Information Retrieval for Multivariate Research Data Repositories

vom Fachbereich Informatik
der Technischen Universität Darmstadt
genehmigte

DISSERTATION

zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
von

Dipl.-Inf. Maximilian Scherer
geboren in Offenbach, Deutschland

Referenten der Arbeit: Prof. Dr. techn. Dieter W. Fellner
Technische Universität Darmstadt
Prof. Dr. Tobias Schreck
Universität Konstanz

Tag der Einreichung: 30.09.2013
Tag der mündlichen Prüfung: 02.12.2013

Erscheinungsjahr 2013
Darmstädter Dissertation
D 17

Erklärung zur Dissertation

Hiermit versichere ich die vorliegende Dissertation selbständig nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort: Darmstadt, den 30.09.2013

Unterschrift: __________________________
(Maximilian Scherer)

Für meinen Vater [Sch80]
* 03.12.1948
† 05.01.2013

Acknowledgments

Throughout this dissertation *I* will be using the first-person to describe the work done in the scope of my time as a researcher and PhD student at the Interactive-Graphics System Group at TU Darmstadt. However, I want to clearly point out that my work would not have been possible without the following advisors, colleagues, institutions, friends and family, to all of whom I am very grateful.

I want to thank my primary PhD advisor, Prof. Dr. Dieter W. Fellner, who supported me in my decisions, yet always challenged me to thought-provoking discussions.

I would like to thank my secondary advisor Prof. Dr. Tobias Schreck, under whose supervision I had the pleasure of starting my work as a PhD student in Darmstadt. His supportive and positive attitude encouraged me to look for and ultimately find my own topic of research I could pursue. Despite his call to Konstanz, he continued to supervise, support and encourage my work. Thank you!

I am also very thankful to Dr. Tatiana von Landesberger who started to supervise and support my work in Darmstadt after Prof. Schreck's leave. She always challenged me to improve my work and had the indispensable quality to ask the critical, yet very important questions.

A final advisor-related thank-you goes out to Dr. Arjan Kuijper. In his position as research coach I could always ask him questions and discuss important topics at a moment's notice.

Of course I am grateful for all the minor and major discussions with Prof. Dr. Michael Goesele, Prof. Stefan Roth, PhD and all fellow PhD students and researchers at GRIS and Fraunhofer IGD.

I would also like to thank the administrative team at GRIS, namely Carola Eichel, Silke Romero and Nils Balke, who made working there a smooth experience.

I am very grateful for the funding I received from DFG, EXIST and SoftwareCampus.

A special thank-you to two colleagues and friends. Jürgen Bernard for lots of interesting discussions, collaborations and a lot of fun at work and beyond. Eduard Rosert for his ideas, his perfectionism and of course for writing kick-ass apps with me.

And a shout-out to my DotA team, GG guys!

I would like to thank my entire, beloved family for always supporting and encouraging me, and for being proud of me no matter what. Particularly my mother and my late father.

With this last line I am thanking my girlfriend (and coincidentally colleague) Meike Becker – you really did change my life.

Abstract

In this dissertation, I tackle the challenge of information retrieval for multivariate research data by providing novel means of content-based access.

Large amounts of multivariate data are produced and collected in different areas of scientific research and industrial applications, including the human or natural sciences, the social or economical sciences and applications like quality control, security and machine monitoring. Archival and re-use of this kind of data has been identified as an important factor in the supply of information to support research and industrial production. Due to increasing efforts in the digital library community, such multivariate data are collected, archived and often made publicly available by specialized research data repositories. A multivariate research data document consists of tabular data with *m* columns (measurement parameters, e.g., temperature, pressure, humidity, etc.) and *n* rows (observations). To render such data-sets accessible, they are annotated with meta-data according to well-defined meta-data standard when being archived. These annotations include time, location, parameters, title, author (and potentially many more) of the document under concern. In particular for multivariate data, each column is annotated with the parameter name and unit of its data (e.g., *water depth [m]*).

The task of retrieving and ranking the documents an information seeker is looking for is an important and difficult challenge. To date, access to this data is primarily provided by means of annotated, textual meta-data as described above. An information seeker can search for documents of interest, by querying for the annotated meta-data. For example, an information seeker can retrieve all documents that were obtained in a specific region or within a certain period of time. Similarly, she can search for data-sets that contain a particular measurement via its parameter name or search for data-sets that were produced by a specific scientist. However, retrieval via textual annotations is limited and does not allow for content-based search, e.g., retrieving data which contains a particular measurement pattern like a linear relationship between water depth and water pressure, or which is similar to example data the information seeker provides.

In this thesis, I deal with this challenge and develop novel indexing and retrieval schemes, to extend the established, meta-data based access to multivariate research data. By analyzing and indexing the data patterns occurring in multivariate data, one can support new techniques for content-based retrieval and exploration, well beyond meta-data based query methods. This allows information seekers to query for multivariate data-sets that exhibit patterns similar to an example data-set they provide. Furthermore, information seekers can specify one or more particular patterns they are looking for, to retrieve multivariate data-sets that contain similar patterns. To this end, I also develop visual-interactive techniques to support information seekers in formulating such queries, which inherently are more complex than textual search strings. These techniques include providing an over-view of potentially interesting patterns to search for, that interactively adapt to the user's query as it is being entered. Furthermore, based on the pattern description of each multivariate data document, I introduce a similarity measure for multivariate data. This allows scientists to quickly discover similar (or contradictory) data to their own measurements.

Zusammenfassung

Diese Dissertation beschäftigt sich mit der Herausforderung der inhaltsbasierten Suche in Sammlungen multivariater Forschungsdaten.

Multivariate Forschungsdaten werden in immer größerem Maße in vielen Wissenschaftsdisziplinen, wie den Human- und Naturwissenschaften oder den Sozial- und Wirtschaftswissenschaften, erhoben. Das Archivieren und Wiederverwerten dieser Daten spielt eine immer wichtigere Rolle in der Informationsversorgung. Hierzu wurden spezialisierte Repositorien geschaffen, die diese Daten archivieren und zur Nachnutzung bereitstellen. Ein multivariater Datensatz beinhaltet dabei m Messgrößen (zum Beispiel Temperatur, Druck, Feuchtigkeit, etc. in der Klimaforschung) und n Beobachtungen. Um solche Datensätze in den Repositorien auffindbar zu machen, werden diese nach einem gewissen Metadatenstandard textuell annotiert und können anhand dieser Annotation gesucht werden. Diese annotierten Metadaten beinhalten beispielsweise Ort, Datum, Messgrößen, Autor, Titel, etc. des zugrundeliegenden Datensatzes. Insbesondere bei multivariaten Daten werden insbesondere die einzelnen Spalten annotiert, um eindeutig festzuhalten, welche Messgröße und Einheit die einzelnen Spalten wiedergeben. Nach diesem Stand können Wissenschaftler ihren Informationsbedarf derzeit decken, indem sie für sie relevante Datensätze anhand der Metadaten finden. Beispielsweise können alle Datensätze gefunden werden, die in einem gewissen Zeitraum oder innerhalb gewisser geographischer Grenzen erfasst wurden. Ebenso können jene Datensätze gefunden werden, die Messungen zu einer bestimmten Messgröße (z.B. Wasserdruck) enthalten oder von einem bestimmten Wissenschaftler aufgenommen wurden. Fragestellungen, die nicht oder nur unzulänglich mit Hilfe textueller Annotationen beantwortet werden können, beinhalten beispielsweise die Suche nach einem speziellen Muster in den multivariaten Daten, wie etwa ein linearer Zusammenhang von Wasserdruck und Wassertiefe. Eine andere solche Fragestellung ist die Suche nach multivariaten Daten, die einem Beispieldatensatz möglichst ähnlich sind, das heißt, solche Datensätze die ähnliche Muster wie der Beispieldatensatz aufweisen.

In dieser Dissertation beschäftige ich mich mit diesen Herausforderungen und entwickle neue Verfahren, um den etablierten Zugang zu multivariaten Forschungsdaten auf Annotationsbasis, durch inhaltbasierte Ansätze zum Beschreiben der Muster innerhalb der Daten zu erweitern. Damit erhöhe ich das Maß an Zugänglichkeit zu diesen Daten, durch die Unterstützung verschiedener Such- und Explorationsmodalitäten, die für die Auffindbarkeit und damit die Nachnutzung der Datensätze entscheidend sind. Durch Analyse und Merkmalsbeschreibung der multivariaten Daten selbst werden Suchanfragen ermöglicht, die anhand der Metadaten allein nicht durchführbar gewesen wären. Dies erlaubt die Suche nach jenen Datensätzen, deren Messungen ein bestimmtes Muster (beispielsweise den bereits oben erwähnten linearen Zusammenhang von Wasserdruck und Wassertiefe) vorweisen. Ebenso entwickle

ich visuell-interaktive Verfahren, um den Nutzer bei der Formulierung solch komplexer Suchanfragen zu unterstützen. So kann beispielsweise eine Übersicht interessanter Muster präsentiert werden, die sich in Echtzeit an die (Teil-)Suchanfrage des Nutzers anpasst. Weiterhin habe ich über die Merkmalsbeschreibungen einzelner Datensätze ein Maß zur Bestimmung der Ähnlichkeiten zwischen multivariaten Datensätzen entwickelt. Dies erlaubt Wissenschaftlern mittels Beispieldatensätzen andere Datensätze gemäß ihrer Ähnlichkeit aufzufinden. So kann beispielsweise schnell festgestellt werden, ob andere Wissenschaftler zu ähnlichen (oder auch widersprüchlichen) Ergebnissen gekommen sind.

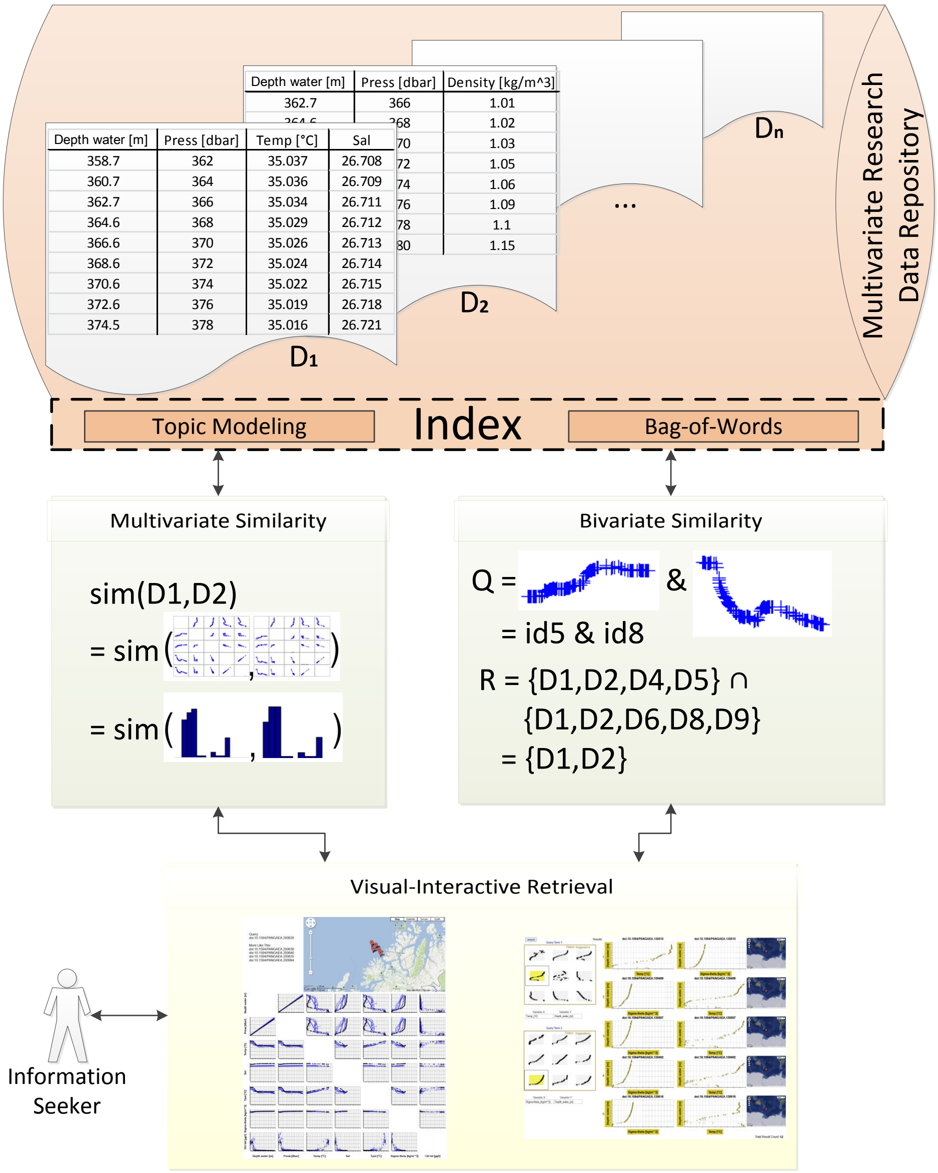
Contents

1 Introduction

This thesis is the culmination of the research I have conducted in the area of information retrieval for multivariate research data over the past three to four years. During this time, I have witnessed research data become a major focus in the digital library community. Different aspects like retrieval, archiving, privacy, data formats, accessibility, meta-data standards, and many more have been investigated and still are being investigated and discussed this very moment. With my already published research, and ultimately this thesis, I focus on retrieval for multivariate research data. In the following sections, I will motivate why this is an important aspect to further enhance our current and future handling of research data, and I will also outline the specific challenges associated with retrieval of multivariate data.

