen

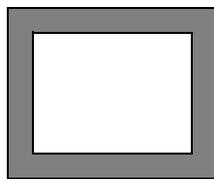
- *Additive Überdeckung*: Durch eine Folge von Rechtecken $\langle R_1, R_2, \dots, R_k \rangle$ ist eine Folge von additiven Überdeckungen $\langle C_0, C_1, \dots \rangle$ wie folgt definiert:

$$C_0 = \emptyset, \quad C_{i+1} = C_i \cup R_{i+1}$$

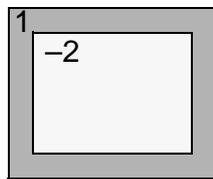
- *Allgemeine Überdeckung*: Neben dem Hinzufügen von Rechtecksflächen (\cup) ist auch das Entfernen von Rechtecksflächen ($-$) möglich:

$$C_0 = \emptyset, \quad C_{i+1} = C_i \cup R_{i+1} \quad \text{oder} \quad C_{i+1} = C_i - R_{i+1}$$

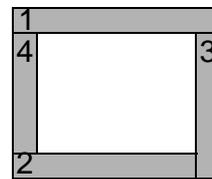
- Für endliche Formen S konvergieren (additive) Überdeckungssequenzen schon im Endlichen, d.h. es gibt ein K , so dass $C_K = S$, und wir definieren $C_j = C_K$ für $j \geq K$.
- Überlappungen sind erlaubt, sollen aber möglichst gering ausfallen.



gegebene Form



allgem. Überdeckung



additive Überdeckung

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

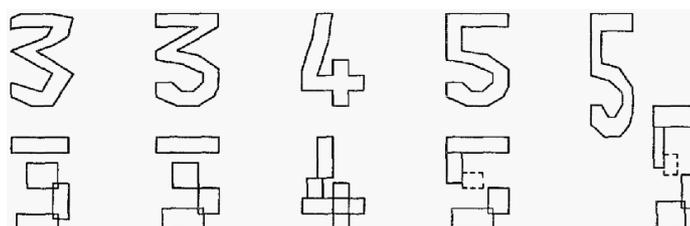
- *Approximative Rechtecksüberdeckungen*
 - Anstatt aller Rechtecke einer Überdeckung werden nur wenige gespeichert.
 - Das Entfernen kleiner Rechtecke entspricht dem Beseitigen hochfrequenter Fehler wie Schmutzflecken oder Diskretisierungsfehlern (z.B. bei eingescannten Bildern).

- *Approximationsqualität*

- Die ersten Rechtecke einer Überdeckung sollen schon eine möglichst gute Approximation der ursprünglichen Form liefern.
- *Kumulatives Fehlerkriterium*: Die Approximationsfehler der Überdeckungssequenz $\langle C_0, C_1, \dots, S \rangle$ werden sukzessive aufsummiert, die Gesamtsumme zählt:

$$\text{kumulativer Fehler} = \sum_{i=1 \dots n} |S - C_i|$$

- Minimierung der Gesamtsumme führt zu Minimierung der "frühen" Fehler $|S - C_i|$ für kleine i , da diese mehrfach gewertet werden.
- Beispiel: Fünf Ziffern und zugehörige erste Rechtecke:

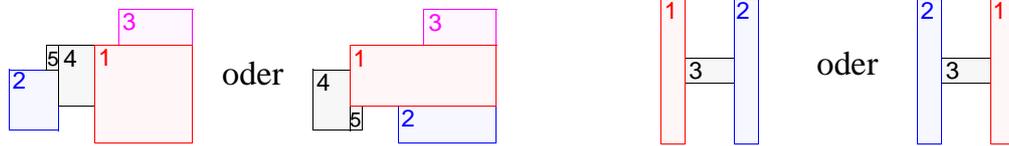


Quelle: [Jag 91]

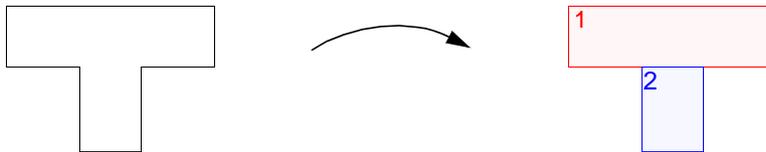
Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

Probleme der Rechtecksüberdeckung

- Nicht-eindeutige Repräsentation
 - Es kann unterschiedliche optimale Zerlegungen eines Objektes geben.
 - Insbesondere bei Symmetrie ist die Reihenfolge der Rechtecke nicht eindeutig.
 - Lösung: Objekt mehrfach speichern oder mehrfache Anfragen für eine Form.



