Proc. Datenbanksysteme in Büro, Technik und Wissenschaft (BTW), Dresden, 1995.

Visualisierungstechniken zur Exploration und Analyse sehr großer Datenbanken

Daniel A. Keim, Hans-Peter Kriegel

Institut für Informatik, Universität München
Leopoldstr. 11B, D-80802 München
e-mail: {keim, kriegel}@informatik.uni-muenchen.de

Zusammenfassung

Unser Ansatz zur Exploration und Analyse sehr großer Datenbanken basiert auf neuartigen Visualisierungstechniken für multidimensionale Daten. Die prinzipielle Idee dabei ist die gleichzeitige Darstellung möglichst vieler Datenobjekte am Bildschirm, wobei jeder Datenwert durch ein Pixel des Bildschirms repräsentiert wird. Die Farbe des Pixels entspricht dem Abstand des jeweiligen Datenwertes zum Anfragewert. Die Anordnung hängt von der Gesamtdistanz des Datensatzes in Bezug auf die Anfrage und von der gewählten Visualisierungstechnik ab. Durch ein graphisches Benutzerinterface kann der Benutzer seine Anfragen inkrementell ändern, wobei er durch das visuelle Feedback, das er bei Änderungen bekommt, in der Verfeinerung seiner Anfragen unterstützt wird. Ein zentrales Anliegen dieses Papiers sind Bewertung und Vergleich unserer Visualisierungstechniken. Bei der Bewertung von Visualisierungstechniken stehen nicht, wie sonst bei Leistungsvergleichen, die CPU-Zeiten oder die Anzahl der Zugriffe auf den Sekundärspeicher im Vordergrund, sondern die Wahrnehmbarkeit von Zusammenhängen und Eigenschaften der Daten. Analog zu den für Leistungsvergleiche von Datenbanksystemen entwickelten Benchmarks werden deshalb für die Evaluierung von Visualisierungstechniken künstlich erzeugte Testdaten mit vorgegebenen Eigenschaften verwendet.

Schlüsselwörter: Datenexploration und -analyse, Data Mining, Visuelle Anfrageunterstützung für Datenbanken, Visualisierung großer Datenmengen, Visualisierung multidimensionaler multivariater Daten

1. Einleitung

Bei Entscheidungen ist es wichtig, im richtigen Augenblick die richtigen Informationen zur Hand zu haben. Durch den schnellen technologischen Fortschritt steigt die Menge an Information, die in gespeicherter Form verfügbar und für die Entscheidungsfindung potentiell von Bedeutung ist, sehr schnell an. Nach neuesten Schätzungen verdoppelt sich die Menge an Information, die weltweit vorhanden ist, alle 20 Monate. Eine Ursache für die ständig ansteigenden Datenmengen ist die Automatisierung fast aller Vorgänge in Wirtschaft, Wissenschaft und Verwaltung. In der heutigen Zeit werden selbst einfache Vorgänge wie das Bezahlen mit Kreditkarte oder das Telefonieren durch Computer erfaßt. Versuchsreihen in Physik, Chemie und Medizin erzeugen große Mengen an Daten, die zumeist automatisch mit Hilfe von Sensoren gesammelt werden. Beobachtungssatelliten werden schon bald täglich Datenmengen im Terabytebereich sammeln und zur Erde übermitteln. Die gesammelten Daten gleichen Heuhaufen, in denen die Stecknadeln wichtiger Informationen versteckt sind. Die großen Mengen gespeicherter Daten stellen eine wichtige Informationsressource dar; es ist in den meisten Fällen aber recht schwer, die relevanten Informationen zu finden.

Die Speicherung großer Datenmengen erfolgt in der Regel mit Hilfe von Datenbanksystemen. Heute verfügbare Datenbanksysteme unterstützen den Benutzer bei der Speicherung und Verwaltung der Daten (RASIS: Reliability, Availability, Security, Integrity, Serviceability) sowie bei der Suche nach exakt spezifizierten Daten. Sie sind im allgemeinen aber ungeeignet, um die unexakt spezifizierte Suche nach interessanten Zusammenhängen sowie besonders 'heißen' Daten, den sog. 'data mining'-Prozeß, zu unterstützen. Zu 'Data Mining' (Datenexploration und Datenanalyse) gehören unter anderem die Suche nach Zusammenhängen, partiellen funktionalen Abhängigkeiten sowie Clustern von Daten mit ähnlichen Eigenschaften. Da in 'Data Mining'-Anwendungen typischerweise a priori nur wenig über die Daten, die Werteverteilung der Attribute und Zusammenhänge zwischen den Attributen bekannt ist, werden neue Anfragemechanismen benötigt. Wichtig sind zum Beispiel die Unterstützung von unscharfen (vage, fuzzy) Anfragen, von Datenanalysetechniken sowie von Techniken, die einen Überblick über die Daten liefern. Zu berücksichtigen sind Forschungsergebnisse aus den Bereichen

- multivariate Statistik - explorative Datenanalyse: Hauptkomponenten-, Faktor- und Cluster-Analyse sowie Multidimensionales Skalieren [DE 82, Hub 85],
- Knowledge Discovery: Decision Tree Inducers und Rule Discovery Techniques [FPM 91],
- Information Retrieval: Approximatives Matching (z.B. Techniken zur Gewichtung der Anfrageteile und zum Rangordnen der Ergebnisse [SB 88, FM 91]),
- Intelligente Datenbank-Benutzerschnittstellen (Kooperative Datenbank-Interfaces [GGM 92], Interfaces für unscharfe Anfragen [ABN 92] und intelligente Browser [Mot 90]).

In den genannten Bereichen wurden in den letzten Jahren beachtliche Ergebnisse erzielt. Mit wenigen Ausnahmen wurden die Techniken jedoch nicht für die Datenexploration und -analyse von sehr großen Datenbanken mit Hunderttausenden oder sogar Millionen von Datensätzen entworfen bzw. adaptiert. Erste Ergebnisse bei der Anwendung einiger Techniken auf großen Datenmengen zeigen, daß die Nutzung der Fähigkeiten des Computers allein nicht ausreichen, um überzeugende Ergebnisse bei der Datenexploration und -analyse zu erzielen. Verfahren, die erfolgreich auf großen Datenmengen arbeiten, nutzen zusätzlich zu der ernormen Verarbeitungsgeschwindigkeit des Computers auch die Fähigkeiten des menschlichen Benutzers, der beim Knowledge Discovery beispielsweise ein Ziel vorgeben oder beim Information Retrieval die Ergebnisse des ersten Suchlaufes bewerten kann.

Eine effektive Unterstützung von Datenexploration und -analyse ist derzeit nur unter Einbeziehung des Menschen und seiner Fähigkeiten möglich. Insbesondere die unübertroffenen Fähigkeiten der Wahrnehmung erlauben es dem Menschen, in kürzester Zeit komplexe Sachverhalte zu analysieren, wichtige Informationen zu erkennen und Entscheidungen zu treffen. Das menschliche Wahrnehmungssystem kann flexibel die verschiedensten Arten von Daten verarbeiten, wobei es automatisch ungewöhnliche Eigenschaften erkennt, bekannte Eigenschaften dagegen ignoriert. Menschen können leichter und besser mit vagen Beschreibungen und unscharfem Wissen umgehen als heutige Systeme, und ihr Allgemeinwissen erlaubt es ihnen, ohne geistige Anstrengung komplexe Schlußfolgerungen zu ziehen.

Das Ziel unseres Ansatzes der Datenexploration und -analyse ist deshalb, den Menschen in den 'Data Mining'-Prozeß mit einzubeziehen und seine Fähigkeiten auf die großen, in heutigen Computersystemen verfügbaren Datenbestände anzuwenden. Da weder Mensch noch Computer allein das Problem der Datenexploration sehr großer Datenbanken lösen kann, ist eine möglichst enge Kooperation zwischen Mensch und Computer erforderlich. Es gilt, die immense Speicherkapazität und Rechenleistung heutiger Computer mit Flexibilität, Kreativität und Allgemeinwissen des Menschen zu vereinen. Dabei ist die Entwicklung von Techniken wichtig, die den Menschen nicht einfach mit Daten überhäufen, sondern einen guten Überblick über die Daten ermöglichen. In diesem Zusammenhang müssen neue Darstellungsformen für große Mengen multidimensionaler Daten entwickelt werden, wobei alle Eigenschaften der visuellen Repräsentation, wie z.B. Anordnung und Farbe, zu berücksichtigen sind.

Unser Ansatz zur Datenexploration und -analyse großer Datenbanken basiert auf neuartigen Visualisierungstechniken für multidimensionale Daten. Die prinzipielle Idee ist die gleichzeitige Darstellung möglichst vieler Datenobjekte am Bildschirm, wobei jeder Datenwert durch ein Pixel des Bildschirms repräsentiert wird. Die Farbe des Pixels entspricht dem Abstand des jeweiligen Datenwertes zum Anfragewert. Die Anordnung hängt von der Gesamtdistanz des Datensatzes in Bezug auf die Anfrage und von der gewählten Visualisierungstechnik ab. Durch ein graphisches Benutzerinterface kann der Benutzer seine Anfragen inkrementell ändern, wobei er durch das visuelle Feedback, das er bei Änderungen bekommt, in der Verfeinerung seiner Anfragen unterstützt wird.

Da Datenvisualisierungstechniken nicht als allgemein bekannt vorausgesetzt werden können, soll im zweiten Abschnitt zunächst eine kurzer Überblick über Visualisierungstechniken für multidimensionale, multivariate Daten gegeben werden. In [KKS 94] wurde die prinzipielle Idee unserer Visualisierungstechnik sowie das interaktive Anfrage- und Visualisierungsinterface des ersten Prototypsystems vorgestellt. Inzwischen liegt eine vollständige Reimplementierung des Systems in C++ / MOTIF vor, die unter X-Windows auf HP 7xx Maschinen läuft. Die derzeitige Version ist um zusätzliche Visualisierungstechniken erweitert worden und ermöglicht einen direkten Vergleich der Techniken. Im dritten Abschnitt werden die Visualisierungstechniken, die durch die derzeitigen Version des Systems unterstützt werden, kurz vorgestellt. Der vierte Abschnitt beinhaltet einen detaillierten Vergleich sowie eine Bewertung der Visualisierungstechniken. Bei der Bewertung von Visualisierungstechniken stehen nicht, wie sonst bei Leistungsvergleichen, die CPU-Zeiten oder die Anzahl der Zugriffe auf den Sekundärspeicher im Vordergrund, sondern die Wahrnehmbarkeit von Eigenschaften der Daten. Für den Vergleich werden künstlich erzeugte Daten mit spezifischen vorgegebenen Eigenschaften verwendet. Abschnitt fünf faßt die Ergebnisse zusammen und erläutert zukünftige Forschungsvorhaben.

Bei unseren Betrachtungen gehen wir zunächst von einer einfachen Strukturierung der Daten, wie sie beim relationalen Modell vorhanden ist, aus. Dies ist für einen großen Teil der betrachteten Anwendungen adäquat, da sehr große Datenmengen heute zumeist mit Hilfe relationaler Systeme verwaltet werden. Unsere Visualisierungstechniken eignen sich jedoch ebenso für die Visualisierung großer Datenmengen, die in objekt-orientierten oder anderen Datenbanken gespeichert sind.