1.1 Motivation

Multivariate research data are produced in ever-more increasing amounts on a daily basis in many areas of research, industrial production and other commercial applications. A multivariate research data document consists of tabular data with m columns (measurement parameters) and n rows (observations) along with annotated meta-data for each column (for example, parameter name and base unit, e.g. *water depth [m]*). Due to increasing efforts in the digital library community over the last decade, such data, particularly data obtained for research purposes, is archived and made publicly available on a large scale in specialized research data repositories and annotated with high-quality meta-data. One research domain where such data repositories are particularly important is earth observation. There, a wide range of sensors and remote-sensing devices (like satellites) are used to measure environmental parameters across all continents, oceans, the atmosphere and the poles. These sensorics are globally connected to earth observation networks. In cooperation with digital libraries, these jointly collected measurement data-sets are archived, curated and made available for re-use and citation. Similar to repositories in other domains, such as web documents, images or other multimedia documents, the task of retrieving and ranking the documents an information seeker is looking for, is an important and difficult challenge. One established way to tackle this challenge is by means of annotated, textual meta-data that an information seeker can search for. However, retrieval via textual annotations is limited and does not allow to search for data patterns themselves. As a motivational example, consider the task of retrieving data which is similar to example data an information seeker provides. To allow for these kinds of query tasks, content-based access to research data repositories is required and has thus started to receive attention from the digital library community. In addition to annotated textual meta-data, such access supports information seekers to search and explore data patterns and to find data that exhibits patterns similar to query patterns an information seeker specifies, e.g., via example data, sketching or



Depth water [m]	Press [dbar]	Density [kg/m^3]
362.7	366	1.01
364.6	368	1.02
	70	1.03
	72	1.05
	74	1.06
	76	1.09
	78	1.1
	80	1.15

Depth water [m]	Press [dbar]	Temp [°C]	Sal
358.7	362	35.037	26.708
360.7	364	35.036	26.709
362.7	366	35.034	26.711
364.6	368	35.029	26.712
366.6	370	35.026	26.713
368.6	372	35.024	26.714
370.6	374	35.022	26.715
372.6	376	35.019	26.718
374.5	378	35.016	26.721

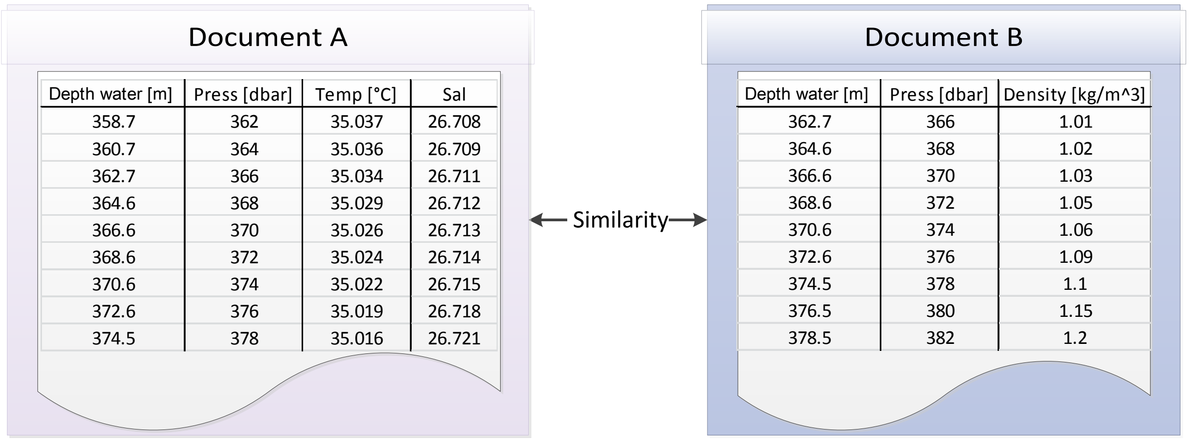
Figure 1.1: Schematic Outline of my Approach for Information Retrieval for Multivariate Research Data Repositories. A collection multivariate data documents *D* is indexed for retrieval. A bag-of-words approach for bivariate similarity (Section 3.4.1) allows an information seeker to retrieve all data-sets *R* that contain the patterns she specified in query *Q*. Multivariate similarity is computed with a topic modeling approach (Section 3.4.2) that enables the retrieval of multivariate data-sets that exhibit a similar scatter-plot-matrix as an example data-set the information seeker provides.

mathematical terms. The motivation behind such access is to enable new and potentially more effective ways for researchers to search for related work, data-sets and experiments that support or contradict their own work, as well as getting an overview of the research data obtained and used thus far.

In the scope of this thesis, I tackle the challenge of information retrieval for multivariate research data and propose novel indexing and retrieval schemes, to render multivariate data documents accessible via the patterns of their actual content, on top of any text-based meta-data annotations. The ideas for my approach originate from visual analysis of multivariate data and content-based retrieval of multimedia documents, in particular content-based image retrieval. The goal is to extract *features* from multivariate data documents, that describe the content of these documents. These features enable content-based retrieval for information seekers. Here, the key to feature extraction is to derive features that describe and discriminate the patterns in multivariate data well. When we look in the area of information visualization, the most widely used technique to visualize and subsequently analyze multivariate data is the scatter-plot-matrix. To construct the scatter-plot-matrix from multivariate data, each of the m columns is plotted versus one another. This results in a square matrix that contains all $m \cdot (m-1)$ scatter-plots of the pair-wise column combinations. Since the information that a human observer can infer from the scatter-plot-matrix is suitable to analyze, understand and compare data, the idea is that features extracted from the scatter-plot-matrix are also suitable for retrieval purposes. In fact, constructing features using analysis techniques that potential information seekers are already used to, makes the retrieval process more transparent and comprehensible. To extract features from each scatter-plot, I develop and benchmark several novel bivariate feature extraction algorithms. These features enable a potential information seeker to look for data documents that contain one or several specific bivariate patterns, e.g. by specifying two variables via their label and sketching the relationship between these variables. For increased robustness and significantly better retrieval times, I propose to convert these features into a bag-of-words representation. These bag-of-words features associated with each multivariate data document can also be used for mining dominant or unusual patterns in a given data-set, as well as suggesting and auto-completing query terms to search for.

Motivated by the successful use of query-by-example in other retrieval domains, I extend this approach to allow the information seeker to specify an example document and the subsequent retrieval of documents with similar data. To this end, the bag-of-words representation of each document will be further analyzed by topic modeling. This is a state-of-the-art technique used in multimedia information retrieval, to enable content-based retrieval in an efficient way. Using Latent-Dirichlet-Analysis (LDA), a topic model for collections of multivariate data documents is learned and one can then represent each document as a mixture of topics. This representation allows us to compute the similarity of two multivariate research data documents by computing the distance of their respective topic activations. Besides query-by-example, this novel approach is very suitable for efficient nearest-neighbor indexing and clustering according to the topic distribution of a document.

Figure 1.1 shows a schematic outline of my approach.



Document A

Depth water [m]	Press [dbar]	Temp [°C]	Sal
358.7	362	35.037	26.708
360.7	364	35.036	26.709
362.7	366	35.034	26.711
364.6	368	35.029	26.712
366.6	370	35.026	26.713
368.6	372	35.024	26.714
370.6	374	35.022	26.715
372.6	376	35.019	26.718
374.5	378	35.016	26.721

Document B

Depth water [m]	Press [dbar]	Density [kg/m^3]
362.7	366	1.01
364.6	368	1.02
366.6	370	1.03
368.6	372	1.05
370.6	374	1.06
372.6	376	1.09
374.5	378	1.1
376.5	380	1.15
378.5	382	1.2

Figure 1.2: Similarity between multivariate documents: Multivariate data documents naturally differ in the number and types of dimensions. To compute a similarity measure between two such documents, the novel approach I developed is based on extracting a bag-of-words representation of each document and comparing their respective topic activations obtained by topic modeling (see Chapter 3).

1.2 Challenges

The challenges in information retrieval for multivariate research data can be summarized by asking the following question.

**"In a collection of multivariate data documents,
how can I find the documents I am looking for?"**

So far, the answer to that question has been to provide the information seeker with query tools for textual meta-data that was manually annotated to the research data. Such annotations can include generic, high-level meta-data information like the author, year, location or title of the experiment or publication for which the data was obtained. More specific annotations include labels for the measurement variables and units according to a standardized vocabulary (e.g., *Water Depth [m]* or *Press [dbar]*) of the multivariate data itself. Given these annotations and the tools to query them, the information seeker can retrieve data she is looking for, as long as her information need can be expressed by textual means and corresponds approximately to the terms chosen by the annotator. For example, querying for data documents that were obtained by a particular author, within a specific region or within certain period in time, is feasible. It is also feasible to query for multivariate data that contains one or more, specific measurement variables, e.g., to retrieve all documents that contain a measurement of *Water Depth [m]* and a measurement of *Press [dbar]* among potential other measurements. However, the limitations of such annotation-based access are reached, if, for example, an information seeker is looking for all data documents that contain a particular relationship between *water depth [m]* and *pressure [dbar]*. By relying merely on annotated, textual data, it is not possible to retrieve documents that contain data which exhibit a pattern *similar* to a specific query pattern, which might be specified by means of sketching or using an example document (e.g., data obtained by the information seeker herself). Another information need of the seeker that cannot be met by mere annotation-based access is to provide an overview, or grouping of data patterns and relationships.

The goal of this thesis is to enable content-based retrieval for multivariate research data as described above. Several aspects of multivariate data make this goal highly challenging. A major contributor is the fact that multivariate data is very heterogeneous, meaning in a collection of such data, documents will (in general) differ significantly in the number and types of data dimensions they contain. This heterogeneity can best be compared with the heterogeneity of a collection of textual documents, for example a collection of newspaper articles. There, the number of words would naturally differ from document to document, just like the choice of words. Analogously for collections of multivariate data, the number of measurement variables and the number of measurements would differ among documents, just like the choice of measurement variables that are being obtained.

This heterogeneity leads to several questions when we think about how to extract descriptors from such data that are suitable for retrieval. On the one hand, we want a descriptor that describes a multivariate document as a whole, as a single entity. This would allow for nearest-neighbor retrieval to find documents similar to a given query document. On the other hand though, we want a descriptor, or a set of descriptors that are able to account for partial similarities between documents. For example, when an information seeker queries for a particular functional relationship between two variables, we want to be able to retrieve all those multivariate documents, that contain at least one measurement that is similar to the one specified.

To account for the second case described above – the retrieval of multivariate documents that contain one or more particular patterns – we are in the area of bivariate data retrieval. The challenge at hand here is to retrieve those 2D point-clouds (scatter-plots) that are most similar to a query 2D point-cloud. For this task, I developed and evaluated several different feature extraction techniques, to find out which yield the best retrieval results. Using the best performing technique, one can then extract a set of feature vectors from each multivariate data document to render this document accessible via each of these patterns. In particular, we extract $m(m-1)$ feature vectors from a multivariate data document with m columns. This corresponds to extracting a feature vector from each entry of the document's scatter-plot-matrix, which is a standard tool for visual analysis of multivariate data. To account for the second case described above – the retrieval of multivariate documents that contain one or more particular patterns – we are in the area of bivariate data retrieval. The challenge at hand here is to retrieve those 2D point-clouds (scatter-plots) that are most similar to a query 2D point-cloud. For this task, I developed and evaluated several different feature extraction techniques, to find out which yield the best retrieval results. Using the best performing technique, one can then extract a set of feature vectors from each multivariate data document to render this document accessible via each of these patterns. In particular, we extract $m(m-1)$ feature vectors from a multivariate data document with m

columns. This corresponds to extracting a feature vector from each entry of the document's scatter-plot-matrix, which is a standard tool for visual analysis of multivariate data.

The other kind of content-based access I want to support, is assessing the similarity between multivariate documents. This requires a descriptor that represents the patterns of a multivariate data document as a whole. Once such a descriptor is extracted, the process of computing the similarity between two multivariate documents is accomplished by measuring the distance between their respective descriptors. However extracting such a descriptor is challenging. The top part of Figure 1.2 shows an example of this challenge. Document A contains four columns of data annotated with *Depth water [m]*, *Press [dbar]*, *Temp [°C]* and *Sal* (Salinity) respectively. Document B contains just three columns of data annotated with *Depth water [m]*, *Press [dbar]* and *Density [kg/m^3]* respectively. When expert users are asked to assess the similarity of these two documents, an intuitive approach is to visualize the multivariate data using scatter-plots and then check if the scatter-plots of columns present in both data-sets (*Water Depth [m]* and *Press [dbar]* in this case) show a similar pattern.

In this thesis, I formalize this idea and propose a novel approach for computing a similarity measure between multivariate data documents. Motivated by the wide-spread usage of the scatter-plot matrix, a visualization technique used to analyze multivariate data, and the success of topic modeling in multimedia retrieval, I propose and investigate the following approach in my thesis. By extracting a bivariate feature vector from each scatter-plot of a document, one obtains a set of feature vectors that describe a multivariate data document. Converting these feature vectors to a bag-of-words representation, allows to learn a topic model for this type of data and ultimately represent each document as a mixture of topics. The topic mixture obtained with this approach represents the different feature patterns occurring in each document, and as such, allows for an effective similarity computation between multivariate documents by measuring the distance between their respective topic activations.

1.3 Thesis Structure

The remainder of this thesis is structured into the following chapters. In the subsequent Chapter 2, I will be providing and describing work that is related to my thesis. The entirety of my approach for retrieval in multivariate research data spans Chapter 3, Chapter 4 and Chapter 5. At first in Chapter 3, I will be introducing my approach schematically as well as detailing my feature extraction algorithms for bivariate data and multivariate data. In Chapter 4, I will describe how to benchmark these feature extraction techniques and evaluate them. The last part of my approach is presented in Chapter 5, where I deal with visual-interactive retrieval to actually provide an information seeker with tools based on my algorithmic approach she can use. After these three chapters on my approach itself, I will show-case the qualitative benefits of my approach using a case-study in climate research in Chapter 6. Finally in Chapter 7, I will draw conclusions from the research I conducted during this thesis and outline future work about digital research objects. Appended to this thesis is a short list of all my publications (Appendix A), a raw, real-world example of multivariate research data (Appendix B) and my curriculum vitæ (Appendix C).

2 Related Work

In this chapter I will introduce and describe previous work by other authors that is related to my thesis. At first I will describe a selection of techniques, algorithms and math that served as a starting point for my work. In particular, I will provide an overview of distance functions, clustering algorithms, topic modeling as well as metrics and ranking functions used in information retrieval. Then I will describe the state-of-the-art in several related retrieval domains, including textual retrieval, multimedia retrieval and time-series retrieval. I will also introduce related techniques from the visual analytics domain, such as result visualization and interactive querying which are also part of my work. Finally, in the last section of this chapter, I will focus on the state-of-the-art in current digital library applications, in particular for research data repositories in the area of earth observation.

2.1 Distance Functions

There is a wealth of distance functions to compute the distance between two vectors. Computing such a distance is essential to many tasks in pattern recognition, data mining and information retrieval. A comprehensive overview and categorization of distance functions is provided by Cha [Cha07]. In the scope of my thesis, the two distance functions used most often (usually for clustering or retrieval) include the Euclidean Distance and the cosine distance.

The Euclidean Distance (often denoted as L_2 Norm) is defined as

$$d_{\mathrm{L2}}(\vec{x},\vec{y}) \quad = \quad \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{2.1}$$

where $\vec{x}$ and $\vec{y}$ are n-dimensional vectors of real numbers.

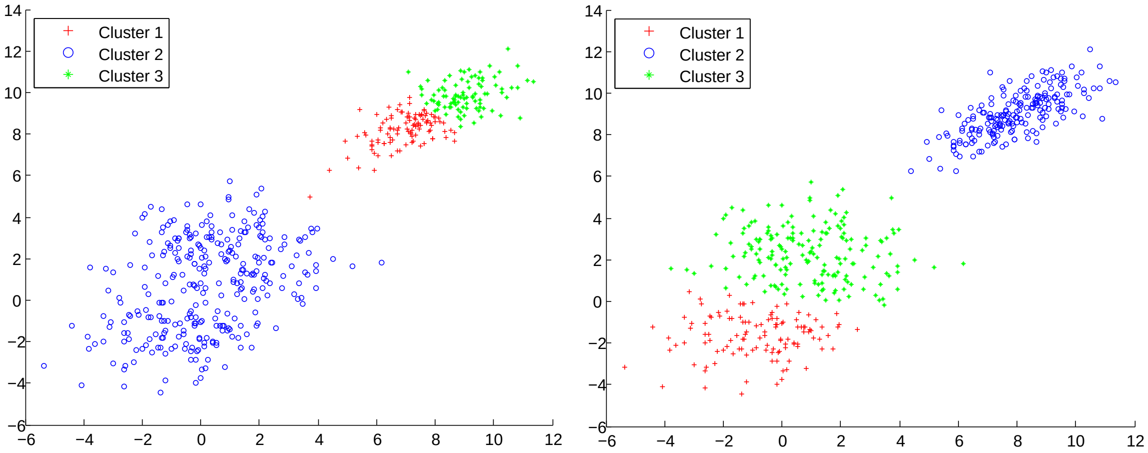
The cosine distance is defined as

$$\begin{aligned} d_{\cos}(\vec{x},\vec{y}) &= 1 - \frac{\vec{x}\cdot\vec{y}}{||\vec{x}||\,||\vec{y}||} && (2.2) \\ &= 1 - \frac{\sum\limits_{i=1}^{n} x_i y_i}{\sqrt{\sum\limits_{i=1}^{n} x_i^2} \cdot \sqrt{\sum\limits_{i=1}^{n} x_i^2}}. && (2.3) \end{aligned}$$

If vectors $\vec{x}$ and $\vec{y}$ are non-negative (i.e. frequency vectors), the cosine distance always lies in the $[0,1]$ interval, which is often advantageous for further processing, as no additional normalization is required.