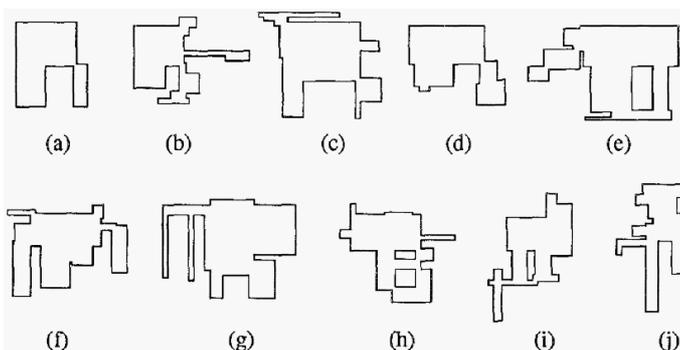
- Rechteckige Formen
 - Wird eine Form schon durch wenige Rechtecke exakt beschrieben, besteht die Überdeckungssequenz ggf. aus weniger Elementen, als im Index gespeichert werden.
 - Lösung: Parameter der weiteren Rechtecke nicht als Punkte, sondern als Bereiche im Feature-Raum speichern -> räumliche Indexe (statt Punktindexe) verwenden.



Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

Ähnlichkeitsanfragen

- Experimentelles Umfeld
 - Datenbank: 16.000 synthetische Formen.
 - Jede Form wurde aus 10 zufällig erzeugten Rechtecken zusammengesetzt, danach wurden jeweils additive Überdeckungen berechnet.
 - Im Index wurden jeweils die größten drei Rechtecke der Überdeckung gespeichert.
 - Anfragen: Bereichsanfragen um zufällig ausgewählte Formen der Datenbank
- Beispiel für das Ergebnis einer Ähnlichkeitsanfrage:



(a) Anfrageform
(b-j) Ergebnisformen

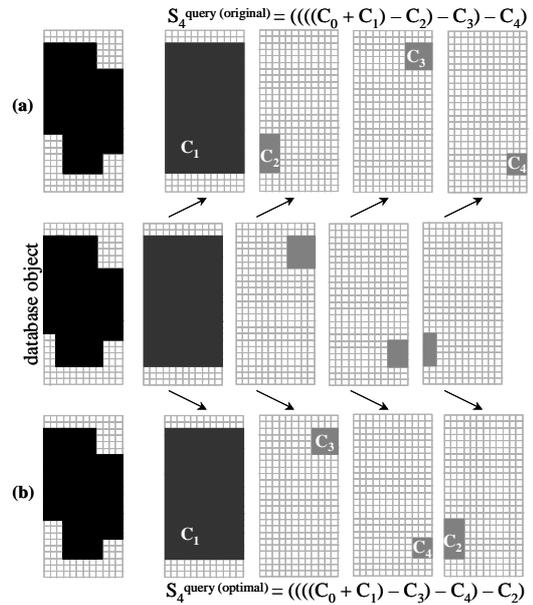
Quelle: [Jag 91]

2.4.2 Erweiterung auf 3D CAD-Daten

[KBK+03] Kriegel H.-P., Brecheisen S., Kröger P., Pfeifle M., Schubert S.: *Using Sets of Feature Vectors for Similarity Search on Voxelized CAD Data*. ACM Int. Conf. on Management of Data (SIGMOD), 2003, 587-598.

- Motivation

- Ähnlichkeitsmodell für 3D-CAD Daten basierend auf Überdeckungsmodell von Jagadish.
- Überdeckungsansatz kann sehr leicht auf voxelisierte 3D Objekte übertragen werden.
- Ziel: Größere Flexibilität beim Vergleich einzelner Überdeckungen innerhalb einer Überdeckungssequenz.



- Umsetzung

- Objekt wird nicht durch einen großen Feature-Vektor repräsentiert, sondern durch eine Menge von Feature Vektoren. Jede Überdeckung wird zu einem Feature-Vektor => Überdeckungssequenz wird zu einer Menge von Feature Vektoren.