2. Visualisierung multidimensionaler Daten

In vielen Bereichen von Forschung und Industrie werden Visualisierungen von Daten, die eine inhärente zwei- oder drei-dimensionale Semantik haben, verwendet. Eine Übersicht über solche Techniken ist beispielsweise in den bekannten Büchern von Edward R. Tufte [Tuf 83, Tuf 90] zu finden. Bis vor kurzem gab es jedoch nur wenige Techniken, die eine Visualisierung multidimensionaler Daten ohne inhärente zwei- oder drei-dimensionale Semantik erlauben. Erste Ansätze sind Matrizen von X-Y-Diagrammen [And 72, Cle 93], die Chernoff'sche Gesichter-Darstellung [Che 73], und andere [And 57, Bri 79]. Durch die zunehmende Verfügbarkeit von Grafik-Workstations mit hoher Rechenleistung wurden in den letzten Jahren zahlreiche neue

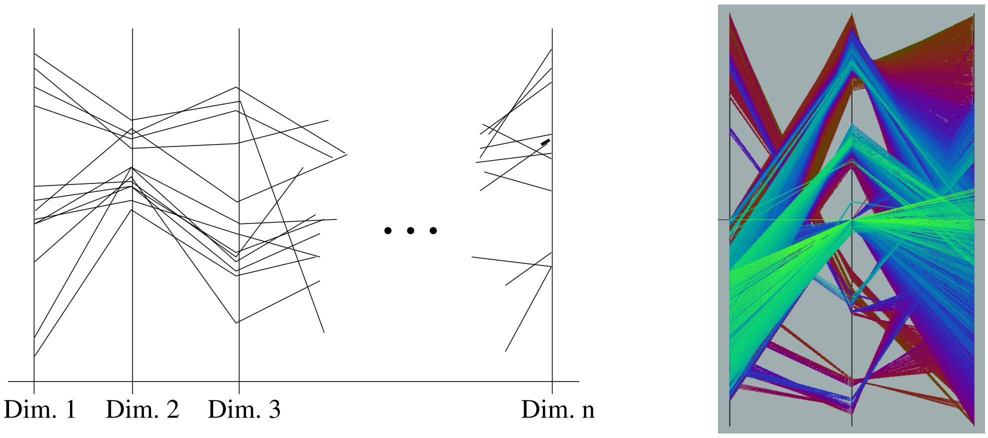


Abb. 1: Technik der Parallelen Koordinaten

Visualisierungstechniken entwickelt. In den bisher untersuchten Ansätzen ist die Anzahl der gleichzeitig visuell darstellbaren Datensätze noch stark begrenzt (100 - 1.000 Datensätze). Um einen Einblick in das Gebiet der Visualisierung multidimensionaler Daten zu geben, sollen im folgenden beispielhaft einige Techniken, die für die Visualisierung von Datenbanken geeignet sind, vorgestellt werden.

2.1 Geometrische Projektionen

Das Ziel geometrischer Projektionstechniken ist es, aussagekräftige Projektionen multidimensionaler Daten zu finden. Die Klasse der geometrischen Projektionen umfaßt Techniken aus der Statistik wie z.B. Hauptkomponenten-Analyse, Faktor-Analyse und multidimensionales Skalieren, die auch unter dem Begriff 'projection pursuit' zusammengefaßt werden [FT 74, Hub 85]. Da die Anzahl der Möglichkeiten, multidimensionale Daten mit hoher Dimension auf zwei Dimensionen abzubilden, sehr groß ist, versuchen 'projection pursuit'-Systeme (z.B. Grand Tour System [Asi 85]), automatisch aussagekräftige Projektionen zu finden oder wenigstens den Benutzer bei der Suche nach geeigneten Projektionen zu unterstützen.

Eine andere geometrische Projektionstechnik ist die Technik der Parallelen Koordinaten (parallel coordinates) [Ins 85, ID 90]. Diese Technik stellt den k-dimensionalen Raum mit Hilfe von k äquidistanten Achsen dar, die parallel zu einer der Bildschirmachsen liegen. Die Achsen entsprechen den Dimensionen und sind vom Minimum- bis zum Maximumwert der Dimensionen linear skaliert. Jeder Datensatz wird als polygonale Linie dargestellt, die jede Achse an dem Punkt schneidet, dessen Wert der jeweiligen Dimension entspricht (vgl. Abb. 1). Obwohl die Grundidee der 'Parallelen Koordinaten'-Technik einfach ist, ermöglicht sie das Erkennen eines weiten Spektrums von Datencharakteristika, wie z.B. verschiedene Datenverteilungen und funktionale Abhängigkeiten. Wegen der Überlappungen der Linien ist jedoch die Anzahl der Datensätze, die gleichzeitig visuell darstellbar ist, auf ca. 1.000 begrenzt.

2.2 Pixeldiagramme (Iconic Displays)

Eine andere Technik zur visuellen Darstellung multidimensionaler Daten sind Pixeldiagramme (iconic displays), bei denen jedes multidimensionale Datenelement durch ein Icon dargestellt wird. Erste Ansätze der 'iconic display' Technik sind die bereits erwähnten Chernoff'schen Gesichter [Che 73, Tuf 83], bei denen zwei Dimensionen durch die zwei Bildschirmdimensionen und die restlichen Dimensionen durch Merkmale des Gesichts (Form von Nase, Mund, Augen und des Gesichts selbst) dargestellt werden (vgl. Abb. 2). Die Chernoff'sche Visualisierungstechnik basiert auf der Fähigkeit des Menschen, Gesichter bzw. Gesichtszüge zu unterscheiden.

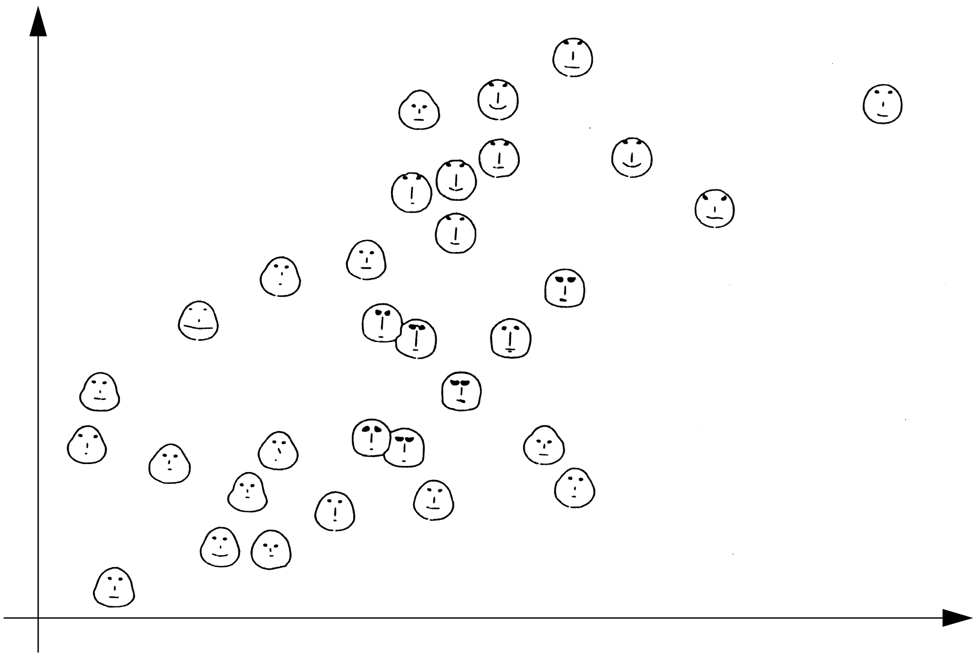
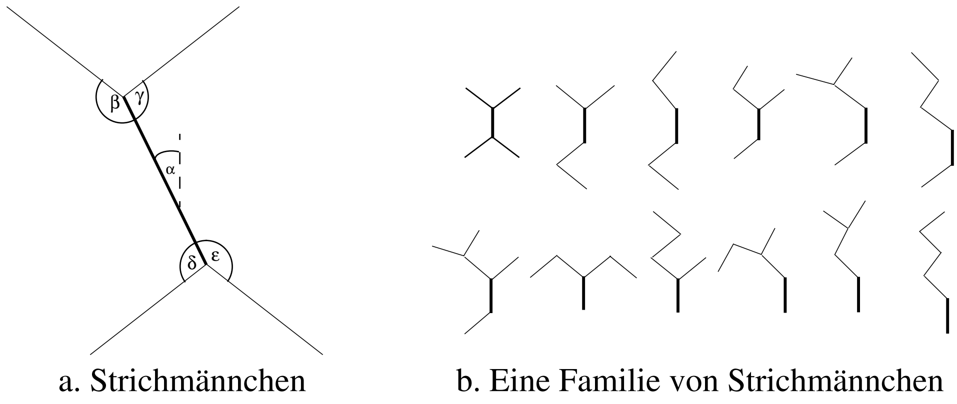


Abb. 2: Chernoff'sche Gesichterdarstellung (vgl. [Tuf 83])

Eine weitere bekannte Pixeldiagramm-Visualisierungstechnik ist die sog. Strichmännchen (stick figure)-Technik [Pic 70, PG 88]. Wie der Name bereits sagt, sind die verwendeten Icons eine Art Strichmännchen, wobei die Winkel und Strichlängen die Datendimensionen repräsentieren. Wenn die Datensätze im Bezug auf die Bildschirmdimensionen verhältnismäßig dicht beieinander liegen, zeigt die resultierende Visualisierung Strukturmuster, die gemäß der Datencharakteristika variieren. Als Strichmännchen können verschiedene Icons mit unterschiedlicher Dimensionalität verwendet werden (vgl. Abb. 3). In Abb. 4 ist eine Visualisierung von fünfdimensionalen Infrarot-Bildern der östlichen Großen Seen in den Vereinigten Staaten dargestellt, die mit Hilfe der Strichmännchen-Technik generiert wurde. An dieser Stelle sei angemerkt, daß sowohl bei der Strichmännchen-Technik als auch bei den Chernoff'schen Gesichtern die Anzahl der gleichzeitig darstellbaren Dimensionen begrenzt ist.