Both of these distance functions are *metrics*, meaning they are non-negative and symmetric, preserve the identity of indiscernibles and adhere to the triangle inequality. These are important properties, especially for applications of distance functions in retrieval and clustering, e.g. space partitioning for fast, approximate distance computations. Without adherence to the triangle inequality for example, one could not guarantee an upper bound for the error for this kind of approximation.

2.2 Clustering



(a) *k*-means clustering

(b) Gaussian mixture clusters using EM

Figure 2.1: Clustering Example: (a) *k*-means clustering is neighborhood based and assigns data points to their closest centroid. (b) Fitting a Gaussian mixture model to the data using expectation-maximization ultimately assigns data-points to clusters based on their density.

Clustering is the process of assigning data instances to groups with respect to some kind of optimality criterion in an unsupervised manner. Several such clustering algorithms have been proposed so far, and their advantages and disadvantages have been studied in detail in the last two decades [Ber06, JMF99, XW*05]. In general, clustering algorithms can be discerned whether they assign cluster memberships in a deterministic or probabilistic way. Deterministic means that a data instance is assigned to one or several clusters and is thus not a member of any other clusters. In contrast, probabilistic clustering computes a probability density for each data instance over all clusters. Another aspect of clustering algorithms (either deterministic or probabilistic) is whether they preserve a given topology or not. A prominent example is the self-organizing maps algorithm proposed by Kohonen (thus also known as *Kohonen-Maps*) [Koh82] which preserves the neighborhoods of a cell-based topology. This algorithm is used mostly for visual-interactive approaches, as it directly lends itself to provide an overview of data by computing and visualizing a data-clustering on a 2D grid.

For this work, my primary application of clustering algorithms is the quantization of feature vectors (see Section 3.4.1). For this purpose, topology preservation is not required. In contrast, preserving a topology leads to poorer clustering results compared to non-topology preserving techniques [BLBS11]. Thus, I primarily used one of the most wide-spread clustering algorithms called k-means [Mac67, Ste56]. This is a deterministic, non-topology preserving clustering algorithm, that iteratively assigns data instances to one of k clusters while trying to minimize the distance of each instance to the current cluster mean. This process is iterated until the cluster memberships do not change between one iteration and the next. If run-time and not optimality needs to be guaranteed, the algorithm is either iterated a fixed number of times or until the distance to the cluster means converges below a given threshold. The two major challenges with k-means clustering in general are finding a suitable number of clusters k (which has to be set a-priori) and computing a suitable initial clustering. However, there has been lots of research about best practices to set these parameters in general, from the still often used rule of thumb $k = \sqrt{n/2}$ (n being the number of data-points) to more advanced statistical analysis [TWH01]. Furthermore there are several reference implementations for different applications (e.g., bag-of-words retrieval [YJHN07] in the case of this thesis), that can serve as a starting point to optimize these parameters.

2.3 Topic Modeling

Topic modeling is a generative learning process that models documents as a mixture of a small number of topics. Latent-Dirichlet-Allocation (LDA) is a popular topic model proposed by Blei et al. in 2003 [BNJ03] which is also used in this work. The following is based on an introduction to probabilistic models and LDA in particular by David M. Blei [Ble12]. LDA is part of a larger field called generative probabilistic modeling. In this framework, data is treated as arising from a generative process including hidden variables. We can analyze the data by computing the conditional distribution of the hidden variables given the observed variables. Here, the observed variables are the words of a document and the hidden variables are the topic structures. The generative process assumes that each document is created by three steps. Randomly choose a distribution over the topics. For each word in the document, choose a topic according to this distribution. According to the word-distribution of that topic, randomly choose a word. See Figure 2.2 for an illustration of this generative process. More formally, the joint distribution of the hidden and observed variables is defined as

$$\begin{aligned} & p(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D}) && (2.4) \\ = & \prod_{i=1}^{K} p(\beta_i) \prod_{d=1}^{D} p(\theta_d) \prod_{n=1}^{N} p(z_{d,n}|\theta_d) p(w_{d,n}|\beta_{1:K}, z_{d,n}). && (2.5) \end{aligned}$$

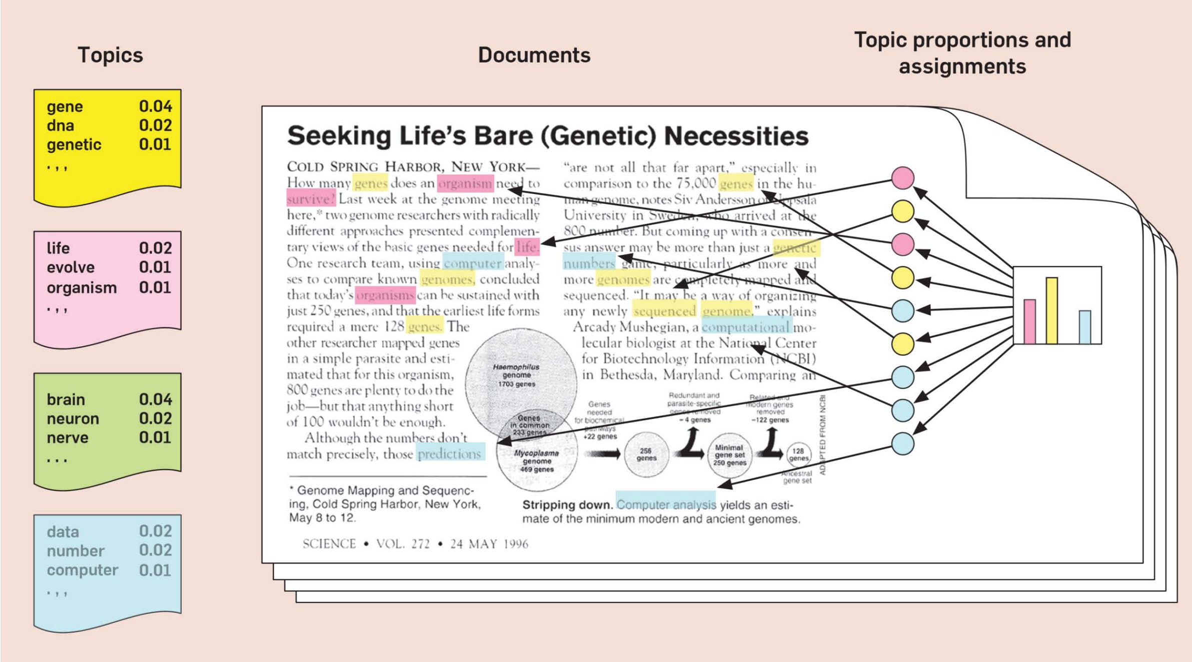

Topics
Documents
Topic proportions and assignments
gene 0.04
dna 0.02
genetic 0.01
, , ,
life 0.02
evolve 0.01
organism 0.01
, , ,
brain 0.04
neuron 0.02
nerve 0.01
. . .
data 0.02
number 0.02
computer 0.01
, , ,
Seeking Life's Bare (Genetic) Necessities
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
SCIENCE • VOL. 272 • 24 MAY 1996

Figure 2.2: Intuition Behind Topic Modeling. "We assume that some number of "topics," which are distributions over words, exist for the whole collection (far left). Each document is assumed to be generated as follows. First choose a distribution over the topics (the histogram at right); then, for each word, choose a topic assignment (the colored coins) and choose the word from the corresponding topic." Image and caption from "Probabilistic Topic Models" by David M. Blei [Ble12].

Whereas the notation is as follows: $\beta_{1:K}$ are the K topics where each topic β_k is a distribution over the words (vocabulary) of the corpus. D is the number of documents on the corpus; θ_d is the topic-distribution of document d. The topic assignments for all words of document d are z_d, where $z_{d,n}$ is the topic assignment of word n in document d. The observed words of document d are w_d, where again $w_{d,n}$ is word n in document d.

In the domain of non-textual documents, LDA was first applied to multimedia documents by Sivic and Zisserman for content-based image retrieval [SZ03]. The basic approach is to first extract a bag-of-words representation of the images by extracting local features, e.g., SIFT [Low04] or SURF features, and quantizing these features via k-means or other suitable clustering methods [XW*05]. Then each document can be represented by a set of tokens and thus, topic modeling in the form of LDA can be readily applied to obtain efficient nearest neighbor indexes of the document collection. Topic modeling has been shown to yield state-of-the-art retrieval performance in other domains and applications as well, including image and music retrieval [LSDJ06, RHG08, JDS10] as well as 3D models and 3D

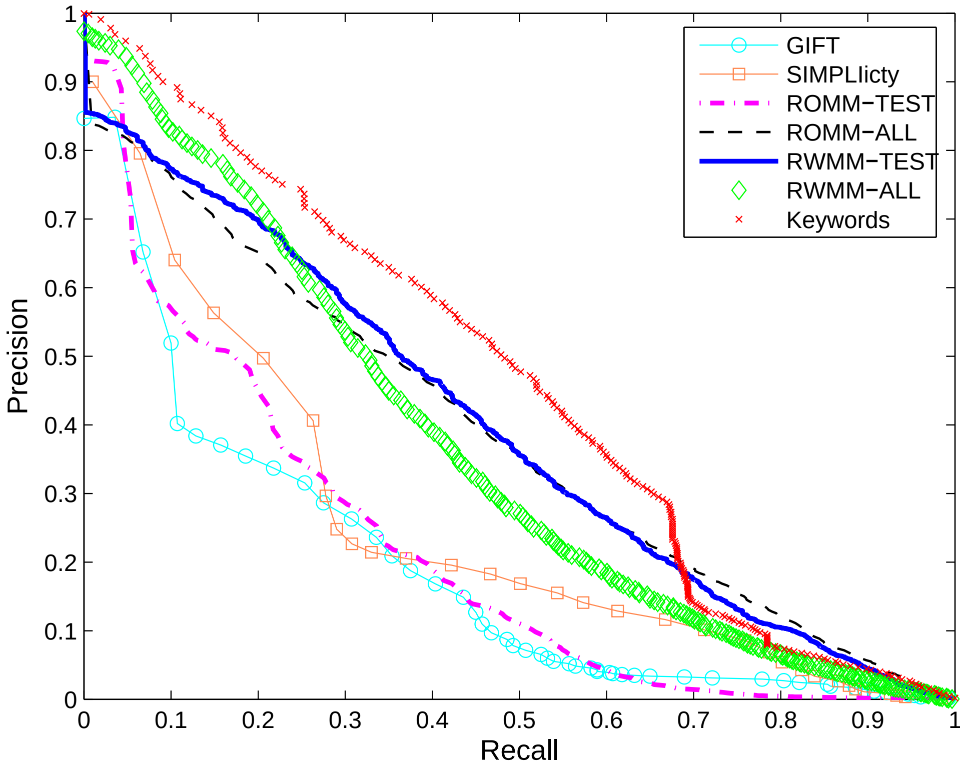


Figure 2.3: Example of precision and recall curves for the evaluation of content-based image retrieval techniques. Image by Shirahatti and Barnard [SB05].

scenes [ERB*12] and time-series data [LKL11]. For more details on applications using bag-of-words approaches and topic modeling please refer to Section 2.6.

2.4 Information Retrieval Metrics

Several metrics are used in information retrieval to judge the performance of ranking algorithms [BYRN99, MRS08]. Here I introduce the terminology and give a short summary of those metrics, which are also partly used for evaluation in the scope of this thesis (see Chapter 4).

Precision In the context of information retrieval, the higher the precision of a retrieval algorithm, the higher the ratio of relevant versus irrelevant documents in a result set. So precision is the fraction of retrieved documents that are relevant to the query. In more formal terms we notate precision as

$$\text{precision} = \frac{|\{\text{relevant docs}\} \cap \{\text{retrieved docs}\}|}{|\{\text{retrieved docs}\}|}. \tag{2.6}$$

In the context of classification, precision is also known as the *positive predictive value* (PPV) and denoted as

$$\text{PPV} = \frac{|\{\text{true positives}\}|}{|\{\text{true positives}\}| + |\{\text{false positives}\}|}. \tag{2.7}$$

Recall Recall is the fraction of relevant documents that are retrieved. So in the context of information retrieval, the higher the recall, the higher the ratio of relevant documents in the result set versus relevant documents in the corpus. We can formalize recall as

$$\text{recall} = \frac{|\{\text{relevant docs}\} \cap \{\text{retrieved docs}\}|}{|\{\text{relevant docs}\}|}. \tag{2.8}$$

Recall is also used in classification where it is known as *sensitivity*:

$$\text{sensitivity} = \frac{|\{\text{true positives}\}|}{|\{\text{true positives}\}| + |\{\text{false negatives}\}|}. \tag{2.9}$$

Mean Average Precision (MAP) For the purpose of evaluating retrieval algorithms, precision is usually measured at all different recall levels. The result of this is a *precision-recall curve*. To make two different precision-recall curves comparable a scalar measure that describes the precision-recall performance is desirable, as those can readily be compared. One such measure is the *mean-average-precision*, usually abbreviated *MAP*[1], being the mean value of all average-precision values of each query.

The average precision is computed as the mean of all precision values at the different recall levels. This can be accomplished by retrieving all n documents of a corpus (thus ranking/sorting the entire corpus) and evaluating the following finite sum.

$$\text{avgP} = \frac{\sum_{k=1}^{n} P(k) \cdot \text{I}_{\text{rel}}(k)}{m} \tag{2.10}$$

[1] not to be confused with *maximum a-posteriori estimation*, which is used as a point estimate in machine learning and abbreviated the same way

where m is the number of relevant documents, $P(k)$ is the precision at cut-off k in the result set and $\mathrm{I}_{\mathrm{rel}}(k)$ is a function that returns 1 if the document at result position k is relevant to the query and 0 otherwise.

First Tier Precision The first-tier-precision (also called r-precision or precision at r) evaluates the precision at a single point but is still highly correlated to the mean-average-precision.

It measures the precision of a retrieved result set of size r, where r is the number of documents relevant to the query. By definition, first-tier-precision is equal to recall at position r. We can compute the first-tier-precision by evaluating the following finite sum.

$$\mathrm{ftP} = P(r) \tag{2.11}$$

$$= \frac{\sum_{k=1}^{r} \mathrm{I}_{\mathrm{rel}}(k)}{r} \tag{2.12}$$

2.5 Textual Document Retrieval

The central challenge in textual document retrieval is related to multimedia retrieval and to retrieval of multivariate data which I consider in this thesis. This central challenge is to rank a collection of documents according to their respective relevance to a query. In textual document retrieval, such a query usually consists of one or several search terms. When querying a document collection with a limited, controlled meta-data vocabulary, it is usually suitable to compute a binary relevance, such that a document is relevant to a query if it contains all search tokens and irrelevant otherwise. However, when considering full-text retrieval (over natural language documents such as web-sites, news articles, etc.), computing a continuous relevance score is important to return a ranked list of documents to the information seeker.

One of the first and most straight-forward approaches to compute the relevance of a token to a document, is to count the (relative) number of occurrences of that token within the document [SB88]. This is referred to as the *term frequency*. To further improve this relevance judgment, it is often useful to normalize it using an *inverse-document frequency*. As the name implies, this is the relative number of documents within a given corpus, which contain the token at least once. This is intuitive, because even though a token occurs often in a given document, it might still be quite irrelevant if it occurs in almost all of the documents. This combined scheme is abbreviated *tf-idf* (term frequency, inverse document frequency). State-of-the-art techniques like *Okapi BM25* [RZ09] for ranked retrieval incorporate further refinements (such as a term that penalizes very long documents) and allow for weighting the individual relevance terms to fine-tune the retrieval scheme for the application at hand.