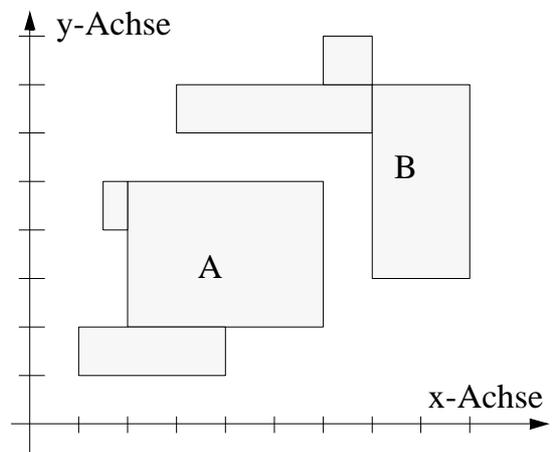
Punktmengenrepräsentation

- Jedes 3D Cover C wird durch einen 6-dimensionalen Feature-Vektor repräsentiert.

$$\begin{aligned}
 f^1 &= x\text{-position of } C \\
 f^2 &= y\text{-position of } C \\
 f^3 &= z\text{-position of } C \\
 f^4 &= x\text{-extension of } C \\
 f^5 &= y\text{-extension of } C \\
 f^6 &= z\text{-extension of } C
 \end{aligned}$$

- 2D Beispiel:

$$\begin{aligned}
 A &= \{(1,1,0,3,1,0), (2,2,0,4,3,0), (1.5,4,0,0.5,1,0)\} \\
 B &= \{(7,3,0,2,4,0), (3,6,0,4,1,0), (6,7,0,1,1,0)\}
 \end{aligned}$$



- Andere Anwendungsgebiete für Punktmengenansatz:

Sport-Manschaften: Jeder Sportler wird durch einen Feature-Vektor repräsentiert, die Mannschaft durch Menge von Vektoren.

Ehepaare: Jeder Partner wird durch einen Feature-Vektor repräsentiert, das Ehepaar durch ein Paar von Vektoren.

Galaxien: Jeder Stern wird durch einen Feature-Vektor repräsentiert, die Galaxie durch Menge von Vektoren.

CD-Sammlung: Jede CD wird durch einen Feature-Vektor repräsentiert, die CD-Sammlung durch Menge von Vektoren.

...

Abstandsmaß auf Punktmengen

Definition 1 (Mengen Aufzählung).

Sei S eine endliche Menge von beliebigen Elementen, und π eine Abbildung die jedem $s \in S$ eine eindeutige Nummer $i \in \{1, \dots, |S|\}$ zuordnet: $\pi(S) = (s_1, \dots, s_{|S|})$. Die Menge aller möglichen Aufzählungen von S wird mit $\Pi(S)$ bezeichnet.

Definition 2 (Minimale Zuordnungsdistanz).

Sei $V \subset \mathbb{R}^d$ und $dist: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ eine Distanzfunktion zwischen zwei d -dimensionalen Feature-Vektoren. Sei $X = \{\vec{x}_1, \dots, \vec{x}_{|X|}\}$ und $Y = \{\vec{y}_1, \dots, \vec{y}_{|Y|}\} \in 2^V$ zwei Vektormengen. Wir nehmen o.B.d.A. an, dass $|X| \leq |Y| \leq k$ sei. Weiterhin, sei $w: V \rightarrow \mathbb{R}$ eine Gewichtsfunktion für die nicht zugeordneten Punkte. Dann ist die minimale Zuordnungsdistanz (*minimal matching distance*) $d_{mm}^{dist, w}: 2^V \times 2^V \rightarrow \mathbb{R}$ wie folgt definiert:

$$d_{mm}^{dist, w}(X, Y) = \min_{\pi \in \Pi(Y)} \left(\sum_{i=1}^{|X|} dist(\vec{x}_i, \vec{y}_{\pi(i)}) + \sum_{i=|X|+1}^{|Y|} w(\vec{y}_{\pi(i)}) \right)$$

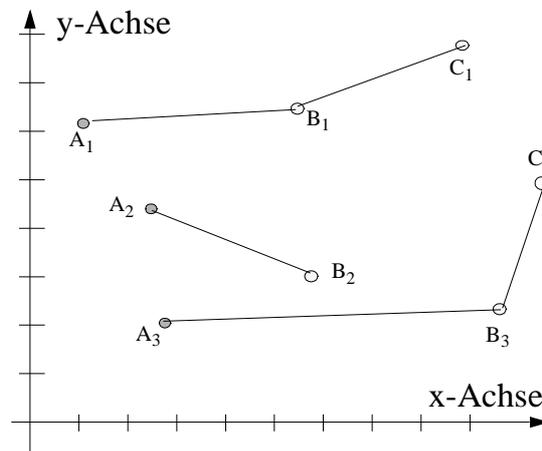
Beispiele für *Gewichtsfunktionen* die zu Metriken bei der minimalen Zuordnungsdistanz führen, basieren auf Dummy Vektoren.