Anders ist dies bei der sog. 'shape coding' Technik [Bed 90]. Bei der 'shape coding' Technik wird jeder Dimension ein kleines Pixel-Array zugeordnet, wobei die Farbe bzw. Graustufe der Pixel dem Wert der Dimension entspricht. Die Pixel-Arrays, die zu den Dimensionen eines Datensatzes gehören, werden dann nacheinander in einem kleinen Quadrat oder Rechteck angeordnet. Die kleinen Quadrate oder Rechtecke, die den Datensätzen entsprechen, werden zeilenweise angeordnet.



a. Strichmännchen b. Eine Familie von Strichmännchen

Abb. 3: Strichmännchen Technik

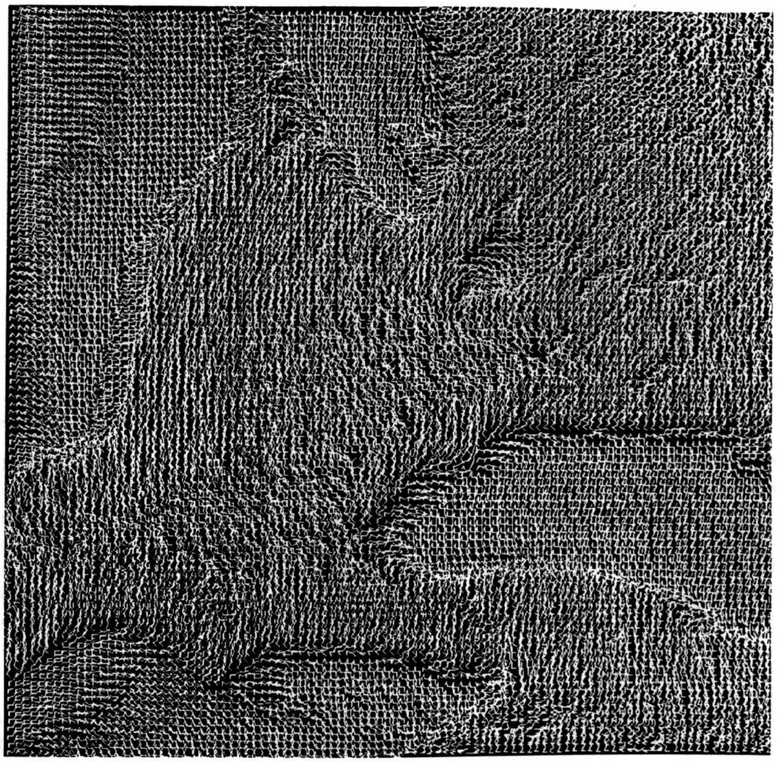


Abb. 4: Strichmännchen-Visualisierung der östlichen Großen Seen (vgl. [SGP 91])

2.3 Hierarchische und Dynamische Techniken

Neben den Geometrischen Projektionen und Pixeldiagrammen gibt es noch zwei weitere Klassen von Visualisierungstechniken - die hierarchischen und dynamischen Techniken. Bei den hierarchischen Techniken sind insbesondere die n-Vision Technik (auch 'worlds within worlds' genannt) [BF 90], das dimensionale Stapeln (dimensional stacking) [LWW 90] und das hierarchische Zeichnen (hierarchical plotting) [MGTS 90] zu nennen. Beispiele für dynamische Techniken sind [MTS 91] und [MZ 92]. Hierarchische Techniken untergliedern den k-dimensionalen Raum und präsentieren ihn in einer hierarchischen Form. Beim dimensionalen Stapeln beispielsweise wird der k-dimensionale Raum in zweidimensionale Teilräume unterteilt. Da hierarchische Techniken hauptsächlich für die Visualisierung mehrdimensionaler Funktionen verwendet werden, unser Anliegen aber die Visualisierung von Datenbankinhalten ist, sollen sie an dieser Stelle nicht weiter behandelt werden.

3. Visualisierungstechniken zur Analyse sehr großer Datenmengen

In den bisher vorgeschlagenen und im letzten Abschnitt erläuterten Visualisierungstechniken für multidimensionale Daten ist die Anzahl der gleichzeitig am Bildschirm darstellbaren Datensätze auf maximal 100 bis 1.000 begrenzt. In diesem Abschnitt sollen kurz die von uns entwikkelten Visualisierungstechniken, die sich auch für sehr große Datenmengen (bis 1.000.000 Datensätze) eignen, vorgestellt werden.

Die Relationen einer relationalen Datenbank können als Mengen von Tupeln der Form (a_1, a_2,..., a_k) angesehen werden, wobei a_1, a_2,..., a_k die Attributwerte eines Datensatzes darstellen. Anfragen an relationale Datenbanken können als Anfrageregion(en) im k-dimensionalen Raum, der durch die k Attribute einer Relation aufgespannt wird, verstanden werden. Alle Datensätze, die innerhalb der Anfrageregion(en) liegen, stellen die Antwort auf die Anfrage dar und werden als

Ergebnis der Anfrage ermittelt. Die Menge der Antworten kann sehr groß, sie kann aber auch leer sein. In beiden Fällen ist es für den Benutzer schwierig, die Antwort zu verstehen und die Anfrage entsprechend zu modifizieren. Um dem Benutzer mehr Feedback auf seine Anfrage zu geben, werden durch unsere Visualisierungstechniken nicht nur die Datensätze visualisiert, die innerhalb der Anfrageregion(en) liegen und damit die Anfrage erfüllen, sondern auch solche, die 'in der Nähe' der Anfrageregion(en) liegen und damit die Anfrage nur approximativ erfüllen.

Unabhängig davon, ob ein Datensatz die Anfrage erfüllt oder nicht, kann für jedes Attribut der Abstand von dem vorgegebenen Anfragewert (oder -intervall) berechnet werden. Macht man dies für jedes Attribut, so erhält man Tupel $(d_1, d_2,..., d_k)$, die die Distanzen der Datenwerte bezüglich der Anfrage beinhalten. Verändert man die Anfrageregion, so ändern sich die Distanztupel entsprechend. Das Distanztupel kann um einen (k+1)-ten Werte erweitert werden, der die Gesamtdistanz des Datensatzes bezüglich der Anfrage darstellt. Der Wert von d_{k+1} ist '0', falls der Datensatz die Anfrage erfüllt; ansonsten gibt d_{k+1} den Abstand des Datensatzes bezüglich der Anfrage wieder. Die Menge der Distanztupel $(d_1, d_2,..., d_k, d_{k+1})$ wird nach dem Wert d_{k+1} (Resultat) aufsteigend sortiert, d.h. am Anfang stehen die Tupeln mit $d_{k+1} = 0$ (falls vorhanden) und am Schluß die Tupel mit den größten Distanzen.

Als nächstes werden den Distanzwerten Farben zugeordnet. Die Abbildung des Werteintervalls für jedes Attribut inklusive des Gesamtresultats wird dabei auf eine spezielle Farbskala abgebildet. Die Farbskala ist so entworfen, daß dem Distanzwert '0' die Farbe gelb zugeordnet ist; Distanzwerte größer '0' werden in aufsteigender Reihenfolge immer dunkler. Das Farbspektrum durchläuft die Farben hellgrün, blau, rot bis dunkelbraun. Die gelbe Farbe ist besonders hervorgehoben und zeigt an, daß der zugehörige Datenwert innerhalb des vorgegebenen Anfrageintervalls liegt; die übrigen Farben zeigen die relative Entfernung des Attributwertes von dem Intervall an. Für eine einfache Zuordnung von Datenwerten zu den Farbpixeln sorgt eine Option des interaktiven Interfaces: Durch Anklicken von Pixeln können die zugehörigen Datenwerte abgefragt werden. Details des interaktiven Interfaces sind in [KKS 94] und [Kei 94] beschrieben.

Die Verfahren zur Berechnung der Distanzen und ihre Kombination in die Gesamtdistanz sowie die Behandlung komplexer Anfragen, die aus einer beliebigen Boole'schen Verknüpfung von Anfragebedingungen (geschachtelte 'AND's und 'OR's) bestehen, mehrere Relationen betreffen oder aus einer Schachtelung von Teilanfragen bestehen, wurden in [KKS 94] vorgestellt und sollen deshalb an dieser Stelle nicht näher erläutert werden.

3.1 Spiralanordnung

Bei der Spiralanordnung wird jeder Distanzwert durch ein Pixel repräsentiert. Die Distanzwerte für die einzelnen Attribute sowie das Gesamtergebnis werden in separaten Fenstern dargestellt (vgl. Abb. 7). Die Anordnung der Pixel geschieht spiralförmig um die Mitte der Fenster herum (vgl. Abb. 5). Die Reihenfolge der Pixel entspricht dabei der Sortierung entsprechend der Gesamtdistanz. Im Fenster für das Gesamtergebnis sind in der Mitte die gelben Pixel; weiter nach außen verlaufen die Farben kontinuierlich von hellgrün bis dunkelbraun. Die Fenster für die einzelnen Attribute weisen keine kontinuierlichen Farbübergänge auf, da die Pixel der Attribute in derselben Reihenfolge angeordnet sind wie im Fenster für das Gesamtergebnis. Die Farben der Pixel sind von den Attributwerten abhängig und daher nicht gleichmäßig verteilt. Die Visualisierung der Datenbank besteht damit aus insgesamt k+1 Fenstern der gleichen Größe, wobei jedes Fenster eine Dimension des R^k (bzw. R^{k+1}) repräsentiert. Die Pixel, die zu den Attributwerten eines Datensatzes gehören, liegen in verschiedenen Fenstern. Da sie jedoch in jedem Fenster die

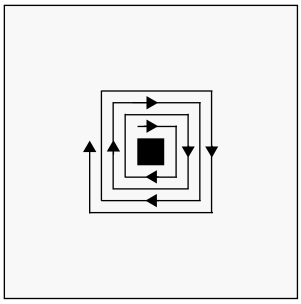


Abb. 5: Spiralanordnung eines Attributs

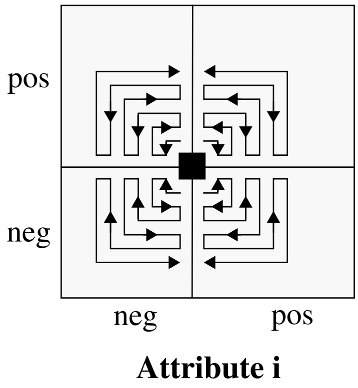


Abb. 6: Achsenanordnung eines Attributs

gleichen Koordinaten haben, können Zusammenhänge zwischen den Attributwerten eines Datensatzes hergestellt werden (vgl. Abb. 7).

3.2 Achsenanordnung

Die Achsenanordnung ist eine Weiterentwicklung der Spiralanordnung und beruht auf der Idee, die Daten entsprechend ihrer Distanz bezüglich zweier ausgewählter Attribute auf dem Bildschirm anzuordnen. Da Attributwerte im allgemeinen kleiner oder größer als das vorgegebene Anfrageintervall sein können, ergeben sich bei der Berechnung der Distanzen positive und negative Werte. Diese zusätzliche Information wird bei der Achsenanordnung ausgenutzt, um die Daten in vier Quadranten anzuordnen. Dazu werden zwei Attribute ausgewählt, und die Teilbilder für die Dimensionen bzw. die Gesamtdistanz durch zwei orthogonale Achsen in jeweils vier Quadranten aufgeteilt. Datensätze mit positiven Distanzen werden rechts bzw. oberhalb, Datensätze mit negativen Distanzen links bzw. unterhalb der jeweiligen Achse gezeichnet. Die Anordnung der Datensätze innerhalb der Quadranten erfolgt rechtwinklig um das Zentrum herum (vgl. Abb. 6). Die Datensätze, die die Anfrage erfüllen, liegen wie bei der Spiralanordnung im Zentrum der Achsen. Da die Quadranten bei der Achsenanordnung im allgemeinen nicht gleichmäßig gefüllt sind, können weniger Datensätze als bei der Spiralanordnung dargestellt werden. Dies ist aber der Preis für die größere Aussagekraft der entstandenen Visualisierungen, die in einigen Fällen Eigenschaften der Daten besser hervorheben (siehe Abschnitt 4).