I describe Okapi BM25 here in detail, as I will be using this state-of-the-art technique for indexing and ranked retrieval of multivariate research data (see Section 3.5 Retrieval Scheme). BM25 is a family of functions for ranked retrieval of documents using a bag-of-words representation. The bag-of-words assumption states that all words (usually called tokens) of a document are considered independent of their location within the document and their proximity to other words. Given a collection (usually called corpus) of n documents $D_1 \dots D_n$, I compute the BM25 score of a document D_i to a query Q (containing the query tokens $q_1 \dots q_m$) using the following function [MRS08].

$$\text{sim}_{\text{bm25}}(D_i, Q) = \sum_{j=1}^{m} \text{IDF}(q_j) \cdot \frac{\text{TF}(q_j, D_i) \cdot (\alpha + 1)}{\text{TF}(q_j, D_i) + \alpha \cdot (1 - \beta + \beta \cdot \frac{|D_i|}{\frac{1}{n} \sum_{k=1}^{n} |D_k|})} \tag{2.13}$$

Here $\text{TF}(q_j, D_i)$ denotes the term frequency of token q_j in document D_i. $|D_i|$ is the length (number of tokens) of document D_i. α and β are weight parameters to fine-tune the ranking function to give more weight to the term frequency or more weight to the inverse document frequency.

$\text{IDF}(q_j)$ is the inverse document frequency of token q_j. It is given by

$$\text{IDF}(q_j) = \log \frac{n - n(q_j) + 0.5}{n(q_j) + 0.5}, \tag{2.14}$$

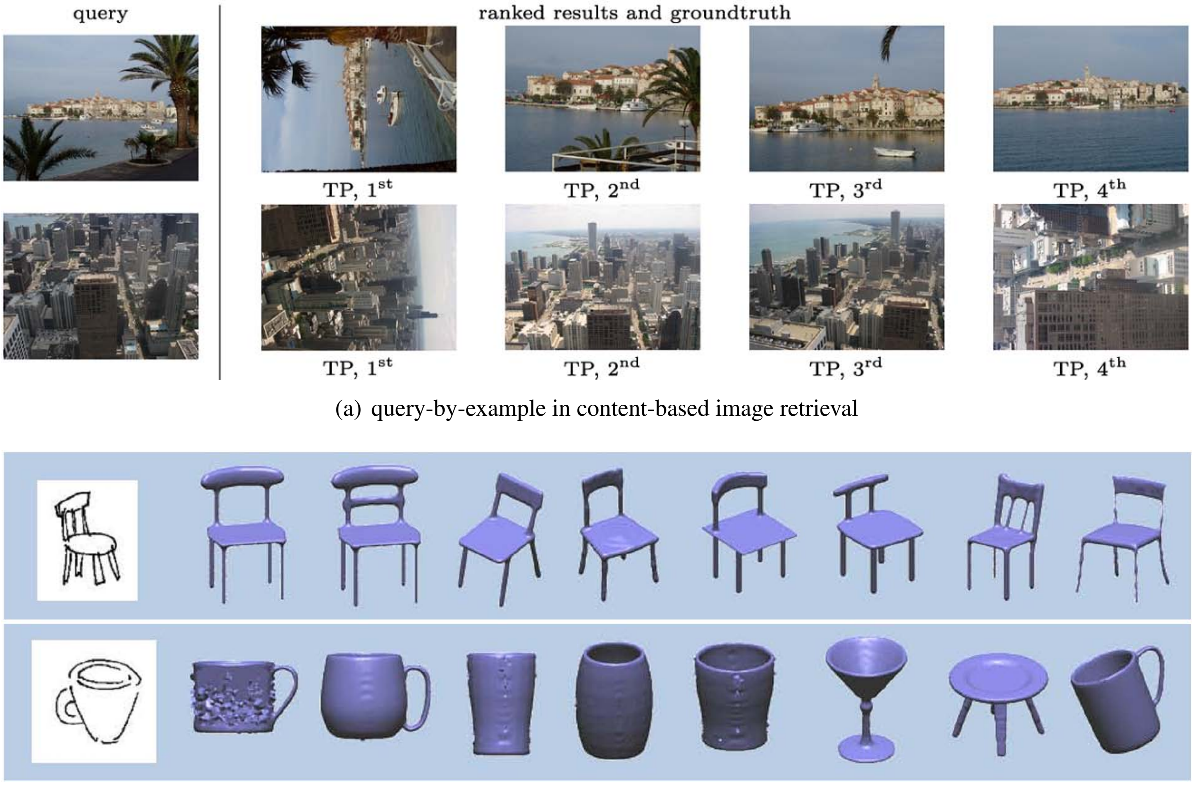
where n is the total number of documents in the collection and $n(q_j)$ is the number of documents that contain token q_j.

To evaluate and compare the retrieval performance of ranking algorithms like Okapi BM25, several benchmarking challenges exist. Three of the most prominent, large-scale retrieval challenges are TREC [TRE], CLEF [The] and NTCIR [NTC]. Tracks and challenges include web retrieval, micro-blog mining, image retrieval, patent retrieval, cross linking, math retrieval and several more. Older, but still widely-used benchmark data-sets for text retrieval and classification are collections of newspaper articles from Reuters called *Reuters-21578* and *RCV1* [LYRL04].

2.6 Multimedia Retrieval

Multimedia retrieval encompasses research how an information seeker can find non-textual, multimedia documents that are relevant to her information need [LSDJ06]. Multimedia retrieval focuses on content-based approaches, that is the analysis of the actual content of a multimedia document (e.g. the colors occurring in an image) and allowing for retrieval based (directly or indirectly) on this analysis.

What constitutes a multimedia document is loosely defined, although most research was done in the areas of image retrieval [DJLW08], music retrieval [TWV05], video retrieval [HXL*11] and 3D model retrieval [TV08].



(a) query-by-example in content-based image retrieval

(b) query-by-sketch in content-based 3d retrieval

Figure 2.4: Examples of Multimedia Information Retrieval Approaches. (a) Given an example query, similar images are retrieved. Image by Datta et al. [DLW05]. (b) By sketching what the information seeker is looking for, similar 3D models are retrieved. Image by Yoon et al. [YSSK10]

For retrieval, a popular approach is feature extraction. The goal of this approach is to extract a mathematically tractable summarization of the media content, such as a feature vector. A widely-used query paradigm relying on feature extraction is query-by-example, where an information seeker supplies an example document and the ranking algorithms retrieves documents similar to the query with respect to some of its properties according to the extracted features [SWS*00]. An intuitive example is content-based image retrieval using a color descriptor. By supplying an example image, other images with similar color distributions are retrieved.

A different approach that does not directly use extracted features for retrieval is the categorization of documents into classes according to their extracted features. Such approaches use machine learning algorithms to automatically classify the media documents. One example where a lot of learning data is available is the task to annotate textual tags to images. Given such automatic tags, an information seeker can use a textual search to retrieve documents of interest [LW08]. Other approaches exist that try to combine textual and visual cues [LCSS98].

Other retrieval approaches for visual documents like images [EHBA10] or 3D models [ERB*12] are based on sketching. In the absence of an example object, which arguably is often the case when searching for new documents, the information seeker can sketch what she is looking for. The basic idea here is to first convert the documents in the database such that they resemble user-drawn sketches. For images this can be accomplished using edge-detection, and for 3D models non-photo-realistic rendering techniques provide suitable views [YSSK10]. The draw-back to sketch-based methods is the dependence on the information seeker's drawing abilities, which can differ drastically from person to person.

Similar to sketch-based retrieval for visual documents, there exist humming-based approaches for music retrieval [KNS*00]. The idea here is to have the information seeker hum or sing a partial melody to use as a query.

Figure 2.4 shows two approaches in multimedia information retrieval, for content-based image retrieval and sketch-based 3d model retrieval. In (a) one can see that images similar to the query example are retrieved, even though they differ in rotation, composition and colors. In (b) 3D models that are similar to the hand-drawn query-sketches are retrieved.

To evaluate performance in multimedia information retrieval, many manually annotated data-sets exist and are used for benchmarking. The MPEG-7 benchmark is used for 2D shape analysis [LLE00]. In the area of 3D model retrieval there is the Princeton Shape Benchmark [SMKF04] as well as the SHREC (shape retrieval contest) [SHR]. Several large benchmarks and challenges for content-based image retrieval exist [DJLW08, DKN08], with ImageCLEF [Ima] being one of the most prominent. For these benchmarks, objects are usually assigned to similarity classes (either manually by humans, or automatically by using *social tags*, e.g., from Flickr), and precision-recall can be computed, to measure effectiveness of feature extraction algorithms for similarity assessment. However, their suitability is sometimes discussed [MMP02], since automatically designed benchmarks lack specified query sets and relevance judgments of retrieval results.

Retrieval for feature-vector based techniques relies on ranking these feature vectors to a query object, usually by means of a k-nearest neighbor computation. For large scale multimedia data-bases, efficient indexing structures are required, as exhaustive nearest neighbor search is too costly.

Approximate indexing relies on data structures such as KD-trees, random forests or hierarchical k-means trees [ML09], that allow to quickly skim through the feature space.

Other approaches rely on dimension reduction to speed up retrieval. One of the most successful such approaches is local sensitive hashing (lsh) [LJW*07]. The basic idea is to compute a hash value of high-dimensional feature vectors, such that similar feature vectors are assigned to the same bucket with a high probability.

An efficient content-based indexing method that has become highly popular in multimedia information retrieval is the bag-of-words (BOW) approach. It has been shown to yield state-of-the-art retrieval performance in different domains, including image and music retrieval [LSDJ06, RHG08, JDS10]. This

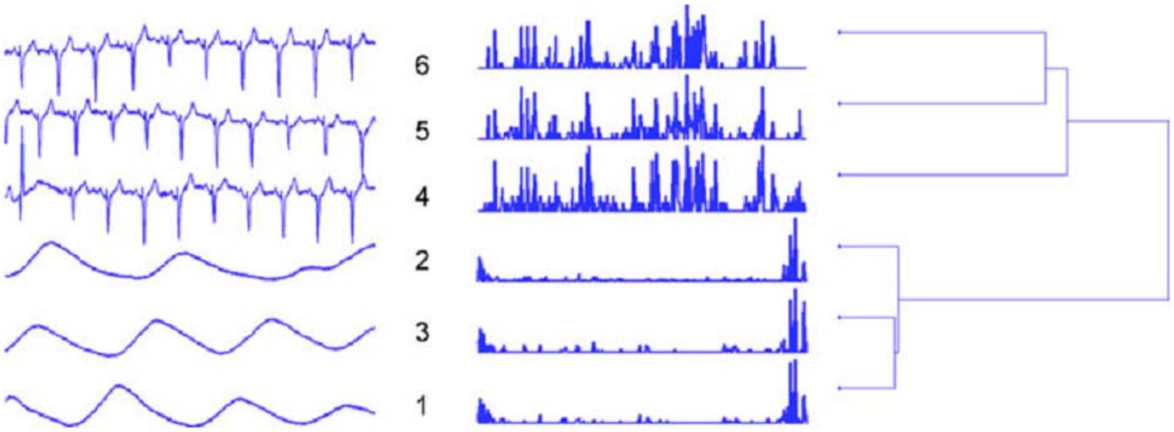


Figure 2.5: A bag-of-words approach for rotation invariant clustering of time-series data [LKL11]. The bag-of-words representation is invariant to shifts in the time-series. Image by Lin et al. [LKL11]

allows for similarity measurements between multimedia objects via their associated bag-of-words (usually the terms are encoded as a histogram), as well as querying or clustering the documents via specific terms (e.g., a predominant color in an image). Most recently, such a bag-of-words approach was also applied successfully to the retrieval of 3D models and 3D scenes [ERB*12]. I also use this approach for indexing multivariate data (see Section 3.4).

2.7 Time-Series Retrieval

A research topic in information retrieval that is closely related to this thesis is the retrieval of time-series data. Time-series data consist of one or more dependent variables (e.g. temperature, income, radiation, blood sugar, etc.) that are measured at specific intervals over one independent variable (usually time, hence the name *time-series*, though other variables like pressure might also be suitable). For time-series retrieval, the goal is to extract a mathematical descriptor from the data that models its properties. Given such a descriptor, similar data can be retrieved by computing the distance between descriptors. Time-series retrieval has received attention from the data-mining and information retrieval communities for over two decades. One of the first successful approaches is based on the discrete Fourier transform and approximates the data by extracting the first n Fourier coefficients from its Fourier transform [AFS93]. Other prominent approaches for time-series retrieval include the *piece-wise aggregate approximation* [YF00,KCPM01], which splits the data into n uniformly spaced parts and computes the average value of each part. Another approach which is particularly efficient for indexing and retrieving sub-sequences in time-series data is called *symbolic aggregate approximation* [LKLC03, KLF05]. Here the time-domain and the value-domain are averaged and quantized, such that time-series can be represented by

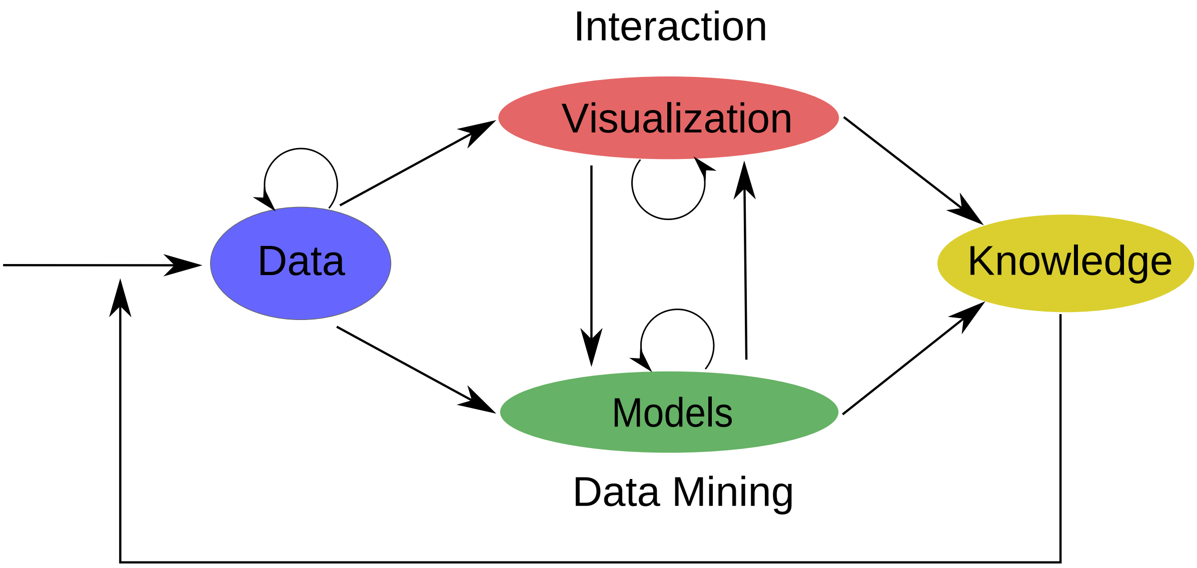


Figure 2.6: The Visual Analytics Pipeline [KAF*08].

a sequence of letters. More recent work shown in Figure 2.5 also considers the bag-of-words approach to create a rotation-invariant descriptor for time-series [LKL11].