Definition 3 (Dummy Vectors).

Sei $V \subset \mathbb{R}^d$. Sei $\|\vec{x} - \vec{y}\|_2$ die Euclidische Distanz zwischen $\vec{x}, \vec{y} \in \mathbb{R}^d$. Weiterhin, sei $\vec{w} \in \mathbb{R}^d \setminus V$ ein "dummy" Vektor. Dann bezeichnet $w_{\vec{w}}: V \rightarrow \mathbb{R}: w_{\vec{w}}(x) = \|\vec{x} - \vec{w}\|_2$ eine Menge von Gewichtsfunktionen basierend auf dummy Vektoren.

Ein sinnvoller dummy-Wert für unsere Applikation ist der Null-Vektor.

Beispiel:



$$d_{mm}^{dist, w}(A, B) = dist(A_1, B_1) + dist(A_2, B_2) + dist(A_3, B_3)$$

$$d_{mm}^{dist, w}(B, C) = dist(B_1, C_1) + dist(B_3, C_2) + w(B_2)$$

Centroid Filter

- Filter helfen Ähnlichkeitsanfragen zu beschleunigen.
- Für Punktmengen gibt es effektiven Filter basierend auf deren Centroid.

Definition 4 (Erweiterter Centroid).

Sei $V \subset \mathbb{R}^d$ und $\vec{w} \in \mathbb{R}^d \setminus V$. Sei $X \in 2^V$ eine Vektormenge mit $|X| \leq k$ und $X = \{\vec{x}_1, \dots, \vec{x}_{|X|}\}$. Dann ist der erweiterte Centroid $C_{k, \vec{w}}(X)$ wie folgt definiert :

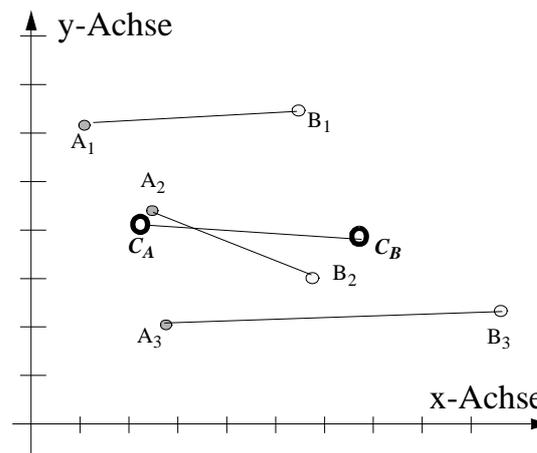
$$C_{k, \vec{w}}(X) = \frac{\sum_{i=1}^{|X|} \vec{x}_i + (k - |X|) \cdot \vec{w}}{k}$$

Satz (Centroid Filter).

Sei $V \subset \mathbb{R}^d$ und $\vec{w} \in \mathbb{R}^d \setminus V$. Sei $w_{\vec{w}}: V \rightarrow \mathbb{R}$, $w_{\vec{w}}(\vec{x}) = \|\vec{x} - \vec{w}\|_2$, eine Gewichtsfunktion für die nicht zugeordneten Knoten. Weiterhin, sei $X, Y \in 2^V$, $X = \{\vec{x}_1, \dots, \vec{x}_{|X|}\}$, $Y = \{\vec{y}_1, \dots, \vec{y}_{|Y|}\}$ zwei Vektormengen mit $|X|, |Y| \leq k$. $C_{k, \vec{w}}(X), C_{k, \vec{w}}(Y)$ seien die erweiterten Centroide, und $d_{mm}^{L_2, w_{\vec{w}}}: 2^V \times 2^V \rightarrow \mathbb{R}$ die minimale Zuordnungsdistanz mit $w_{\vec{w}}$ als Gewichtsfunktion auf V . Dann gilt die folgende Aussage:

$$k \cdot \|C_{k, \vec{w}}(X) - C_{k, \vec{w}}(Y)\|_2 \leq d_{mm}^{L_2, w_{\vec{w}}}(X, Y)$$

Beispiel:

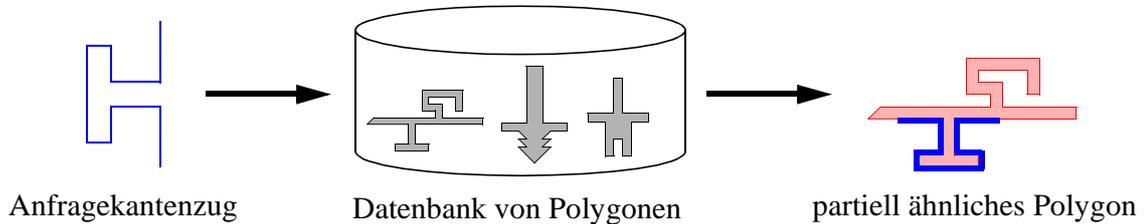


$$3 \cdot \|C_A - C_B\|_2 = \|A_1 - B_1\|_2 + \|A_2 - B_2\|_2 + \|A_3 - B_3\|_2$$

2.4.3 Partielle Ähnlichkeitssuche für Polygone

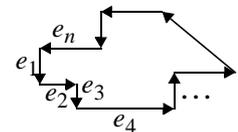
[BKK 97] Berchtold S., Keim D. A., Kriegel H.-P.: *Using Extended Feature Objects for Partial Similarity Retrieval*. VLDB Journal 6(4), 1997, 333-348.

- Ziel
 - Translations-, rotations- und skalierungsinvariante Ähnlichkeit von Polygonen.
 - Unterstützung von partieller Ähnlichkeitssuche.



- Polygondarstellung
 - Ein Polygon ist ein geschlossener Kantenzug $\langle e_1, \dots, e_n \rangle$.
 - Polygon wird als parametrische Kurve $p(t)$ dargestellt, wobei t von 0 bis 2π läuft.
 - Da der Kantenzug geschlossen ist, gilt:

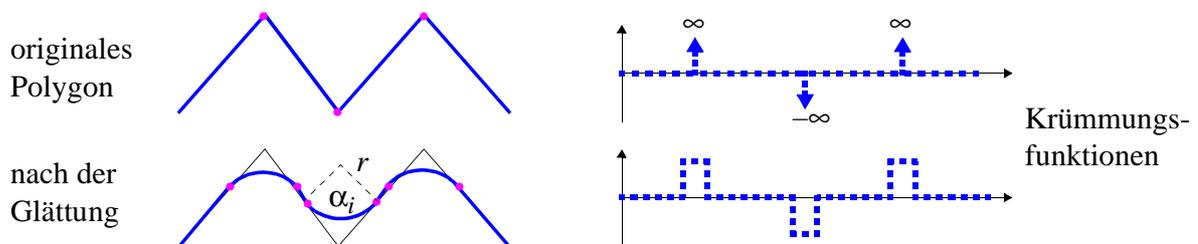
$$p(2\pi) = \int_{0..2\pi} p(t) dt = \sum_{i=1..n} e_i = p(0) = \mathbf{0}$$



- Die Länge des Kantenzuges ist auf 2π normiert: $\sum_{i=1..n} |e_i| = 2\pi$
- Über zwei Parameter a und b lassen sich Ausschnitte aus Polygonen beschreiben.

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

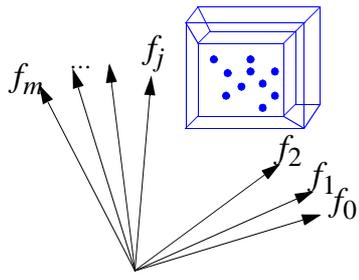
- Invariante Darstellungen
 - Skalierungsinvarianz: Durch Normierung auf Länge 2π erreicht.
 - Translationsinvarianz: Die Polygone sind nicht absolut positioniert.
 - Rotationsinvarianz: Betrachte statt der konkreten Richtungen die Krümmungen!
 - Invarianz gegenüber gewähltem Startpunkt: In partieller Ähnlichkeit enthalten.
- Repräsentation der Krümmungen
 - Problem: Kantenzug ist nicht stetig differenzierbar (scharfe Knicke an den Ecken).
 - Lösung: Glättung durch Approximation der Ecken mit Kreisabschnitten.



- Die Amplituden der Krümmungsfunktion hängen vom gewählten Radius r ab.
- Die Breite eines Ausschlags in der Krümmungsfunktion hängt vom Winkel α_i ab.
- Für die Krümmungsfunktion eines Polygons werden die Fourier-Koeffizienten gespeichert. Diese lassen sich analytisch berechnen.