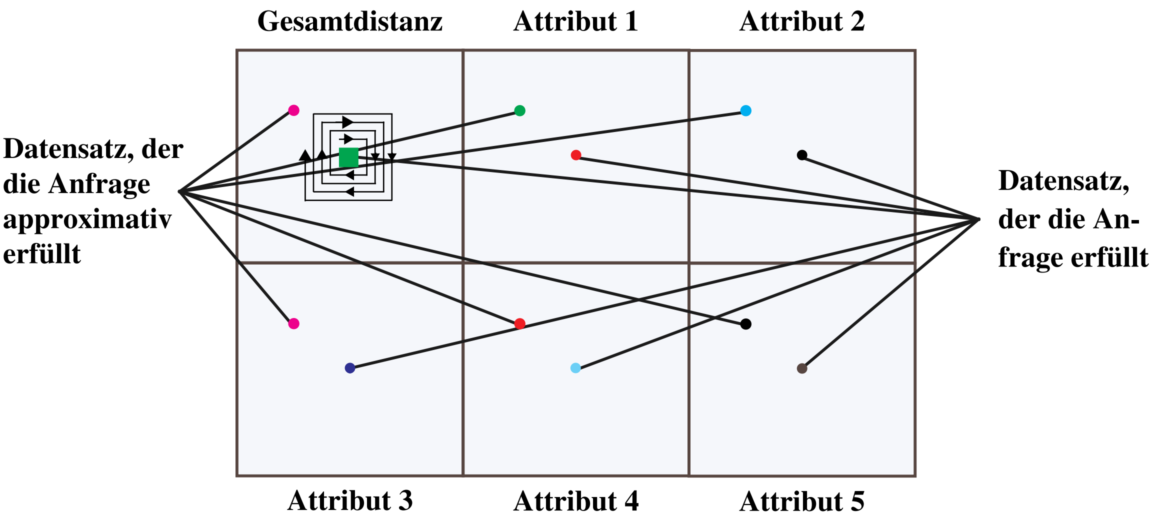


Abb. 7: Anordnung der Fenster bei der Visualisierung fünfdimensionaler Daten

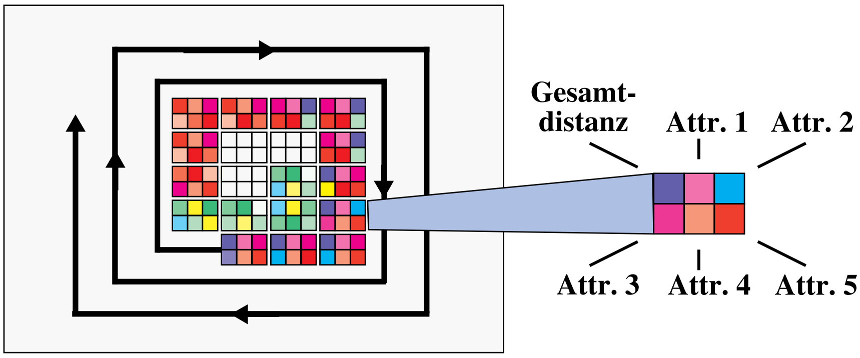


Abb. 8: Gruppenanordnung fünfdimensionaler Daten

3.3 Gruppenanordnung

Bei der Spiral- und Achsenanordnung sind die Pixel, die die Distanzen eines Datensatzes bezüglich seiner Attribute darstellen, in mehrere Teilbilder aufgeteilt. Im Gegensatz dazu werden bei der Gruppenanordnung die zu einem Datensatz gehörenden Distanzen in einem zusammenhängenden Bereich dargestellt (vgl. Shape Coding Technique in Abschnitt 2). Die Bereiche sind wie bei der Spiralanordnung spiralförmig um die Mitte des Fensters angeordnet. Im Gegensatz zur Spiralanordnung besteht die Visualisierung aus nur einem Fenster, das die Distanzwerte sämtlicher Datensätze darstellt, d.h. die Visualisierung ist nicht in Teilbilder für die Attribute und die Gesamtdistanz unterteilt. Die Gruppenanordnung benötigt wesentlich mehr Platz auf dem Bildschirm, da zum einen die Attributwerte nicht mehr mit nur einem Pixel dargestellt werden können (für die Erkennbarkeit einzelner Werte sind mindestens 2 x 2 Pixel erforderlich), zum anderen wird zusätzlich Platz für die Zwischenräume benötigt, damit die Datensätze voneinander unterscheidbar sind. Insgesamt können deshalb deutlich weniger Datensätze auf dem Bildschirm dargestellt werden (siehe Abschnitt 4).

3.4 Sequenzen von Visualisierungen

Bei den bisher vorgestellten Visualisierungstechniken ist die Anzahl der Datenwerte, die visualisiert werden können, durch die Anzahl der Pixel des Bildschirms beschränkt und liegt damit höchstens in der Größenordnung von einer Million Pixel; für die uns zur Verfügung stehenden 19 Zoll Bildschirme mit einer Auflösung von 1.024 x 1.280 sind es 1.3 Millionen Pixel. Um noch größere Datenmengen am Bildschirm darstellen zu können, müssen weitere Darstellungsdimensionen hinzugenommen werden. Naheliegend ist die Verwendung der Dimension ‘Zeit’. Die prinzipielle Idee dabei ist, durch Verschieben der Anfrageregion im k-dimensionalen Raum Sequenzen von Visualisierungen zu erzeugen. Dadurch können zum einen größere Datenmengen visualisiert werden, zum anderen werden durch die Veränderung der Bilder aber auch Abhängigkeiten innerhalb der Daten besser wahrnehmbar. Bewegt man die Anfrageregion beispielsweise entlang einer Dimension des k-dimensionalen Raumes, so erhält man einen Überblick über die entsprechenden Veränderungen anderer Dimensionen. Die Verschiebung der Anfrageregion ist aber nicht auf einzelne Dimensionen beschränkt, sondern kann entlang eines beliebigen Pfades im k-dimensionalen Raum erfolgen. Der Bereich des k-dimensionalen Raumes, der durch die Visualisierungssequenz dargestellt wird, ist aber nicht nur vom Pfad sondern auch von der Verteilung der Daten im k-dimensionalen Raum abhängig. Die Ermittlung von Pfaden, die eine vollständige Überdeckung des gesamten k-dimensionalen Raumes garantieren, ist deshalb im allgemeinen ein schwieriges und bisher noch ungelöstes Problem.

4. Bewertung und Vergleich unserer Visualisierungstechniken

Ein zentrales Anliegen dieses Papers sind Bewertung und Vergleich unserer Visualisierungstechniken. Bei der Bewertung von Visualisierungstechniken stehen nicht, wie sonst bei Leistungsvergleichen, die CPU-Zeiten oder die Anzahl der Zugriffe auf den Sekundärspeicher im Vordergrund, sondern die Wahrnehmbarkeit von Zusammenhängen und Eigenschaften der Daten. Für die Bewertung und den Vergleich wurden sowohl reale als auch künstlich erzeugte Daten verwendet. Die Realdaten stammen aus einem Forschungsprojekt im Bereich der Molekularbiologie [EKSX 95]. Ziel des Projektes ist es, ein effizientes Durchsuchen von Proteindatenbanken nach potentiellen Dockingkandidaten und geeigneten Dockingstellen zu unterstützen. Um einen effizienten Zugriff auf potentielle Dockingkandidaten und -stellen zu gewährleisten, wurde in dem Projekt ein sogenannter 'Feature-Index' entwickelt, mit dessen Hilfe alle Proteine bzw. Regionen ermittelt werden können, deren Parameter zu den Oberflächenparametern des gegebenen Proteins komplementär sind [Ald 94]. Der Feature Index basiert auf einer Partitionierung der Proteinoberflächen in Regionen, die entsprechend eines oder mehrerer Oberflächenparameter vorgenommen wird. Bei der Partitionierung ist es wichtig, die richtigen Wertebereiche für die beteiligten Parameter sowie eine adäquate Gewichtung der Parameter bei der Verknüpfung zu finden. Für diese Teilaufgabe wurde unser Anfrage- und Visualisierungssystem erfolgreich eingesetzt, da es eine interaktive Veränderung der Anfragebereiche erlaubt und Feedback über die Verteilung der Daten bezüglich der Attribute liefert. Details sind in [Kei 94] beschrieben.

Ähnlich wie bei Untersuchungen der CPU-Zeit bzw. der Anzahl der Zugriffe auf den Hintergrundspeicher eignen sich reale Daten wegen ihrer schwer charakterisierbaren Eigenschaften jedoch nur bedingt zur Evaluierung von Visualisierungstechniken. Analog zu den, für Leistungsvergleiche von Datenbanksystemen entwickelten Benchmarks, haben wir deshalb für die Evaluierung von Visualisierungssystemen Teststrategien entwickelt, die künstlich erzeugte Daten mit vorgegebenen Eigenschaften verwenden und damit die Reproduzierbarkeit und Vergleichbarkeit der Ergebnisse gewährleisten. In folgenden werden die Teststrategien, das Testdaten-Generierungssystem sowie die Ergebnisse einiger Tests vorgestellt.

Für die Generierung künstlicher Daten wird die Testdatengenerierungsumgebung *'TestVis'* verwendet, die die Spezifikation eines breiten Spektrums von Datencharakteristika erlaubt. *TestVis* basiert auf einem Testdaten-Modell, das die Spezifikation verschiedener, für Visualisierungszwecke relevanter Datenmengen (z.B. statistische Daten oder Bilddaten) erlaubt [BKP 94].

Die in Datenbanken gespeicherten Daten können im allgemeinen am besten mit Hilfe statistischer Methoden beschrieben werden. *TestVis* erlaubt zu diesem Zweck die Spezifikation von Daten mit einer beliebigen Anzahl von Attributen (Dimensionen) und einer beliebigen Anzahl von Clustern. Für jedes Attribut kann die zugehörige Verteilungsfunktionen bzw. eine funktionale Abhängigkeit[1] von anderen Attributen festgelegt werden. Gleiches gilt entsprechend für die Cluster, wobei zusätzlich die Größe und Lage der Cluster festgelegt wird. Die funktionalen Abhängigkeiten, die mit Hilfe von *TestVis* spezifiziert werden können, sind von der Form

$$(1 + r \times rf) \times \left(\sum_{i=1}^{k} c_{i1} \times a_i^{c_{i2}} \right),$$

1. Mit funktionaler Abhängigkeit zwischen Attributen sind nicht nur Attributgleichheit sondern der allgemeinere Fall beliebiger mathematischer Zusammenhänge zwischen Attributen gemeint.

wobei

- c_{i1} und c_{i2} benutzer-definierte Konstanten sind,
- a_i die Attribute sind, von denen das spezifizierte Attribut funktional abhängig ist,
- r eine Zufallszahl im Bereich [-1, 1] und
- rf eine benutzer-definierte Zahl ist, die eine gewisse Zufälligkeit der Daten induziert.

Eine genauere Beschreibung des *TestVis*-Systems ist in [Kei 94] zu finden.

Im folgenden sollen Beispiele für generierte Testdaten mit spezifischen Eigenschaften gegeben sowie die zugehörigen Visualisierungen beschrieben werden. Die verwendeten Testdaten zeichnen sich durch einen großen, zufällig erzeugten Grunddatenbestand aus, der mindestens zwei Drittel der gesamten Daten ausmacht. In diesen Grunddatenbestand sind dann ein oder mehrere Cluster eingefügt, die eine unterschiedliche Dimensionalität haben bzw. unter Verwendung verschiedener Verteilungsfunktionen und funktionaler Abhängigkeiten definiert wurden. Aufgrund der hohen Anzahl an zufällig erzeugten Datensätzen führen mathematische und statistische Methoden für die betrachteten Testdaten nicht zu befriedigenden Ergebnissen. Aus Platzgründen verwenden wir für die Auswertung als Visualisierungstechnik nur die Spiralanordnung. Ein Vergleich von Spiral-, Achsen- und Gruppenanordnung wird anschließend vorgenommen.