In 2003, Keogh and Kasetty discussed the need for benchmarking retrieval in time-series data [KK03]. They empirically show that given a sufficiently large number of data-sets to choose from, the superiority of any technique can be shown when only considering numeric similarity of retrieval results. Thus, they argue for the need of similarity concepts to construct a meaningful benchmark. This leads to the creation of the UCR Time Series repository [KZH*11]. There, researchers using time-series data are invited to submit their (labeled) data-sets for others to use. In total, 47 time-series data-sets have been archived thus far. In each data-set, each individual time-series is annotated with a class label, which allows the evaluation of time-series descriptors via precision and recall and classification tests. Several time-series retrieval techniques were evaluated on these community data-sets by Ding et al. [DTS*08]

2.8 Visual-Interactive Analysis

Visual analytics is a relatively young research domain that attracts more and more attention from different research communities. Visual analytics is the combination of (semi-) automatic data analysis techniques and (interactive) visualization and has the goal, to provide users with insight and understanding of potentially complex and large data. The classical visual information seeking mantra by Shneiderman is to provide overview first and details on demand [Shn96]. Keim defined an extension to this mantra to explain the visual analytis process: Analyze first, show the important, zoom, filter and analyze further, details on demand [KAF*08].

In the area of time-series data and multivariate data, several visual analytics approaches exist to enable and support users in analyzing data of interest. Visual analysis of time-series data is, in contrast to retrieval of time-series data (see Section 2.7), usually not concerned with finding data of interest, but rather with analyzing data of potential interest. In general, time-series analysis has the goal to increase our understanding of systems, to distinguish regular from extraordinary characteristics [KLF05] and to predict future development [KS02]. Aigner et al. provide an overview of time-series visualization techniques [AMST11]. However, visualizing large time-series data-sets, particularly when using line charts, often leads to over-plotting if fitted into given display space or requires extensive user-interaction (e.g., for panning and zooming) otherwise [BWS*12]. Hence, data mining techniques are often used in time series visualization. The goal is to reduce the size of the data to be visualized. This can be achieved by resampling [AFS93], averaging [YF00], aggregating [BM04] or reducing the dimensionality of the time-series data [KCPM01].

For the analysis of multivariate data, a standard tool is the *scatter-plot matrix* (SPLOMs) [Cle85]. This square matrix consists of scatter-plots for all pair-wise column combinations of the multivariate data under concern. A prominent technique to filter or cluster large SPLOMs for those plots of highest potential interest to the user, based on certain interestingness scores, is *Scagnostics* by Wilkinson et al. [WAG05] The basic idea of this approach is to represent each scatter-plot as a graph and then compute graph-theoretic features to model such properties as *skinny*, *convex*, *skewed*, etc. These features allow for filtering, highlighting and aggregating individual scatter-plots of interest.

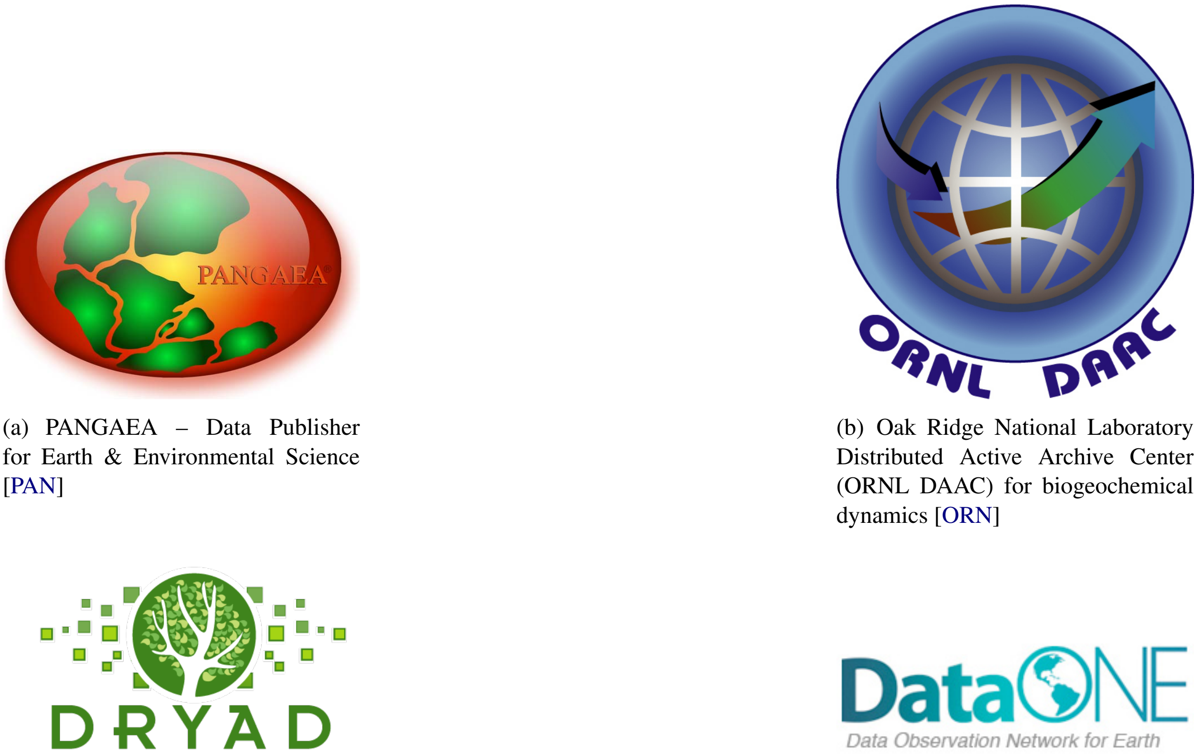
Visual access methods have shown to be highly successful for providing overview and search functionality for users in the digital library domain [Hea09]. Effective interfaces can help to more effectively browse, search or analyze large data repositories [WCR*11]. The major challenges in providing visual-interactive access to large amounts of data (which is typically the case in data libraries) include scalability, streaming analysis, user-driven data reduction and data summarization and triage for interactive querying [WSJ*12].

A recent example of such a system in the digital library context was presented in [BRS*12a]. By analyzing meta-data and time-series based content at the same time, this system generates an interactive layout of research data to enable the discovery of interesting co-occurrences of meta-data based and time-series based patterns. Such approaches can combine traditional meta-data based and content-based methods and can extend the standard search support with elements of exploratory search systems useful for hypothesis generation [WR09].

2.9 Digital Libraries

Digital library systems have evolved from mere research prototypes into production stable pieces of software, allowing us to cope with the rapidly increasing number of digital documents. Prominent digital-library systems include [CP02, LPSW06, WMBB00].

So far, these digital-library systems focus on *annotation-based* access to documents, as well as rendering *textual*-content accessible (e.g., by full-text search). This is well-suited for textual documents,



(a) PANGAEA – Data Publisher for Earth & Environmental Science [PAN]

(b) Oak Ridge National Laboratory Distributed Active Archive Center (ORNL DAAC) for biogeochemical dynamics [ORN]

(c) Dryad Digital Repository [Dry]

(d) DataONE – Data Observation Network for Earth [Dat]

Figure 2.7: Digital Data Libraries: Logos of four digital data libraries that make research data for different scientific domains available. All images are property of their respective copyright owners and / or registered trademark holders.

however support for *non-textual* documents usually relies on some metadata standard (e.g., MPEG7 for multimedia) and is often lacking appropriate *content-based* access (e.g., comparing similarity of images based on color distribution). Multimedia documents (e.g., audio, image, video, 3D models) and recently, research data gathered in natural and empirical sciences, have been recognized as important non-textual documents with a need for library-oriented treatment. Content-based analysis and indexing is an important research domain within digital libraries to provide additional access paradigms to documents besides access based on annotated meta-data [Lyn09].

As a motivating example, consider the many scientific disciplines relying on empirical data, e.g., earth observation, experimental physics, medical and biological science, economics and the social sciences. In these disciplines, vast amounts of research data are produced or gathered on a daily basis. Often being funded by the public, demand for *open access* to the produced data is increasing. Making research data publicly available has several benefits. First, *reproducibility* and transparency of obtained results is a principal requirement for good scientific practice and publishing. Second, finding data *related* to one's own work is crucial for many researchers. Often though, research data is provided on an individual basis, with researchers putting up undocumented data, in an arbitrary format on personal web-space. Such data is usually available only for a limited time. Therefore, such practice hardly supports the demand for reproducibility, let alone the possibility to find related data. Hence, a need for *library-oriented* handling of research data exists [FMM*06].

Repositories and data libraries collecting research data from different domains include generic data underlying natural sciences publications [Dry], geoscientific and environmental data [PAN], psychological data [Psy], or biological information [ELI] and highly motivate research to increase data-accessibility. DataONE with MercuryONE [CMV*12] is a another recent example of a digital data library for geo-spatial data.

In a recent publication, Marcial et al. conducted a survey on available research data repositories with respect to scientific area, accessibility, size, business model, and many more aspects [MH10]. Out of 100 repositories they looked at in detail, 60 classified themselves as containing a large amount of data. With 26, the most repositories archived data in the area of earth observation / geo-sciences. Figure 2.8 provides an overview of the scientific areas.

The aim of these data libraries is the long-term availability of data, while adhering to specific formatting and documentation requirements. As such, this treatment of research data allows for reproducibility by supplying data associated with scientific publications as well as finding related data by searching for related textual publications. Well-established database techniques and thorough data curation, to guarantee format-adherence and meaningful metadata annotations, allow digital libraries to provide research data in such a way [GWCS09]. However, research data typically contains large quantities of non-textual, digital data content for which no native system support beyond *annotation-based* access is provided. As mentioned above, in the multimedia retrieval context, to date several systems exists that support content-based search relying on automatically extracted descriptors. However, devising meaningful retrieval methods for research data is a difficult problem.

Recent examples of research for meta-data based retrieval in data libraries include automatic tag recommendations [TPG13] to improve retrieval performance. The idea is to learn tags from a completely annotated training data-set and then propagate these terms to non-annotated (or partially annotated) documents based on the similarities between their textual content. As such, these approaches do not consider the data content itself.

Examples of recent digital library systems that provide different means of content-based access include systems for 3D models and classical music [BBC*10], images [RBPK12], time-series data [BBF*10], climate data [SBS11] and chemical data [KTB12]. On top of access via annotated meta-data, these digital library systems extract domain-specific *descriptors* from the underlying data as a basis to implement distance functions in support of search and access functionality. Such access includes query-by-example, e.g., supplying an example image and retrieving similar images [RBPK12, DJLW08]; query-by-sketch, e.g., drawing a shape and retrieving similar 3D models; or content-based layouts, e.g., clustering time-series by data similarity and presenting the user with an overview [BRS*12a].

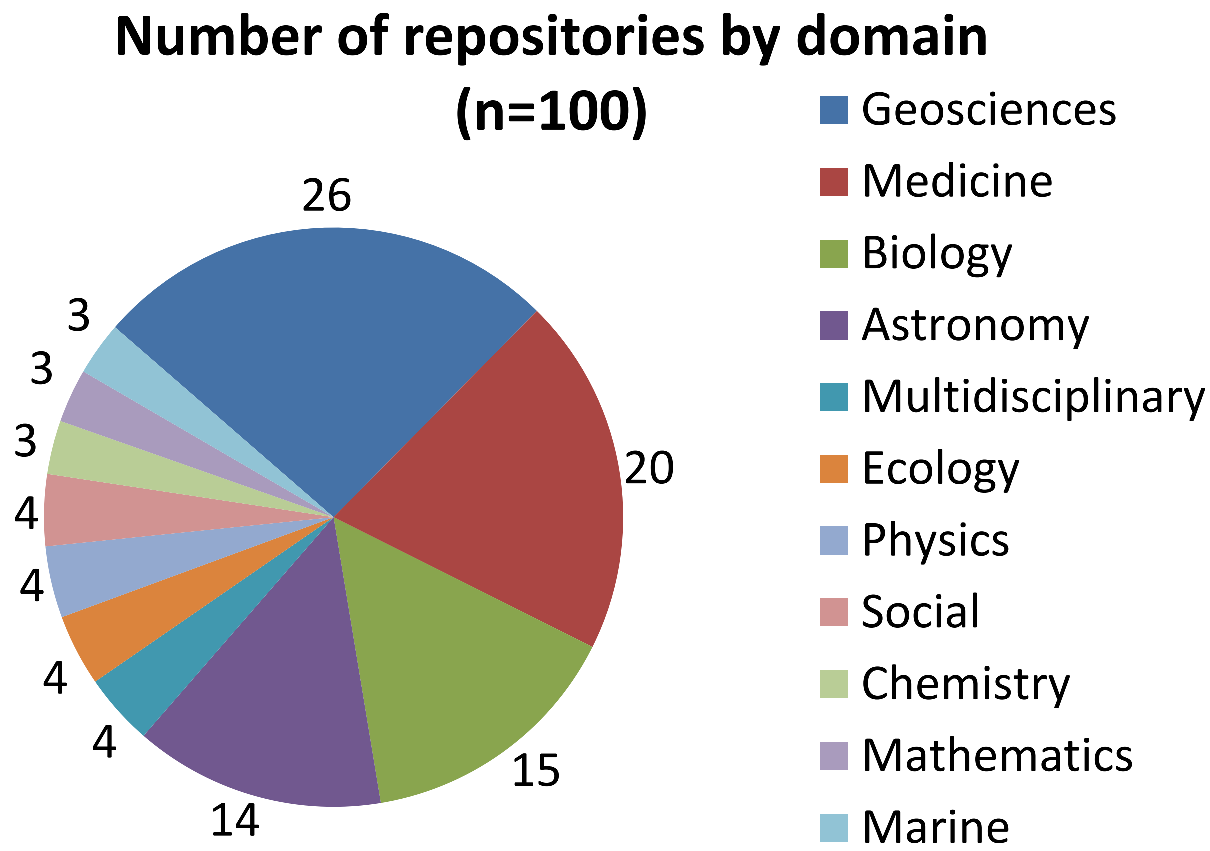


Figure 2.8: Number of research data repositories by their respective scientific domain (surveyed in 2010) [MH10].

2.9.1 Earth Observation Repositories

Though multivariate research data is obtained and archived in many disciplines, a major area of research where the long-term availability and accessibility to such data is of paramount importance is earth observation. Figure 2.8 shows an overview of a sample of categorized research data repositories. Most repositories exist for the geo-sciences and make up for 26% of data repositories. Therefor I chose to apply and evaluate my information retrieval approach in this domain. Accordingly, I want to provide a concise overview of the wide range of topics covered in earth observation and outline the state-of-the-art concerning the supply of information there.

According to recent reports of GEO[2] [GEOa], investments by governments and organizations in environmental monitoring and forecasting systems have reached a critical mass. This results in large, still expanding, global arrays of measurement and observation systems [SoEO10]. Common instruments

[2] Group on Earth Observations

used by these systems include meteorological stations and balloons, seismic and global positioning system stations, ocean buoys, remote-sensing satellites and computerized forecasting models and early warning systems. Individually, each instrument provides a specific point-of-view on some aspects of the earth system, whether it monitors the earth from space, the atmosphere, the oceans or the land. However when combined, these systems have the capability to offer a comprehensive picture of the entire planet.

Many of these observation devices are arranged in regional and national arrays, though more and more are connected to global earth observation networks through international partnerships. A major goal of GEO is to further increase data sharing, long-term availability, re-use and interoperability by building a Global Earth Observation System of Systems (GEOSS) [GEOb] to link all the existing earth observation networks.