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

Behandlung der partiellen Ähnlichkeit

- Repräsentation von Polygonausschnitten.
 - Ausschnitte (a, b) aus Kantenzügen $p(t)$, d.h. $a \leq t \leq b$, werden selbst wieder auf den Bereich $0, \dots, 2\pi$ skaliert. Der Datenraum ist zyklisch für den Fall $b < a$.
 - Für die Polygonausschnitte werden ebenfalls die Krümmungsfunktionen berechnet und deren Fourier-Koeffizienten ermittelt.
 - Für den Featureraum werden einige der Koeffizienten als Dimensionen ausgewählt.
 - Die Ähnlichkeit von Polygonausschnitten wird über eine geeignete Distanzfunktion im hochdimensionalen Featureraum definiert (z.B. p -Norm).
 - Unendlichkeitsproblem
 - Es gibt unendlich viele Ausschnitte (a, b) und damit Punkte im hochdimensionalen Featureraum, die weder alle berechnet noch alle gespeichert werden können.
 - Lösung: Speichere nicht die einzelnen Punkte, sondern jeweils das minimal umgebende Hyperrechteck mehrerer Featurepunkte.
- 
- Dazu verschiedene Strategien, wieviele Featurepunkte zusammengefaßt werden.

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

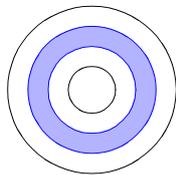
- Zusammenfassen von Featurepunkten
 - Erster Schritt: Zusammenfassen der Featurepunkte für Abschnitte, die auf derselben Kante des Polygons beginnen und auf einer bestimmten anderen Kante enden (d.h. $n \cdot n$ viele Hyperrechtecke).
 - Beobachtung: Manche dieser Boxen sind sehr klein, andere sehr groß.
- Kleine Boxen können weiter zusammengefaßt werden:
 - Die Hyperrechtecke für benachbarte Kanten im Polygon werden zusammengefaßt.
- Große Boxen können weiter zerlegt werden:
 - Zerlegung im Featureraum:
 - Auswahl bestimmter Achsen oder
 - Auswahl aller Achsen: dann gibt es 2^m viele Zerlegungsprodukte.
 - Alternative: Zerlege Boxen im zweidimensionalen Parameterraum, d.h. fasse andere Polygonausschnitte zusammen.
- Anforderungen an eine Indexstruktur
 - Verwaltung hochdimensionaler ausgedehnter Objekte (Hyperrechtecke).
 - Bearbeitung von Punktanfragen für die exakte Suche nach Polygonausschnitten.
 - Bearbeitung von ε -Anfragen für die Suche nach ähnlichen Polygonausschnitten.

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

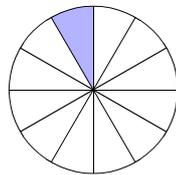
2.4.4 Formhistogramme für 3D-Objekte

[AKKS 99] Ankerst M., Kastenmüller G., Kriegel H.-P., Seidl T.: *3D Shape Histograms for Similarity Search and Classification in Spatial Databases*. Proc. Int. Symposium on Large Spatial Databases (SSD) 1999 (LNCS 1651), 207-226.

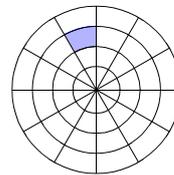
- Ziel
 - Translations- und rotationsinvariante Suche nach ähnlichen Formen im 3D.
 - Objekte sind als Mengen von Oberflächenpunkten gegeben.
 - Beispielanwendungen: Moleküle, CAD-Bauteile.
- Grundidee: Formhistogramme
 - Partitioniere den 3D-Raum in Zellen (Histogramm-Bins).
 - Bestimme den Anteil an Punkten des Objektes pro Zelle (normiertes Histogramm).
 - Durch die Normierung werden die Histogramme unabhängig von der Punktedichte.
- Beispiele
 - Verschiedene Raumpartitionierungen



Schalenmodell



Sektorenmodell



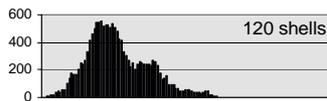
kombiniertes Modell

Quelle: [AKKS 99]

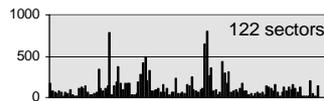
Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

- Beispielobjekt Seryl-tRNA Synthetase (PDB-Code: 1SER-B)

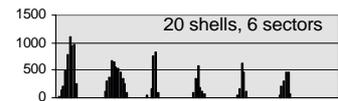
Quelle: [AKKS 99]