4.1 Cluster mit unterschiedlicher Dimensionalität

In Abb. 9 sind Visualisierungen von sechsdimensionalen Testdaten dargestellt, die vier- bzw. fünfdimensionale Cluster enthalten. Die verwendeten Testdaten bestehen aus 15.000 Datensätzen, von denen zwei Drittel zufällig (im Bereich [0,100] für jede Dimension) generiert wurden. Das verbleibende Drittel besteht aus drei Clustern, die dadurch definiert sind, daß in vorgegebenen Wertebereichen für die Clusterdimensionen zusätzliche Datensätze generiert wurden. Abb. 9a zeigt die Visualisierung der Daten mit vierdimensionalen Clustern, Abb. 9b die Daten mit fünfdimensionalen Clustern. Als Anfrageregion wurde in beiden Fällen der Bereich [0, 10] für jede Dimension gewählt. Vergleicht man Abb. 9a und b, so fällt auf, daß die Wahrnehmbarkeit der Cluster mit kleiner werdender Dimension der Cluster deutlich abnimmt. Cluster mit kleiner Dimension können jedoch wahrnehmbar gemacht werden, indem der Gewichtungsfaktor eines Attributs auf einen Wert gesetzt wird, der deutlich höher ist als die Gewichtungsfaktoren der übrigen Attribute. Diese Technik führt insbesondere zu guten Ergebnissen, wenn das gewählte Attribut auch Clusterattribut ist. In Abb. 10a sind die Testdaten mit vierdimensionalen Clustern abgebildet, wobei Attribut zwei eine deutlich höhere Gewichtung als die übrigen Attribute hat. Sind die Gewichtungsfaktoren aller Clusterattribute höher als die der Nicht-Clusterattribute, so werden die Clusterregionen in der Visualisierung deutlicher und die Strukturen in den Fenstern für die Nicht-Clusterattribute verschwinden (c.f Abb. 10b).

Bei Experimenten mit ähnlichen Testdaten stellte sich heraus, daß die Ausdehnung der Cluster im multidimensionalen Raum nur einen kleinen Effekt auf die Visualisierung hat. Wichtiger hingegen ist die Anzahl der Datensätze, die zum Cluster gehören. Cluster, die nur aus wenigen Datensätzen bestehen, können nur wahrgenommen werden, wenn sie nahe an der Anfrageregion liegen und sich deutlich von den übrigen Datensätzen unterscheiden. Der Prozentsatz an Datensätzen, der zu einem Cluster gehören muß, damit es wahrnehmbar ist, hängt von der Unterschiedlichkeit zwischen Cluster und übrigen Daten, von der Dimension des Clusters im Vergleich zur Dimension der Daten, und von der Entfernung des Clusters von der Anfrageregion ab.

4.2 Cluster mit verschiedenen Datenverteilungen

In Abb. 11 sind Visualisierungen von Testdaten abgebildet, bei denen verschiedene Verteilungsfunktionen für die Clusterdefinition verwendet wurden. Der Grunddatenbestand von

10.000 Datensätzen ist im Bereich [-10000, 10000] gleichverteilt für jede Dimension. Das Cluster besteht aus 1.000 Datensätzen, wobei die Clusterdimensionen sich in der verwendeten Verteilungsfunktion sowie ihren Parametern unterscheiden. Die Parameter der Verteilungsfunktionen für die Dimensionen sind in folgender Tabelle zusammengefaßt:

	gleichverteilt			normalverteilt					
Dimension	1	2	3	4	5	6	7	8	9
Untere Grenze bzw. Mittelwert	-100	-1000	-10000	0	0	0	100	1000	5000
Obere Grenze bzw. Standardabweichung	100	1000	10000	10	100	1000	1000	1000	5000

Als Anfrageregion wurde der Bereich [0, 10] für jede Dimension benutzt. Da sich Anfrageregion und Cluster in den Dimensionen eins, zwei, vier und fünf überschneiden, ist das Cluster im Zentrum der entsprechenden Fenster gut erkennbar (vgl. Abb. 11a). Wird eine andere Anfrageregion verwendet, so ist es deutlich schwieriger das Cluster zu erkennen (vgl. Abb. 11b). Durch die Visualisierungen wird deutlich, daß für die Dimensionen mit einem großen Bereich bei der Gleichverteilung (Dimension drei) bzw. einer hohen Standardabweichung bei der Normalverteilung (Dimensionen sechs und neun) keine Clusterung vorhanden ist. Durch eine höhere Gewichtung einer Dimension wird dieser Effekt noch klarer (vgl. Abb. 11c).

4.3 Cluster mit funktionalen Abhängigkeiten

In Abb. 12 sind Visualisierungen von Testdaten zu sehen, bei denen verschiedene funktionale Abhängigkeiten für die Clusterdefinition verwendet wurden. Der Grunddatenbestand von 10.000 Datensätzen ist wieder gleichverteilt, diesmal allerdings in verschiedenen Bereichen: Für Dimension eins bis drei im Bereich [0, 1000], für Dimension vier bis sechs im Bereich [0, 2000], und für Dimension sieben bis neuen im Bereich [0, 1000000]. Das Cluster besteht aus 2.000 Datensätzen. Dimensionen eins bis drei des Clusters sind die unabhängigen Dimensionen, die im Bereich [0, 1000] gleichverteilt sind. Die funktionalen Abhängigkeiten der Clusterdimensionen sind in folgender Tabelle zusammengefaßt:

Dimension	1	2	3	4	5	6	7	8	9
funktionale Abhängigkeit	gleichverteilt im Bereich [0, 1000]			linear abhängig von Dimension			quadratisch abhängig von Dimension		
				1	2, 3	1, 2, 3	1	2, 3	1, 2, 3

Als Anfrageregion wurde der Ursprung des Koordinatensystems (Bereich [0, 0] für jede Dimension) benutzt. In Abb. 12a - c sind drei Visualisierungen der Daten abgebildet. Die drei Visualisierungen der Daten unterscheiden sich in den verwendeten Gewichtungsfaktoren. In Abb. 12a wird ein höheres Gewicht für Dimension eins verwendet, wodurch eine Strukturierung in den abhängigen Dimensionen vier und sieben sichtbar wird. In den Fenstern für die nicht von Dimension eins abhängigen Dimensionen fünf und acht dagegen ist keinerlei Strukturierung zu erkennen. Die Fenster für Dimensionen sechs und neun, die teilweise von Dimension eins abhängig sind, zeigen nur eine relativ schwach wahrnehmbare Strukturierung. In Abb. 12b wird ein höheres Gewicht für Dimension zwei und drei verwendet. Die entsprechenden abhängigen Dimensionen fünf und acht bzw. sechs und neun des Clusters sind zwar erkennbar, jedoch nicht so deutlich wie erwartet. Wird ein höheres Gewicht für die Dimensionen eins bis drei verwendet (vgl. Abb. 12c), so nimmt die Wahrnehmbarkeit der Abhängigkeiten weiter ab.

Um funktionale Abhängigkeiten sichtbar zu machen, entwickelten wir eine Technik, die wir 'Farbinvertierung' nennen [Kei 94]. Farbinvertierung bedeutet dabei, daß die Farbtabelle und damit die Zuordnung zwischen Farben und Distanzwerten invertiert wird. Die Auswirkungen der Farbinvertierungstechnik auf die Visualisierungen von Abb. 12 sind in Abb. 13 dargestellt.[1] Besonders auffällig ist die bessere Erkennbarkeit von Clusterdimensionen, die von mehreren Dimensionen abhängig sind.

4.4 Vergleich unserer Visualisierungstechniken

Für den Vergleich von Spiral- und Achsenanordnung verwenden wir vierdimensionale Testdaten mit mehreren Clustern, die sich an verschiedenen Stellen des k-dimensionalen Raumes befinden. Ein Ergebnis unserer Untersuchungen ist, daß die Cluster im allgemeinen bei der Achsenanordnung besser erkennbar sind als bei der Spiralanordnung. In Abb. 14 beispielsweise ist bei der Spiralanordnung keinerlei Strukturierung erkennbar, wohingegen die Achsenanordnung die Cluster recht deutlich zeigt. Die Wahl der Zuordnung von Attributen zu den Achsen hat bei der Achsenanordnung jedoch einen entscheidenden Einfluß auf die Erkennbarkeit der Cluster. Je nachdem, welche der Attribute den Achsen zugeordnet sind, sind die Cluster mehr oder weniger gut zu erkennen. Die Visualisierungen in Abb. 14b und Abb. 15 unterscheiden sich nur in der Zuordnung von Attributen zu den Achsen. Die Achsenanordnung bietet im allgemeinen mehr Information als die Spiralanordnung (vgl. Abb. 14), jedoch müssen zunächst geeignete Attribute für die Zuordnung zu den Achsen gefunden werden. Dies ist insbesondere bei Daten mit einer hohen Dimensionalität nicht immer einfach. Ein Nachteil der Achsenanordnung ist, daß die Anzahl der Datensätze, die visualisiert werden können im allgemeinen geringer ist. Im Extremfall sind zwei gegenüberliegende Quadranten leer (vgl. Abb. 16b), was bedeutet, daß nur halb so viel Datensätze wie bei der Spiralanordnung dargestellt werden können (Achsenanordnung in Abb. 16 ist verkleinert).

In Abb. 17 und Abb. 18 vergleichen wir alle drei Visualisierungstechniken. Zu diesem Zweck benutzen wir zwei achtdimensionale Testdatenmengen, die gleiche Clustereigenschaften haben, aber aus einer unterschiedlichen Anzahl an Datensätzen bestehen. Die in Abb. 17 visualisierten Testdaten bestehen aus 1.000 Datensätzen, die in Abb. 18 visualisierten aus 7.000 Datensätzen. Die Spiral- und Achsenanordnung in Abb. 17 und Abb. 18 sind vergrößert, wohingegen die Gruppenanordnung verkleinert ist (die Gruppenanordnung in Abb. 18 ist auf ca. 3% ihrer Originalgröße verkleinert). Durch die Größe der Visualisierungen wird bereits deutlich, daß die Gruppenanordnung nur für kleinere Datenmengen geeignet ist. Ein Vorteil der Gruppenanordnung ist jedoch, daß sie auch für Daten sehr hoher Dimension brauchbare Visualisierungen liefert. Bei der Spiral- und Achsenanordnung stehen die Pixel für die einzelnen Dimensionen nur durch ihre Position miteinander in Beziehung. Bei einer geringen Anzahl an Dimensionen ist es für den Menschen relativ leicht, die Fenster miteinander in Beziehung zu setzen. Je größer die Anzahl der Dimensionen jedoch wird, desto schwieriger wird es, die große Anzahl an Fenstern zu überblicken und Korrelationen zwischen den Fenstern zu erkennen. Bei der Gruppenanord-

1. Die Qualität der gedruckten Version unserer Visualisierungen ist relativ schlecht im Vergleich zur Qualität der Visualisierungen auf dem Bildschirm. Eigenschaften der Daten, die auf dem Bildschirm leicht zu erkennen sind, sind deshalb in der gedruckten Version zum Teil relativ schlecht erkennbar.

nung ist dies nicht notwendig und deshalb ist sie insbesondere für Daten höherer Dimension geeignet. Das blaue Cluster beispielsweise, das in der Gruppenanordnung von Abb. 17 deutlich erkennbar ist, ist in der Spiral- und Achsenanordnung nur relativ schlecht auszumachen. Eine weitere Beobachtung beim Vergleich der Visualisierungen von Abb. 18 ist, daß die Cluster im allgemeinen am deutlichsten in der Achsenanordnung zu erkennen sind. In vielen Fällen liegen die Cluster vollständig in einem der Quadranten (zum Beispiel das braune Cluster im Fenster für Dimension 8 in Abb. 18), was zusätzlich Rückschlüsse auf die Lage des Clusters bezüglich der den Achsen zugeordneten Attribute erlaubt.