These advances and aspirations in the area of earth observation show the vast amount of data that is already available, and will be available for researchers and decision makers in the future. Kumar provides several data mining techniques to analyze large collection of earth observation data for knowledge discovery [Kum10]. Efficient information retrieval systems are needed to provide access to these huge data repositories, such that information seekers are able to find what they are looking for. Recent advances that are applicable here include automatic tagging of meta-data using topic modeling [TPG13,TPN*12], to enhance meta-data based access.

2.10 Summary

Summarizing this chapter, I provided an overview of related, previous work. In particular, I described the state-of-the-art in textual retrieval, multimedia retrieval (with a focus on content-based image retrieval) and time-series retrieval, as several approaches in these retrieval domains motivated my approach for retrieval of multivariate data, which I will describe in the following chapter.

3 Approach to Multivariate Research Data Retrieval

In this chapter I am going to describe my approach for retrieving multivariate research data documents. Currently, repositories providing multivariate research data offer annotation-based access to retrieve documents. This means an information seeker can search over a set of meta-data annotations, such as author, title, year or location (among potentially many more), to retrieve documents of interest. My approach extends upon this by enabling content-based access, which allows an information seeker to search for particular patterns occurring in the data. As multivariate data is often analyzed by visualizing it as a scatter-plot-matrix, my motivation is to define a ranking and retrieval algorithms that are based on features that work in a similar vein.

3.1 Multivariate Data

A multivariate research data document consists of tabular data with n columns and m rows (observations) along with annotated meta-data for each column (for example parameter name and base unit, e.g. *water depth [m]*). Though not required for this approach, such data is usually annotated with further, descriptive meta-data like author, tile, year of publication and the geo-location in the case of geo-spatial data. A real-world example of such a multivariate research data document from the area of climate research is appended to this thesis in Appendix B.

Given a collection (or repository) of such multivariate research data documents, a major challenge is retrieving those documents that are of interest to an information seeker. In repositories that are in productive use today, this retrieval challenge is mostly met with textual search tools for the annotated meta-data. As noted in the introduction, several information needs cannot be (suitably) met with these kinds of retrieval tools. Content-based methods that render the actual patterns of the underlying data accessible are thus required.

My approach extends upon annotation-based access by providing additional content-based access. Motivated by the widely-used scatter-plot-matrix by human analysts, I extract all bivariate patterns from each multivariate document and index each document with the set of its patterns using a bag-of-words approach. This process is explained in detail in the subsequent sections. An evaluation of the resulting retrieval performance can be found in Chapter 4 Evaluation of Retrieval Performance .

3.2 Indexing Scheme

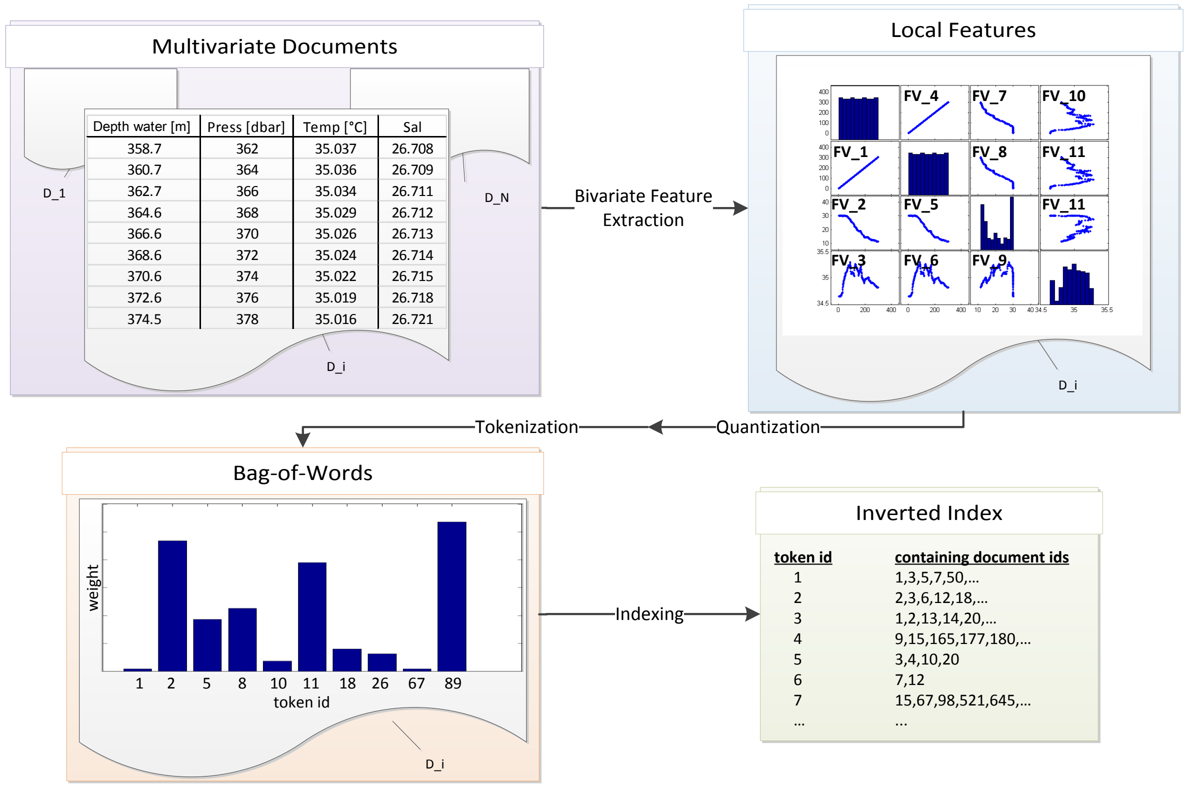
My goal is to build a content-based index for multivariate data on top of an annotation-based (textual) index. This content-based index should allow an information seeker to search for actual data patterns to retrieve data-sets of interest.

An established and well studied approach for an information seeker to visually analyze multivariate data is to inspect the scatter-plot matrix [CM84]. The scatter-plot matrix is an information visualization technique that is rendered by plotting the data-points of each column versus one another. Thus the scatter-plot matrix contains $n(n-1)$ scatter-plots of bivariate data, that as a whole, describe the multivariate data. I use this standard visualization approach as the basic idea behind my indexing approach for multivariate data. As this visualization technique is suitable to allow the human observer to compare different sets of multivariate data, information extracted in a similar vein should be suitable for retrieval.

The goal is to extract a set of mathematical descriptors from multivariate data that model each of the $n(n-1)$ bivariate data patterns visualized in the scatter-plot matrix. That way, we obtain a set of local features for each document. We then quantize each feature vector, by assigning the id of the closest cluster centroid (obtained, e.g., via k-means clustering) to each feature vector. Thus we can represent a multivariate data document by the set of content-based tokens obtained from this quantization. Such a representation is similar the way textual documents are modeled for indexing (namely by their term vectors) and allows us to leverage efficient indexing methods known from textual retrieval such as inverted lists. A concise visualization of these steps is shown in Figure 3.1; each step is explained in detail in Section 3.4.

To index the bag-of-words representation of each document we build an inverted list for each distinct term. This means, that I compute and store a hashed look-up from each term to each document that contains this term along with an associated weight. This allows for ranked retrieval by intersecting the inverted lists of each search term, aggregating the term weights and sorting them in descending order. This approach scales very well. The required main memory for this indexing structures increases linearly with the number of indexed documents. Retrieval time is constant with respect to the index look-up and is dominated by the time required to read the document data from the hard disks.

Furthermore, I want to build an index that allows the retrieval of multivariate data documents that are similar to a given example object. Within the database, this can be used to explore the data-sets using nearest-neighbor retrieval or clustering. For querying, this allows an information seeker to provide an example data-set to retrieve the most similar ones. To this end, I develop a topic modeling approach that extracts a feature vector from each document's bag-of-words representation containing its topic activations. Using approximate indexing of high-dimensional feature spaces, one can efficiently index the resulting feature vectors for fast nearest-neighbor retrieval.



Depth water [m]	Press [dbar]	Temp [°C]	Sal
358.7	362	35.037	26.708
360.7	364	35.036	26.709
362.7	366	35.034	26.711
364.6	368	35.029	26.712
366.6	370	35.026	26.713
368.6	372	35.024	26.714
370.6	374	35.022	26.715
372.6	376	35.019	26.718
374.5	378	35.016	26.721

token id	containing document ids
1	1,3,5,7,50,...
2	2,3,6,12,18,...
3	1,2,13,14,20,...
4	9,15,165,177,180,...
5	3,4,10,20
6	7,12
7	15,67,98,521,645,...
...	...

Figure 3.1: Indexing scheme for multivariate research data repositories. From each multivariate data document, extract all bivariate features, quantize and weight those features to obtain a bag-of-words representation and finally index this representation using inverted lists.

3.3 Bivariate Feature Extraction

The first goal for my proposed approach for multivariate research data retrieval is to extract bivariate features from the multivariate data-sets we want to index. Feature extraction is the process of computing a descriptor, that mathematically represents one or several properties of an object under consideration. Such a descriptor allows to assess pairwise similarity between objects, by computing a distance measure between their respective descriptors. A prominent type of descriptors are feature vectors. As the name implies, the idea is to capture descriptive and discriminative object features as a vector of numerical values.

For the purpose of this section, we will look at bivariate feature extraction as a sub-problem of multivariate feature extraction. It is reasonable to expect that, if we extract features from bivariate data that describe that data well, a set of such features will describe multivariate data well.

I propose nine techniques which I developed for and/or adapted to feature extraction of bivariate data, and I will benchmark these techniques with respect to their bivariate data retrieval performance [SvLS12]. The nine techniques are based on time-series analysis, regression and image processing, respectively and are detailed below.

3.3.1 Techniques

In this subsection I provide a technical overview of nine techniques to extract feature vectors from bivariate, numerical data. The first technique (3.3.1.1) can be considered as a straw-man or baseline technique, that resamples data to allow for measuring Euclidean distance between data objects. The second technique (3.3.1.2) is another baseline technique that is based on correlation coefficients. The next two techniques (see 3.3.1.3 and 3.3.1.4) are based on parametric and non-parametric regression respectively, and I propose two straight-forward algorithms to extract features from the resulting regression coefficients. Three of the techniques (see 3.3.1.5, 3.3.1.6 and 3.3.1.7) are well established methods to describe univariate, sequential data (i.e. time-series) and are thus readily applied to bivariate data by sorting the data-points along one dimension. Finally, I adapt a kernel-density-based technique (see 3.3.1.8), as well as a shape based technique from image processing (see 3.3.1.9) to bivariate feature extraction.

Raw, bivariate data is usually neither directly applicable for meaningful similarity assessment nor feature extraction because of different dimensionality, missing values, outliers, etc. Although amending (most of) these aspects is necessary, it arguably introduces bias to the data. I try to make this bias transparent for our evaluation, by considering a minimal pre-processing scheme. I remove rows with missing values and normalize the data to zero-mean and a standard deviation of one (z-normalization).

3.3.1.1 Resampling

As a baseline for retrieval, finding nearest neighbors for a query can be conducted by computing a distance directly on the complete input data. Any meaningful feature extraction should perform at least as well as such a strawman technique. For retrieval in bivariate data though, computing a distance between data objects is not straight-forward due to different dimensionality of the data. Instead, I consider interpolating and resampling every data object to achieve matching dimensionality. To this end a smoothing spline [Cle85] is fitted to the data and I uniformly resample 100 data points from that fit. I compute the distance between these 100 data points as a baseline for similarity measurement between two data objects. I denote this technique as `SM` for *strawman*.

3.3.1.2 Correlation Features

Correlation coefficients are a statistical measure to indicate how strong a linear relationship exists between two variables. For bivariate data, it can be assumed that two data objects measuring the relationship between the same two variables should have similar correlation coefficients. Hence, I consider correlation coefficients as a second baseline technique for similarity measurement between two data objects.

To extract a feature vector for retrieval, I compute Pearson's sample correlation coefficient r, Kendall's tau rank correlation coefficient τ and Spearman's rank correlation coefficient ρ for each bivariate data object. Denoted as $\vec{v}_{\text{corr}}$, this feature vector is accordingly computed as

$$
\begin{aligned}
r &= \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}} && (3.1)\\
\tau &= \frac{n_c-n_d}{\frac{1}{2}n(n-1)} && (3.2)\\
\rho &= \frac{\sum_i(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_i(x_i-\bar{x})^2\sum_i(y_i-\bar{y})^2}} && (3.3)\\
\vec{v}_{\text{corr}} &= (r,\tau,\rho)^T && (3.4)
\end{aligned}
$$

3.3.1.3 Regressional Features

I developed a feature extraction method called 'regressional features' which is based on nonlinear, parametric regression.

The algorithm to compute these features is outlined in Figure 3.2. The basic idea is to generalize the notion of correlation, by fitting the data to a number of representative functional models using nonlinear parametric regression. Computing the relative goodness-of-fit (GOF) to each of these models and storing these parameters in a vector, yields a descriptor of the functional form of the data. I therefore denote this descriptor as *regressional feature vector* $\vec{v}_{\text{rf}}$, abbreviated as *RF*. The functional models I use are included in the figure and their respective functional form is visualized as a colored plot. I chose these models with complementarity and completeness in mind, such that at least one of the models should be suitable to describe any kind of functional relationship in the data, while not capturing any functional properties the other models would be able to. Please note that this approach may also easily be extended by further functional models as possibly required by specific application domains.

In the lower left of Figure 3.2 we see $\vec{v}_{\text{rf}}$ computed for some exemplary data. It is visualized as a colored histogram, since each entry of the vector relates to the probability of the correspondingly colored functional model being applicable.

Algorithmically, Matlab's `nlinfit` is used for nonlinear regression, which employs a Levenberg-Marquardt minimization. This iterative algorithm is quite expensive in terms of time required until convergence and only a local optimal fit can be guaranteed.

The resulting feature vector however is of low dimensionality (42×1, the number of models plus number of coefficients in each model) and allows for fast distance computation between data-sets.