Schalenmodell



Sektorenmodell



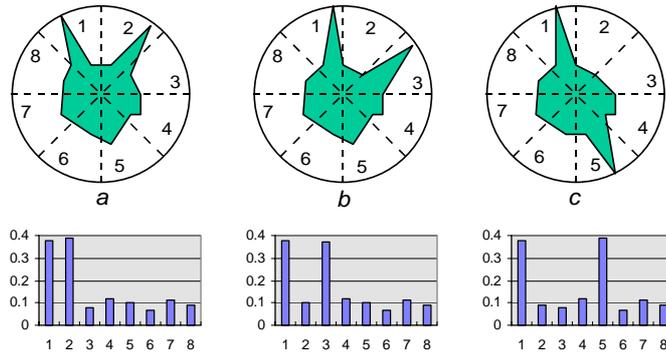
kombiniertes Modell

- Formale Definition der Histogramme
 - *Schalenmodell*: Definiere die Bins über den Abstand zum Mittelpunkt.
 - *Sektorenmodell*: Die Zellen sind Voronoizellen, wobei als Kernpunkte gleichmäßig verteilte Punkte auf der Einheitskugel verwendet werden.
 - *Kombiniertes Modell*: Die Einheits-Kernpunkte werden auf die Schalen projiziert.
- Invarianzen
 - Translationsinvarianz durch Lagenormierung:
Verschiebung des Schwerpunkts eines Objektes in den Ursprung.
 - Rotationsinvarianz durch Hauptachsentransformation:
 - Drehung der Objekte, sodaß die Hauptachsen auf den Koordinatenachsen liegen.
 - unnötig beim Schalenmodell, dieses ist inhärent rotationsinvariant.

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

Distanzfunktion

- Probleme mit dem euklidischen Abstand (Beispiel im 2D):



Quelle: [AKKS 99]

- Die Form *c* gilt als genauso ähnlich zu *a* wie *b* zu *a*.
- Die Ähnlichkeit räumlich benachbarter Histogramm-Bins wird nicht berücksichtigt.

- Quadratische Formen als Distanzfunktionen

$$d_A(p, q) = \sqrt{(p-q) \cdot A \cdot (p-q)^T} = \sqrt{\sum_i \sum_j a_{ij} (p_i - q_i)(p_j - q_j)}$$

- Für die Formhistogramme enthält die Ähnlichkeitsmatrix $A = [a_{ij}]$ die Ähnlichkeit von Einträgen in den Zellen *i* und *j* der Raumpartitionierung.

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

- Diese Ähnlichkeit läßt sich aus dem Abstand d_{ij} der Zellen *i* und *j* berechnen, z.B.:

$$a_{ij} = \exp(-\sigma (d_{ij} / d_{\max})^2)$$

- Als Abstand d_{ij} eignet sich beispielsweise der euklidische Abstand der Kernpunkte.

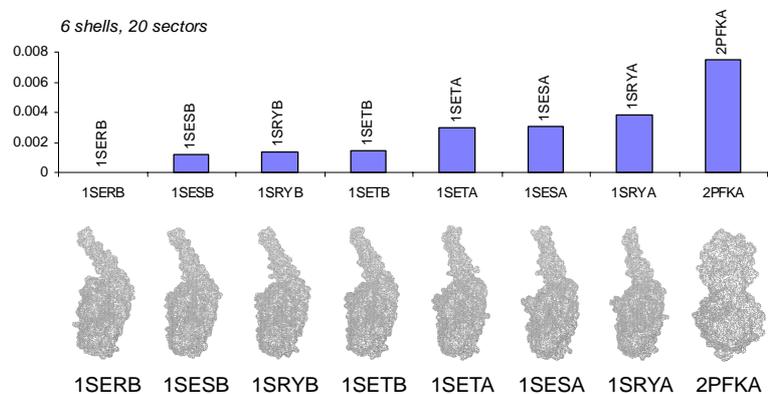
Experimentelle Ergebnisse

- Einfache Suche nach ähnlichen Molekülen

Anfrage: 1SER-B

Die erwarteten Ergebnisse (Seryl-Proteasen) treten auf den ersten Positionen auf.

Das erste nicht-Seryl-Protein (2PFKA) unterscheidet sich in seiner Form sowie in seinem Distanzwert zur Anfrage deutlich von den Seryl-Proteinen.