Bei der Exploration unbekannter Daten ist es sinnvoll, zunächst mit der Spiralanordnung zu beginnen, um einen Überblick über die Daten zu bekommen. Hat man einen ersten Eindruck und insbesondere eine Idee, welche Dimensionen den Achsen zugeordnet werden sollen, so kann man mit Hilfe der Achsenanordnung eine detailliertere Analyse vornehmen. Die Gruppenanordnung kann dann in einem späteren Schritt für eine gezielte Analyse kleinerer Datenmengen verwendet werden. Datenexploration mit Hilfe unseres Anfrage- und Visualisierungssystems ist ein interaktiver Vorgang, der mit Hilfe von Abbildungen nur zum Teil beschrieben werden kann. Unser Datenbankvisualisierungssystem bietet zahlreiche Optionen, die die Interaktivität unterstützen. Beispiele sind die Optionen, die Datenwerte für ausgewählte Pixel oder Farbbereiche liefern, und die Slider zur Reduktion der Datenmenge. Details sind in [Kei 94], zum Teil auch in [KKS 94] zu finden.

Um die vorgestellten Datenexplorations- und Visualisierungstechniken möglichst effektiv nutzen zu können, ist es notwendig, eine globale Strategie zur Datenexploration zu haben. Einige aus der intensiven Benutzung des Systems stammenden Erfahrungen sollen im folgenden kurz beschrieben werden: Ist nichts über die Daten bekannt, so empfiehlt es sich, mit dem Ursprung des Koordinatensystems als Anfrageregion zu beginnen, wobei alle Dimensionen das gleiche Gewicht haben. Der folgende Datenexplorationsprozeß ist weitgehend durch das visuelle Feedback bestimmt, das der Benutzer durch die Visualisierungen erhält. Erhält der Benutzer nämlich Hinweise auf Korrelationen, funktionale Abhängigkeiten oder sonstige Cluster, so wird er versuchen, mit Hilfe des Systems diese Hypothesen zu überprüfen. Für diesen Zweck kann der Benutzer beispielsweise die Anfrageregion oder die Gewichtung der Dimensionen verändern. Falls in den Visualisierungen jedoch keine Hinweise auf interessante Eigenschaften der Daten erkennbar sind, so kann er einer beliebigen Dimension ein höheres Gewicht geben, die Farbinvertierungstechnik verwenden, den Prozentsatz der angezeigten Datensätze verändern oder eine andere Anfrageregion verwenden. In dieser Aufzählung wurden die Möglichkeiten, die unserer Erfahrung nach das beste Aufwand-Nutzen-Verhältnis haben, zuerst aufgeführt.

5. Zusammenfassung und Ausblick

Visualisierungstechniken können bei der Exploration und Analyse sehr großer multidimensionaler Daten hilfreich sein, um interessante Daten und ihre Eigenschaften zu finden. Unser Ansatz der Datenexploration zielt auf eine adäquate Unterstützung des Menschen durch den Computer ab und kombiniert Datenbankanfrage- und Information Retrieval-Techniken mit neuartigen Visualisierungstechniken. Die Anzahl der Datenwerte, die zu einem Zeitpunkt am Bildschirm dargestellt werden können, ist dabei nur durch die Auflösung des Bildschirms beschränkt. Verschiedene Varianten unserer Visualisierungstechniken unterstützen den Benutzer in den verschiedenen Phasen des Datenexplorations-Prozesses. Für Vergleich und Bewertung unserer Techniken werden künstlich erzeugte Testdaten verwendet, die entsprechend eines systematischen Testdaten-Modells generiert wurden.

Für die Entwicklung verbesserter Visualisierungstechniken ist eine systematische Analyse der existierenden Techniken im Hinblick auf ihre Möglichkeiten und Grenzen sowie ein detaillierter Vergleich der Techniken notwendig. Insbesondere ist ein Vergleich unserer Techniken mit anderen Visualisierungstechniken, die sich für größere Datenmengen eignen (z.B. die 'parallel coordinates'- und 'stick figure'-Technik), notwendig. Es ist beispielsweise zu untersuchen, welche Arten von Korrelationen, Clustern, funktionellen Abhängigkeiten, usw. in den durch die verschiedenen Techniken erzeugten Visualisierungen 'erkennbar' sind. Dabei spielen die betrachteten (realen oder künstlich erzeugten) Testdaten ebenso eine Rolle wie psychologische Fragen der Wahrnehmung (perception).

Ein weiterer Untersuchungsgegenstand ist die effiziente Unterstützung unseres Visualisierungssystems durch die unterliegenden Datenbanksysteme. Wie erste Untersuchungen zeigen [Kei 94], eignen sich heute kommerziell verfügbare Datenbanksysteme nur sehr beschränkt, Visualisierungstechniken wie die oben beschriebenen zu unterstützen. Datenbanksysteme unterstützen zwar hohe Transaktionsraten und die schnelle Suche nach exakt spezifizierten Daten, bieten aber nur unzureichende Performanz für Bereichsanfragen, die sich auf mehrere Attribute beziehen. Solche Anfragen erfordern eine schnelle multidimensionale Suche, die im allgemeinen nur unzureichend unterstützt wird. Ein weiteres Problem ist, daß in heutigen Datenbanksystemen jede Anfrage separat bearbeitet wird. Es gibt keine Möglichkeit, für eine Folge von Anfragen, die sich nur wenig von einander unterscheiden, nur den sich ändernden Teil der Antwortmenge zu erzeugen. Dies wirkt sich besonders negativ auf die Interaktivität unseres Visualisierungssystems aus. Zusätzlich ist es für unsere Visualisierungstechniken erforderlich, auf sämtlich Teilergebnisse für jedes Attribut bzw. für jede Teilanfrage zugreifen zu können. Auch dies wird von heutigen Datenbanksystemen nicht oder nur unzureichend unterstützt. Zusammenfassend kann man sagen, daß bei der Entwicklung einer sekundärspeicherbasierten Version des Systems, basierend auf einem kommerziell verfügbaren Datenbanksystem, noch eine Reihe interessanter Probleme gelöst werden müssen.

Referenzen

[ABN 92] Anwar T. M., Beck H. W., Navathe S. B.: *'Knowledge Mining by Imprecise Querying: A Classification-Based Approach',* Proc. 8th Int. Conf. on Data Engineering, Tempe, AZ, 1992, pp. 622-630.

[Ald 94] Aldinger K., Ester M., Förstner G., Kriegel H.-P., Seidl T.: *'Datenbankunterstützung für das Protein-Protein-Docking: Ein effizienter und robuster Feature-Index',* Proc. 2nd GI-Fachtagung 'Informatik in den Biowissenschaften', Jena, Germany, 1994.

[And 57] Anderson E.: *'A Semigraphical Method For The Analysis of Complex Problems',* Proc. Nat. Acad. Sci. USA, Vol. 13, 1957, pp. 923-927.

[And 72] Andrews D. F.: *'Plots of High-Dimensional Data',* Biometrics, Vol. 29, 1972, pp. 125-136.

[Asi 85] Asimov D.: *'The Grand Tour: A Tool For Viewing Multidimensional Data',* SIAM Journal of Science & Stat. Comp., Vol. 6, 1985, pp. 128-143.

[Bed 90] Beddow J.: *'Shape Coding of Multidimensional Data on a Mircocomputer Display',* Visualization '90, San Francisco, CA, 1990, pp. 238-246.

[BF 90] Beshers C., Feiner S.: *'Visualizing n-Dimensional Virtual Worlds with n-Vision'*, Computer Graphics, Vol. 24, No. 2, 1990, pp. 37-38.

[BKP 94] Bergeron R. D., Keim D. A., Pickett R.: *'Test Data Sets for Evaluating Data Visualization Techniques',* in: Perceptual Issues in Visualization, Springer, 1994.

[Bri 79] Brissom D.: *'Hypergraphics: Visualizing Complex Relationships in Art, Science and Technology',* Amer. Association for the Advance of Science, Westview Press, Boulder, 1979.

[Che 73] Chernoff H.: *'The Use of Faces to Represent Points in k-Dimensional Space Graphically',* Journal Amer. Statistical Association, Vol. 68, pp 361-368.

[Cle 93] Cleveland W. S.: *'Visualizing Data',* AT&T Bell Laboratories, Murray Hill, NJ, Hobart Press, Summit NJ, 1993.

[DE 82] Dunn G., Everitt B.: *'An Introduction to Mathematical Taxonomy',* Cambridge University Press, Cambridge, MA, 1982.

[EKSX 95] Ester M., Kriegel H.-P., Seidel T., Xu X.W.: *'Formbasierte Suche nach komplementären 3D-Oberflächen in einer Protein-Datenbank',* Proc. GI-Fachtagung 'Datenbanken in Büro, Technik und Wissenschaft' (BTW), Dresden, Germany, 1995.

[FM 91] Frei H. P., Meienberg S.: *'Evaluating Weighted Search Terms as Boolean Queries',* Proc. GI/GMD-Workshop, Darmstadt 1991, in: Informatik-Fachberichte, Vol. 289, 1991, pp. 11-22.

[FPM 91] Frawley W. J., Piatetsky-Shapiro G., Matheus C. J.: *'Knowledge Discovery in Databases: An Overview',* in: Knowledge Discovery in Databases, AAAI Press, Menlo Park, 1991.

[FT 74] Friedman J., Tukey J.: *'A Projection Pursuit Algorithm for Exploratory Data Analysis',* IEEE Transactions on Computers, Vol. 23, 1974, pp. 881-890.

[GGM 92] Gaasterland T., Godfrey P., Minker J.: *'An Overview of Cooperative Answering',* Journal of Intelligent Information Systems, Vol. 1, 1992, pp. 123-157.

[Hub 85] Huber P. J.: *'Projection Pursuit',* The Annals of Statistics, Vol. 13, No. 2, 1985, pp. 435-474.

[ID 90] Inselberg A., Dimsdale B.: *'Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry'*, Visualization '90, San Francisco, CA, 1990, pp. 361-370.

[Ins 85] Inselberg A.: *'The Plane with Parallel Coordinates, Special Issue on Computational Geometry',* The Visual Computer, Vol. 1, 1985, pp. 69-97.

[Kei 94] Keim D. A.: *'Visual Support for Query Specification and Data Mining',* Dissertation, Universität München, 1994.

[KKS 94] Keim D. A., Kriegel H.-P., Seidl T.: *'Supporting Data Mining of Large Databases by Visual Feedback Queries',* Proc. 10th Int. Conf. on Data Engineering, Houston, TX, 1994, pp. 302-313.

[LWW 90] LeBlanc J., Ward M. O., Wittels N.: *'Exploring N-Dimensional Databases',* Visualization '90, San Francisco, CA, 1990, pp. 230-239.

[MGTS 90] Mihalisin T., Gawlinski E., Timlin J., Schwendler J.: *'Visualizing A Scalar Field on an N-dimensional Lattice',* Visualization '90, San Francisco, CA, 1990, pp. 255-262.

[Mot 90] Motro A.: *'FLEX: A Tolerant and Cooperative User Interface to Databases',* IEEE Transactions on Knowledge and Data Engineering, Vol. 2, No. 2, 1990, pp. 231-246.

[MTS 91] Mihalisin T., Timlin J., Schwegler J.: *'Visualizing Multivariate Functions, Data and Distributions',* IEEE Computer Graphics and Applications, Vol. 11, No. 3, 1991, pp. 28-35.

[MZ 92] Marchak F., Zulager D.: *'The Effectiveness of Dynamic Graphics in Revealing Structure in Multivariate Data',* Behavior, Research Methods, Instruments and Computers, Vol. 24, No. 2, 1992, pp. 253-257.