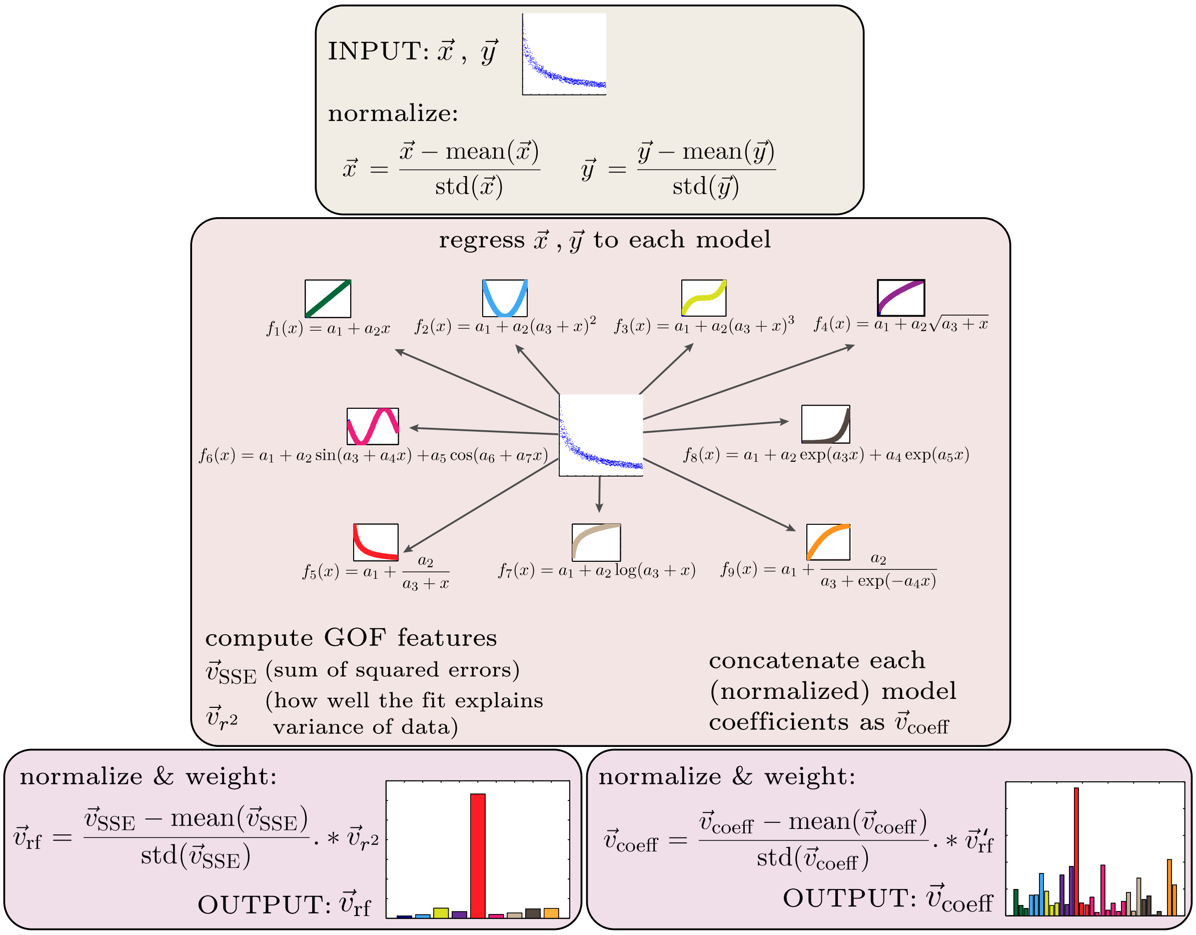

INPUT: $\vec{x}$, $\vec{y}$
normalize:
$\vec{x} = \frac{\vec{x} - \text{mean}(\vec{x})}{\text{std}(\vec{x})}$ $\vec{y} = \frac{\vec{y} - \text{mean}(\vec{y})}{\text{std}(\vec{y})}$
regress $\vec{x}$, $\vec{y}$ to each model
$f_1(x) = a_1 + a_2 x$
$f_2(x) = a_1 + a_2(a_3 + x)^2$
$f_3(x) = a_1 + a_2(a_3 + x)^3$
$f_4(x) = a_1 + a_2\sqrt{a_3 + x}$
$f_6(x) = a_1 + a_2 \sin(a_3 + a_4 x) + a_5 \cos(a_6 + a_7 x)$
$f_8(x) = a_1 + a_2 \exp(a_3 x) + a_4 \exp(a_5 x)$
$f_5(x) = a_1 + \frac{a_2}{a_3 + x}$
$f_7(x) = a_1 + a_2 \log(a_3 + x)$
$f_9(x) = a_1 + \frac{a_2}{a_3 + \exp(-a_4 x)}$
compute GOF features
$\vec{v}_{\text{SSE}}$ (sum of squared errors)
$\vec{v}_{r^2}$ (how well the fit explains variance of data)
concatenate each (normalized) model coefficients as $\vec{v}_{\text{coeff}}$
normalize & weight:
$\vec{v}_{\text{rf}} = \frac{\vec{v}_{\text{SSE}} - \text{mean}(\vec{v}_{\text{SSE}})}{\text{std}(\vec{v}_{\text{SSE}})} .* \vec{v}_{r^2}$
OUTPUT: $\vec{v}_{\text{rf}}$
normalize & weight:
$\vec{v}_{\text{coeff}} = \frac{\vec{v}_{\text{coeff}} - \text{mean}(\vec{v}_{\text{coeff}})}{\text{std}(\vec{v}_{\text{coeff}})} .* \vec{v}'_{\text{rf}}$
OUTPUT: $\vec{v}_{\text{coeff}}$

Figure 3.2: Algorithm to compute regressional features of bivariate data. Relative goodness-of-fit parameters and the actual coefficients obtained by nonlinear regression to each specified model form the extracted feature vector [SBS11].

3.3.1.4 Smoothing Splines

In contrast to parametric regression used for the previous feature extractor, smoothing splines is a non-parametric technique concerned with estimating the functional relationship underlying bivariate data directly. Instead of using parametric models to constrain this relationship to a specific one, smoothing splines employ a regularizer to enforce smoothness of the resulting spline function. I use the implementation in Matlab's Curve Fitting Toolbox, which is based on the original proposal by Reinsch [Rei67]. For further details on smoothing splines, please refer to Silverman [Sil85] and to the book *Functional Data Analysis* by Ramsay and Silverman [RS05].

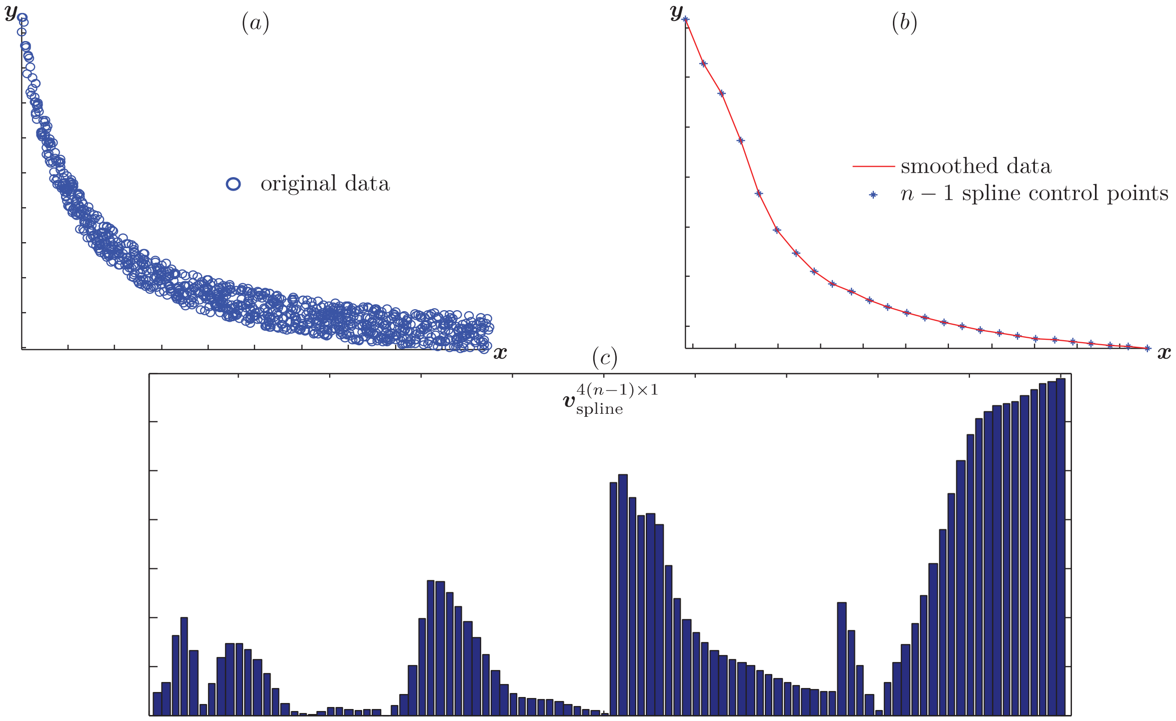


Figure 3.3: Feature extraction from bivariate data using smoothing spline coefficients. Original data (a) is preprocessed by smoothing and uniformly resampling n points. Then a another smoothing spline is fitted to these points (b). The feature vector (c) is composed of four (normalized) coefficients for each spline control point.

The algorithm to extract a feature representation from bivariate data using smoothing splines works as illustrated in Figure 3.3. First, one fits a smoothing spline to the original (preprocessed) data-points. A smoothing spline has a smoothing parameter ρ, which determines the smoothness of the spline by allowing the default cubic spline to deviate from the data-points in order to achieve smoothness. So in the extreme case of $\rho = 1$, the smoothing spline defaults to the cubic spline interpolant and goes through every data-point. In the other extreme of $\rho = 0$, the smoothing spline degenerates to linear regression and produces a least-squares fit of a straight line to the data-points.

After fitting the smoothing spline, one can uniformly resample n new data-points on that spline. By fitting another spline (without any further smoothing, i.e. $\rho = 1$) to these n resampled points, one obtains $4 \cdot (n-1)$ coefficients of this spline in pp-form that are then used as a descriptor for the bivariate data pattern. For each point on the spline, I normalize the four coefficients to sum up to one. Then all normalized coefficients are concatenated into a single vector, which is then normalized, such that it sums up to one.

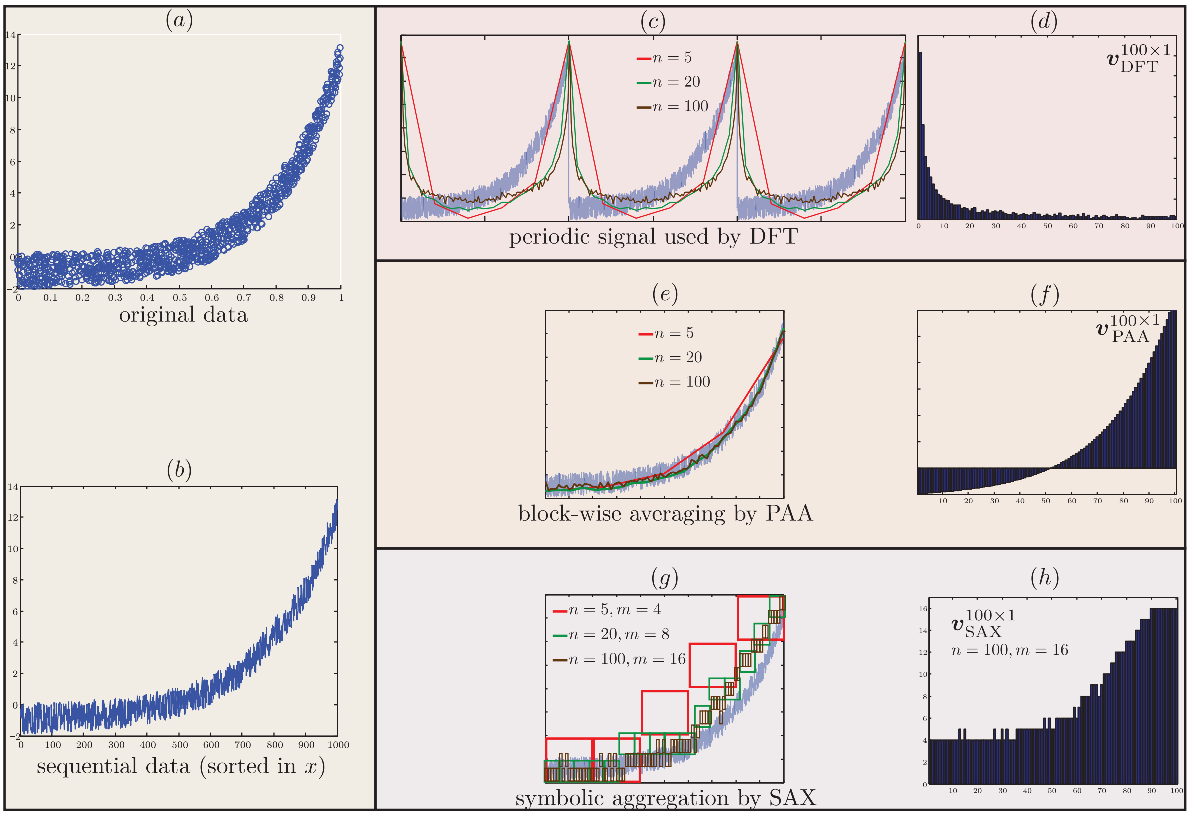


Figure 3.4: Feature extractors for sequential data, applied to bivariate data. Original data (a) is sorted along one dimension (b). By extracting Fourier features (d), the sequential data is interpreted as a periodic signal (c). Also shown in (c) are the signal reconstructions using $n = \{5, 20, 100\}$ Fourier coefficients. The reconstructions for the PAA descriptor (f) are shown in (e) for a different number of aggregation blocks. The SAX descriptor (h) is also reconstructed for different aggregation sizes in both dimensions (g).

3.3.1.5 Discrete Fourier Transform

Any signal can be transformed into an equivalent representation consisting of the weighted sums of sine and cosine waves. Transforming a signal from time to frequency domain is called Fourier Transform, and can also be applied to discrete sequences using the Discrete Fourier Transform (DFT):

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}kn}, \qquad k = 0, \dots, N-1 \tag{3.5}$$

An efficient implementation called *Fast Fourier Transform* (FFT) transforms a signal with N points in only $O(N \cdot \log(N))$ operations. As most of the energy of a signal is in the lower frequencies, a compact signal representation can be obtained by selecting only the first $k = 0, \ldots, M$ Fourier coefficients as a descriptor for the signal. Since computation is fast and this truncated Fourier representation is compact, it is particularly suited for very large datasets as shown before [AFS93].

To describe bivariate data (which, in contrast to time-series is not sequential) with the DFT feature extraction, a straight forward way is to sort the bivariate data along either dimension, which yields sequential data. This data is interpreted by the discrete Fourier transform as a periodic signal. For transformation, the FFT algorithm is used and the first M Fourier coefficients form the feature vector extracted from the bivariate data. Figure 3.4c illustrates this.

3.3.1.6 Piecewise Aggregate Approximation

A simple, yet very powerful technique to describe sequential data is the so-called Piecewise Aggregate Approximation (PAA) [YF00, KCPM01]. The basic idea is to split a sequence of length n into m segments and compute the mean value of all data-points in each segment. Such a block-wise average can be computed extremely fast. Only n (sequence length) additions and m (number of segments) divisions are required.

Applicability of this feature extraction algorithm to bivariate data is straight-forward. One can define blocks by sorting bivariate data along one dimension and aggregating all points in these blocks accordingly. In Figure 3.4e we see an example of this approximation.

3.3.1.7 Symbolic Aggregate approXimation

In 2005, Keogh et al. proposed a descriptor for time-series based on symbolic representation [KLF05]. This *Symbolic Aggregate approXimation* (SAX) quantifies the time domain *and* the value domain of time-series data into discrete representations. This is very robust against small changes in the data and complexity of the descriptor can be chosen by specifying the number of symbols to use for the value domain quantization and number of bins to use for the time domain quantization.

Keogh et al. also showed a very advantageous property of SAX. The distance of this descriptor for two time-series is a lower bound for the Euclidean distance of these two time-series.

Computing such a symbolic representation for bivariate data works analogous to the two previously described techniques. By sorting the data along one dimension we can quantify both axis and compute the descriptor. Figure 3.4g and 3.4h illustrate this.