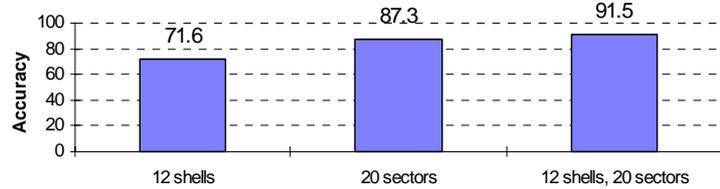


Quelle: [AKKS 99]

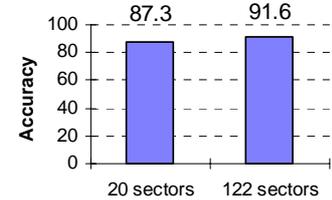
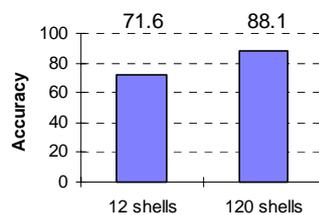
- **Klassifikation auf der gesamten Datenbank**
 - Das Ähnlichkeitsmodell wird in einem nächsten-Nachbar-Klassifikator verwendet.
 - D.h. ein Anfrageobjekt bekommt das Klassenlabel des ähnlichsten Objektes aus der Datenbank zugeordnet.
 - Klassifikationsgenauigkeit: Wie oft wird die Klassenentscheidung richtig getroffen?
 - “Leave-One-Out”-Experiment: jedes Objekt wird gegen die restliche DB angefragt.

- **Ergebnisse:**

- Vergleich verschiedener Histogrammmodelle (12 Schalen, 20 Sektoren, 12×20 Zellen).



- Vergleich verschieden-granularer Raumpartitionierungen (12 bzw. 120 Schalen bzw. 20 bzw. 122 Sektoren).



Quelle: [AKKS 99]

Skript Multimedia-Datenbanksysteme · Modelle der Datenexploration

Referenzen

- [AKKS 99] Ankerst M., Kastenmüller G., Kriegel H.-P., Seidl T.: *3D Shape Histograms for Similarity Search and Classification in Spatial Databases*. Proc. Int. Symposium on Large Spatial Databases (SSD) 1999 (LNCS 1651), 207-226.
- [BK 97] Berchtold S., Kriegel H.-P.: *S3: Similarity Search in CAD Database Systems*. ACM SIGMOD 1997, pp. 564-567.
- [BKK 97a] Berchtold S., Keim D. A., Kriegel H.-P.: *Using Extended Feature Objects for Partial Similarity Retrieval*. VLDB Journal 6(4), 1997, 333-348.
- [BKK 97b] Berchtold S., Keim D.A., Kriegel H.-P.: *Section Coding: Ein Verfahren zur Ähnlichkeitssuche in CAD-Datenbanken*. BTW 1997, 152-171.
- [BM 92] Besl P.J., McKay N.D.: *A Method for Registration of 3-D Shapes*. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 14(2), 1992, 239-256.
- [BMH 92] Badel A., Mornon J. P., Hazout S.: *Searching for Geometric Molecular Shape Complementarity using Bidimensional Surface Profiles*. Journal of Molecular Graphics 10, 1992, 205-211.
- [GM 93] Gary J. E., Mehrotra R.: *Similar Shape Retrieval Using a Structural Feature Index*. Information Systems 18(7), 1993, 525-537.
- [Jag 91] Jagadish H. V.: *A Retrieval Technique for Similar Shapes*. Proc. ACM Int. Conf. on Management of Data (SIGMOD) 1991, 208-217.
- [KSS 97] Kriegel H.-P., Schmidt T., Seidl T.: *3D Similarity Search by Shape Approximation*. Proc. Int. Symposium on Large Spatial Databases (SSD) 1997 (LNCS 1262), 11-28.
- [KS 98] Kriegel H.-P., Seidl T.: *Approximation-Based Similarity Search for 3-D Surface Segments*. GeoInformatica 2(2), 1998, 113-147.
- [LW 88] Lamdan Y., Wolfson H.J.: *Geometric Hashing: A General and Efficient Model-Based Recognition Scheme*. Proc. IEEE Int. Conf. on Computer Vision 1988, 238-249.
- [MG 93] Mehrotra R., Gary J. E.: *Feature-Based Retrieval of Similar Shapes*. Proc. Int. Conf. on Data Engineering (ICDE) 1993, 108-115.
- [MG 95] Mehrotra R., Gary J.E.: *Similar-Shape Retrieval in Shape Data Management*. IEEE Computer 28(9), 1995, 57-62.
- [TC 91] Taubin G., Cooper D. B.: *Recognition and Positioning of Rigid Objects Using Algebraic Moment Invariants*. in *Geometric Methods in Computer Vision*, SPIE 1570, 1991, 175-186.
- [Zha 94] Zhang Z.: *Iterative Point Matching for Registration of Free-Form Curves and Surfaces*. Int. Journal of Computer Vision 13(2), 1994, 119-152.