[PG 88] Pickett R. M., Grinstein G. G.: *'Iconographic Displays for Visualizing Multidimensional Data',* Proc. IEEE Conf. on Systems, Man and Cybernetics, IEEE Press, Piscataway, NJ, 1988, pp. 514-519.

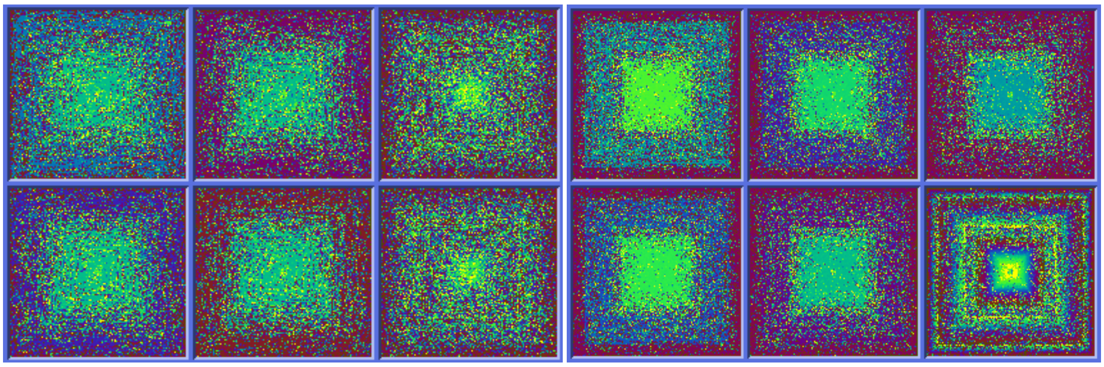
[Pic 70] Pickett R. M.: *'Visual Analyses of Texture in the Detection and Recognition of Objects',* in: Picture Processing and Psycho-Pictorics, Lipkin B. S., Rosenfeld A. (eds.), Academic Press, New York, 1970.

[SB 88] Salton G., Buckley C.: *'Term-Weighting Approaches in Automatic Text Retrieval',* Information Processing and Management, Vol. 24, No. 5, 1988, pp. 513-523.

[SGP 91] Smith S., Grinstein G., Pickett R.: *'Global Geometric, Sound, and Color Controls for Iconographic Displays of Scientific Data',* in: Extracting Meaning from Complex Data: Processing, Display, Interaction II, Vol. 1459, 1991, pp. 197-206.

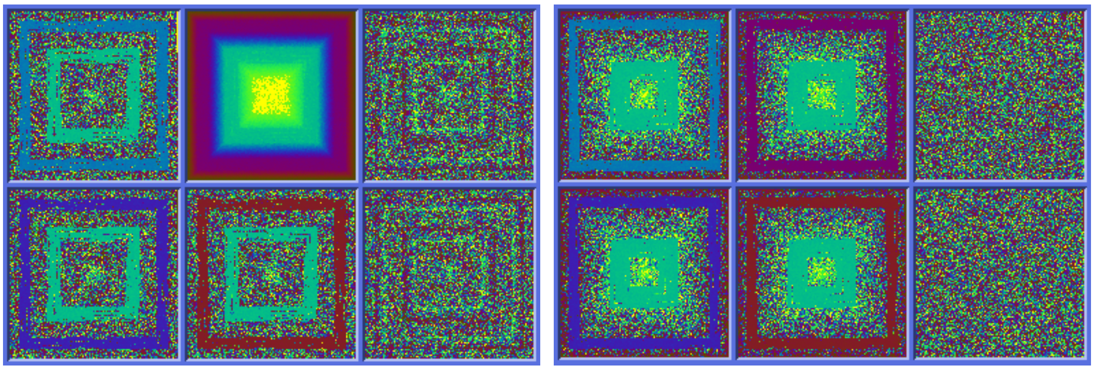
[Tuf 83] Tufte E. R.: *'The Visual Display of Quantitative Information',* Graphics Press, Cheshire, CT, 1983.

[Tuf 90] Tufte E. R.: *'Envisioning Information',* Graphics Press, Cheshire, CT, 1990.



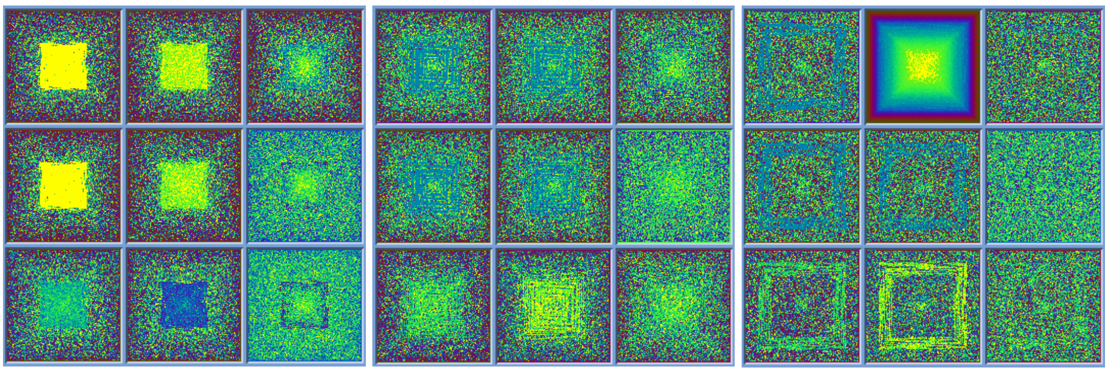
a. Vierdimensionale Cluster b. Fünfdimensionale Cluster

Abb. 9: Visualisierung von Clustern unterschiedlicher Dimensionalität



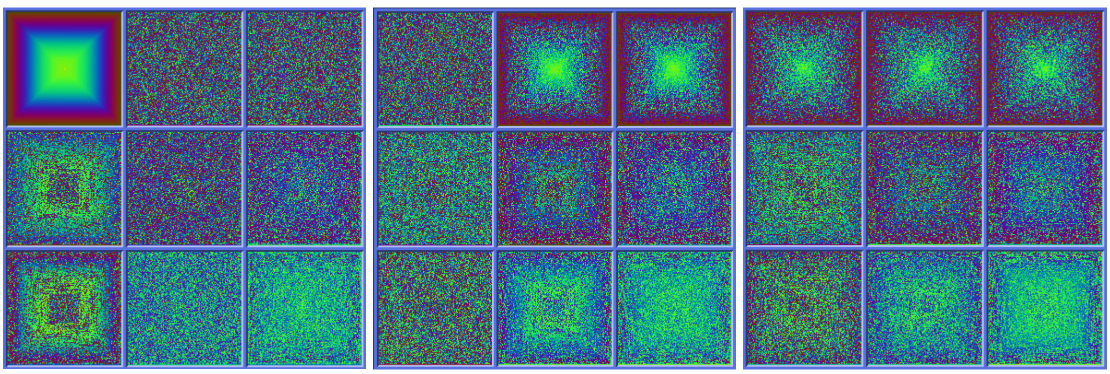
a. Höheres Gewicht für Attribut 2 b. Höheres Gewicht für alle Clusterattr.

Abb. 10: Effekt geänderter Gewichtungsfaktoren



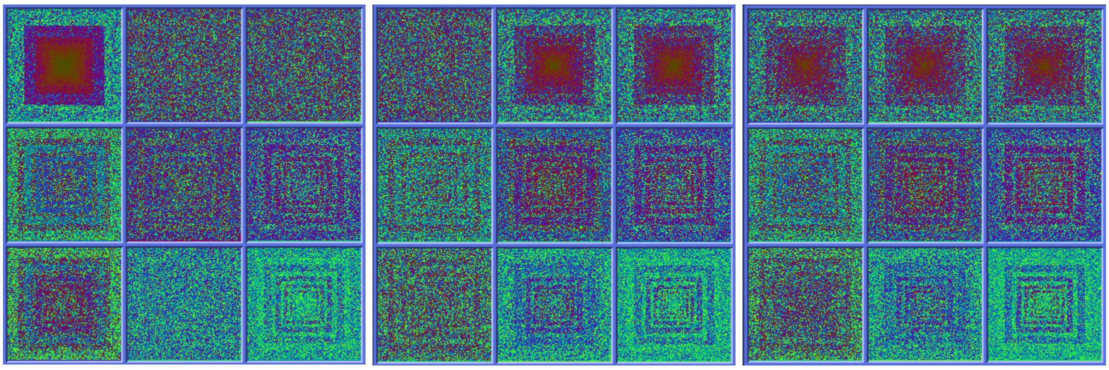
a. Anfrageregion [0, 10] b. Anfrageregion [4900, 5000] c. Höheres Gewicht für Dim. 2

Abb. 11: Cluster mit verschiedenen Datenverteilungen



a. Höheres Gewicht auf Dim. 1 b. Höheres Gewicht auf Dim. 2-3 c. Höheres Gewicht auf Dim. 1-3

Abb. 12: Cluster mit funktionalen Abhängigkeiten



a. Höheres Gewicht auf Dim. 1 b. Höheres Gewicht auf Dim. 2-3 c. Höheres Gewicht auf Dim. 1-3

Abb. 13: Effekt der Farbinvertierung

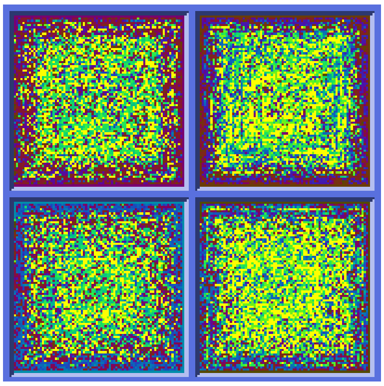
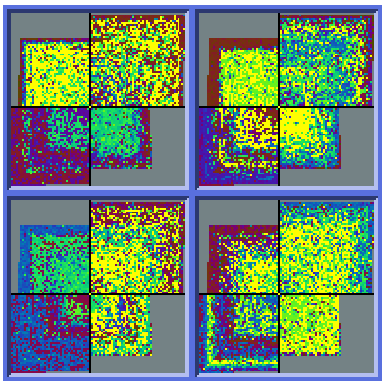
a.  b. 

Abb. 14: Vorteil der Achsenanordnung

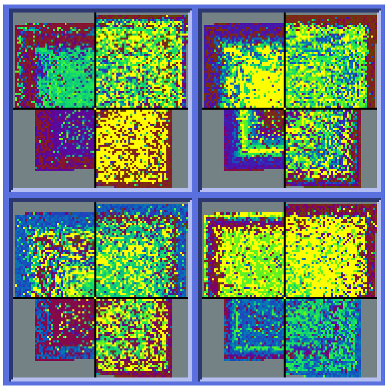
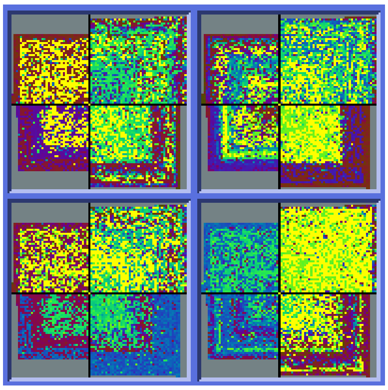
a.  b. 

Abb. 15: Auswirkung verschiedener Zuordnungen von Attributen zu den Achsen

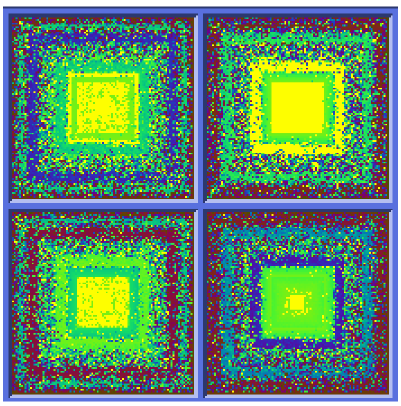
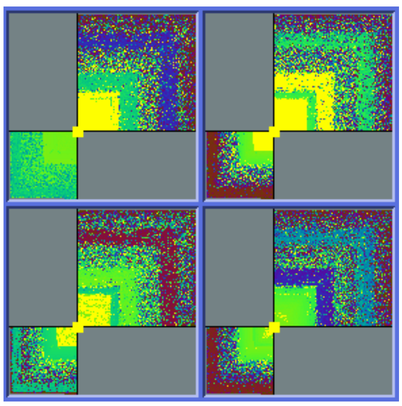
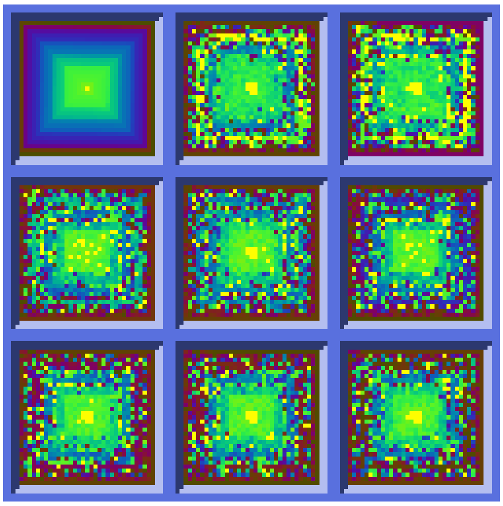
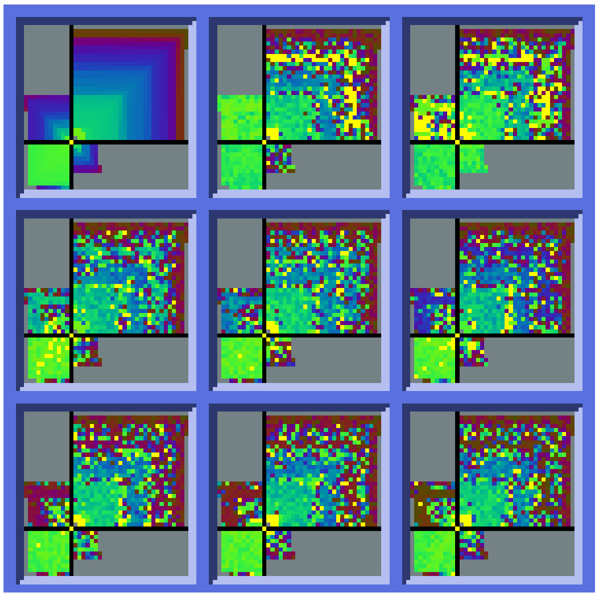
a.  b. 

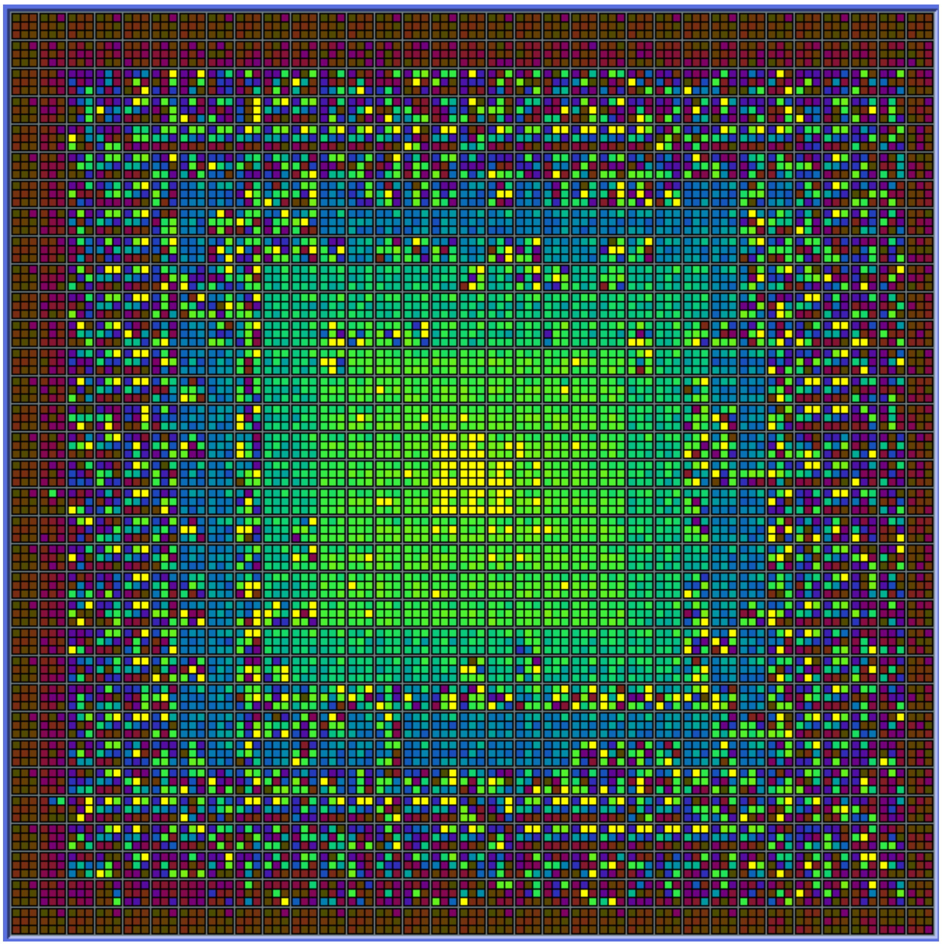
Abb. 16: Nachteil der Achsenanordnung



a. Spiralanordnung

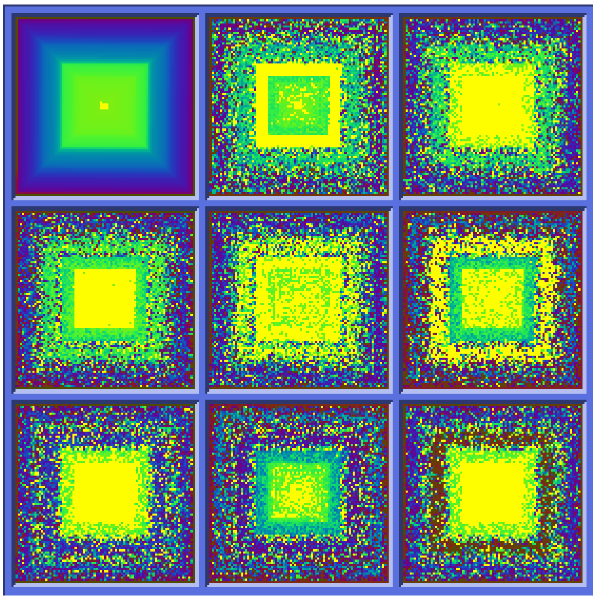


b. Achsenanordnung

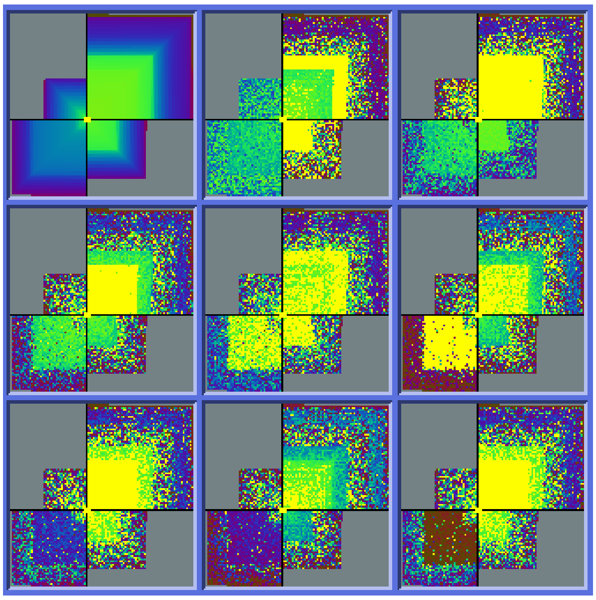


c. Gruppenanordnung

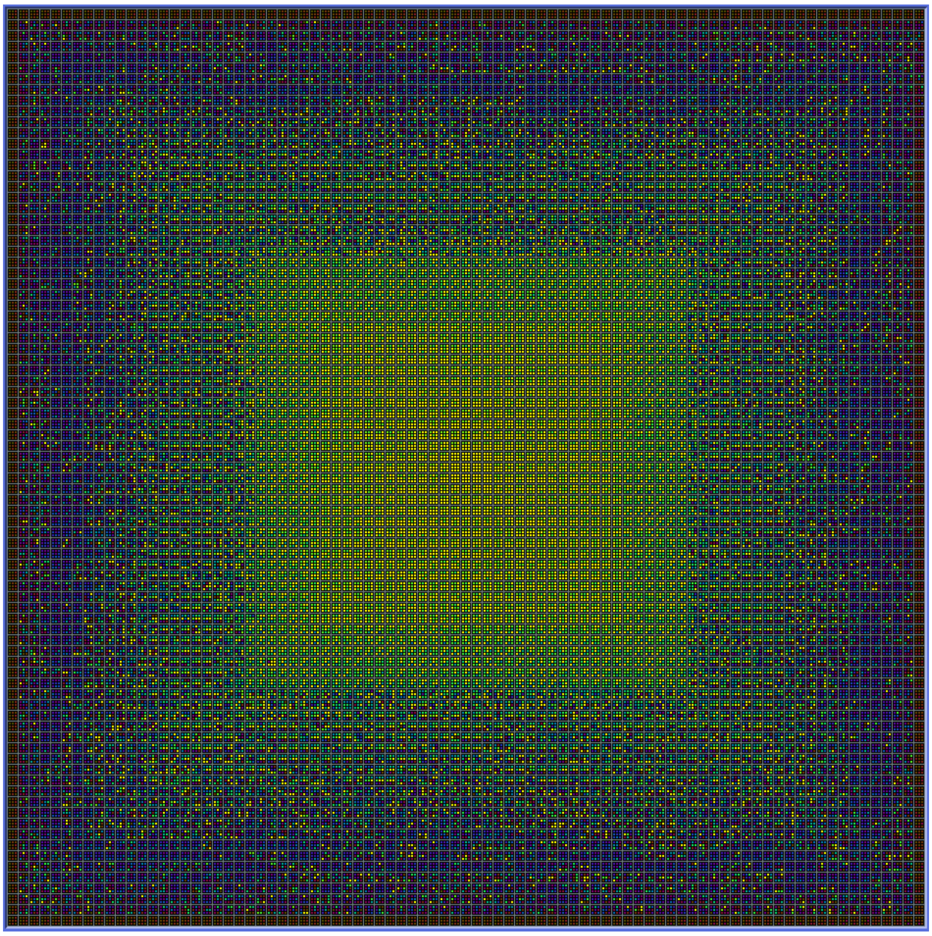
Abb. 17: Achtdimensionale Daten (1.000 Datensätze)



a. Spiralanordnung



b. Achsenanordnung



c. Gruppenanordnung

Abb. 18: Achtdimensionale Daten (7.000 Datensätze)