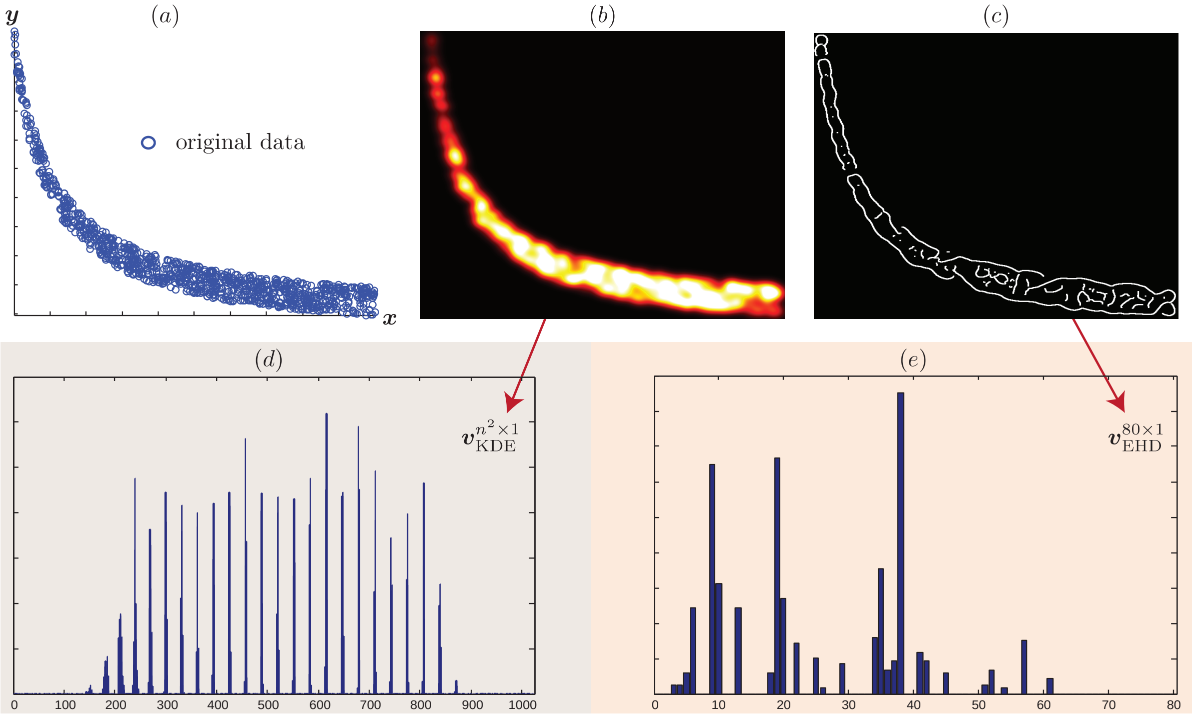


Figure 3.5: Original data (a) is used as input to estimate a 2d kernel density map (b) with resolution $n \times n$. Encoded as a $n^2 \times 1$ column vector, this serves as the kernel density descriptor `KDE` (d). After detecting the edges on the kernel density map in (c), we compute the edge-orientation histogram descriptor `EHD` in (e).

3.3.1.8 Kernel Density Estimate

Estimating the probability density function underlying a set of sampled data is a well studied field, and finds application in many areas. Kernel density estimation constrains the kernel function of the probability density to a specific form, often a Gaussian kernel is assumed.

Recently in [BGK10], Botev et al. propose a method for estimating the kernel density of two-dimensional data using a Gaussian kernel. Their contribution is the optimal choice of the two bandwidth parameters for this Gaussian kernel, without assuming a parametric model for the sample data. For further details please refer to their paper.

I use the Matlab implementation `kde2d` provided by Botev et al. for their kernel density estimation method. As additional output, this function returns a discrete, two-dimensional probability density map (in the specified resolution $N \times N$) based on the estimation. This probability density map (encoded as a $N^2 \times 1$ column vector) serves as our density feature vector for bivariate data. The left part of Figure 3.5b shows an example.

3.3.1.9 Edge Histogram Descriptor

One prominent approach in image processing to describe the shapes seen in an image, is to look at the distribution of the orientation of *edges* in that image. A brief overview of such algorithms, applied to sketch-based image retrieval, is given by Eitz et al. [EHBA10]

I chose to use one of these algorithms, the Edge Histogram Descriptor (EHD) [PJW00], which is included in the MPEG-7 standard. The basic idea is to partition an image into four equally sized, rectangular regions and compute the distribution of edge-orientation in each rectangle as a histogram with 20 bins.

To extract features from bivariate data, I first use the aforementioned kernel density estimator to produce a 128×128 probability density map of the data. I simply regard this density map as a gray-scale image, and convert it to a binary edge map using the *Canny*-edge detector algorithm. This binary edge map serves as input for the EHD algorithm, and thus we obtain an 80-dimensional descriptor that captures the *shape* of the bivariate data distribution. Figure 3.5c illustrates an example.

3.3.2 Summary

So far, I presented nine techniques to extract a feature vector from bivariate data. These approaches span a wide range of processing schemes, from regression, to aggregation and density estimation combined with edge description. As multivariate data with n columns contains $n(n-1)$ bivariate patterns, all of the proposed feature extraction techniques can be used to retrieve multivariate data documents that contain a particular bivariate pattern.

Before evaluating which of these techniques provides the best retrieval performance (see Section 4.2), I will look into building an efficient index of bivariate patterns for multivariate data and extracting a feature vector that describes multivariate data documents as a whole.

3.4 Multivariate Feature Extraction

Multivariate research data documents consist of tabular data with n columns (measurement variables / parameters) and m rows (observations) along with annotated meta-data for each column (usually parameter name and base unit, e.g. *water depth [m]*).

Recall that there are two goals for the proposed content-based access to multivariate data. The first goal is to build a content-based index, such that one can retrieve multivariate data by the bivariate patterns it contains. The second goal is to compute a similarity score between two multivariate documents such that one can do nearest neighbor retrieval.

My approach to achieve the first goal, is to obtain a set of local, bivariate features for each document and index those features using a bag-of-words representation. Therefore, one quantizes each feature vector by assigning the id of the closest cluster centroid (obtained, e.g., via k-means clustering) to each feature vector. This is similar to the way textual documents are represented for indexing (namely by their term vectors) and allows us to leverage efficient indexing methods known from textual retrieval such as inverted lists.

Furthermore, to achieve the second goal of retrieving multivariate data documents that are similar to a given example object, I develop a topic modeling approach. The idea is to analyze which terms from each document's bag-of-words representation co-occur in different documents in a given corpus. This allows us to generate a probabilistic model of topics, from which the documents might have been created. Inferring the topic activations of each document yields a compact feature representation of all the bivariate patterns this document contains. Using approximate indexing of high-dimensional feature spaces, one can efficiently index the resulting feature vectors for fast nearest-neighbor retrieval.

With this motivation and overview in mind, I will now explain how to obtain the bag-of-words representation and how to compute a topic model based thereupon.

3.4.1 Bag-of-Words Representation

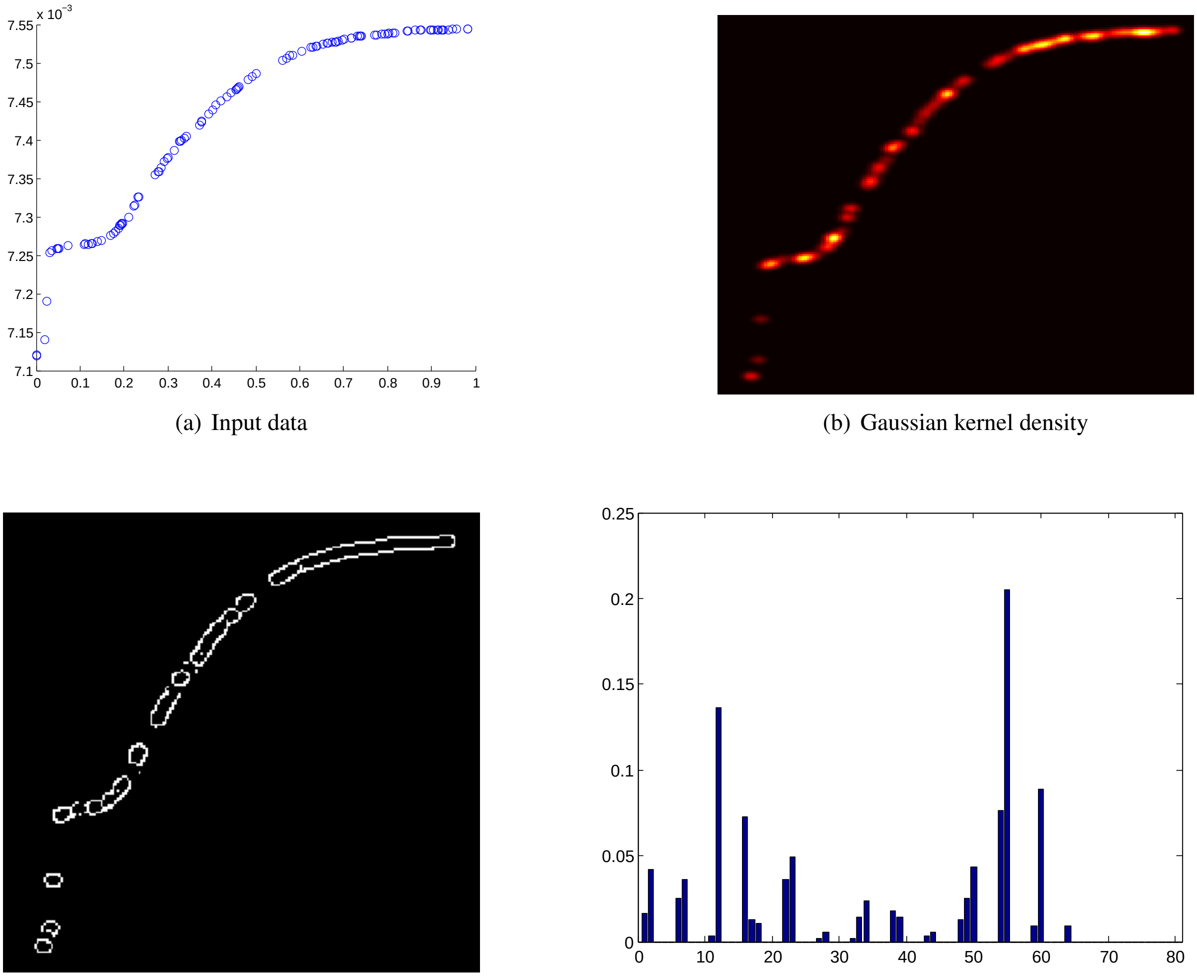
As the first step of my approach I want to represent each multivariate document with a bag-of-words descriptor. This approach originates from text processing, where a given textual document can be represented with the histogram (thus, word-counts) of all the words it contains. I apply this approach to multivariate data using the following steps.

Step 1: Feature Extraction To extract a set of feature vectors from a document containing multivariate data, I propose to compute all bivariate variable combinations, and compute a feature vector from each of these two-dimensional point-clouds (scatter-plots). These feature vectors are *local* in the sense that each represents a local pattern (bivariate) in the whole (multivariate) document.

Based on my previous results on feature extraction and benchmarking for bivariate data [SvLS12] (see previous section 3.3 Bivariate Feature Extraction), I will use the algorithm that yielded the best overall results in the benchmark. Please refer to Chapter 4 for the evaluation results.

The best performing bivariate feature extraction algorithm is based on the MPEG-7 descriptor "edge histogram detector" (EHD) widely used in image and shape retrieval. The basic idea of EDH is to first estimate the density of the bivariate data using a Gaussian kernel [BGK10]. This density estimate is then rendered as an actual scatter-plot of the bivariate data. During this process the scatter-plots is min-/max-normalized, resulting in translation and linear-trend invariance. Then the full canny edge filter pipeline is applied to this rendered image. The edge image is then partitioned into uniform regions and the orientation of the edges in each region are extracted as a histogram. Figure 3.6 shows an illustration of this extraction process.

The result of this feature extraction step is a set of 80-dimensional feature vectors for each multivariate document. The number of feature vectors equals $n \cdot (n-1)$, the number of possible scatter-plots for multivariate data with n dimensions.



(a) Input data

(b) Gaussian kernel density

(c) Detected edges

(d) Edge histogram

Figure 3.6: Bivariate feature extraction: Given bivariate input data (a), estimate the Gaussian kernel density (b), apply a Canny edge detector (c) and compute the edge histogram descriptor (d). This algorithm has shown to yield state of the art performance in my previous work for bivariate data retrieval [SvLS12].

Step 2: Quantization The result of the feature extraction step is a set of feature vectors for each document. Since we need to obtain a set of tokens for each document, I train a quantization model that is suitable to project each input feature vector to a categorical integer value. There is a wealth of clustering algorithms suitable for this task [XW*05]. I chose k-means clustering as this has shown good performance for image-retrieval tasks at reasonable performance costs. I compute a k-means

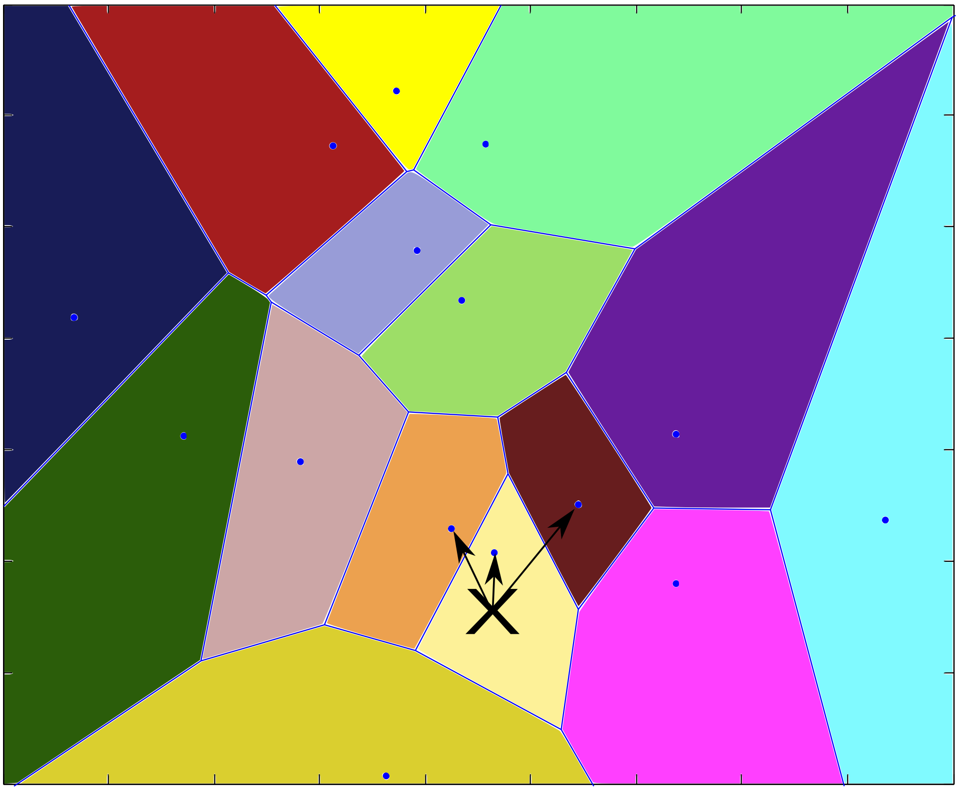


Figure 3.7: Quantization of the feature space using k-means. A feature vector $\vec{x}$ is assigned to its three nearest neighbor centroids of the k-means clustering. The distance to each centroid is used as a weight for ranking.

clustering using all available feature vectors $\vec{v}_i$. The number of clusters k was set set to 1000 based on manual optimization by looking at intra-cluster and inter-cluster divergence. Depending on the domain and retrieval application at hand, k needs to be tuned accordingly (lower for more robustness and invariance, higher for better pattern discrimination).

One can then compute the nearest of the k centroids to a given feature vector and assign the ID of this centroid as the token for this feature vector. For a more robust and fuzzy bag-of-words, I assign each feature vector to its three nearest neighbor centroids, but omit this detail in the remaining description of the algorithm. See Figure 3.7 for an illustration of this quantization process.

Step 3: Tokenization Since we are not only interested in bivariate data patterns (which are encoded in the quantized feature vectors), but also in the variable combination that exhibits this pattern, I combine the feature tokens with the annotated column meta-data. For example, if the feature vector of the

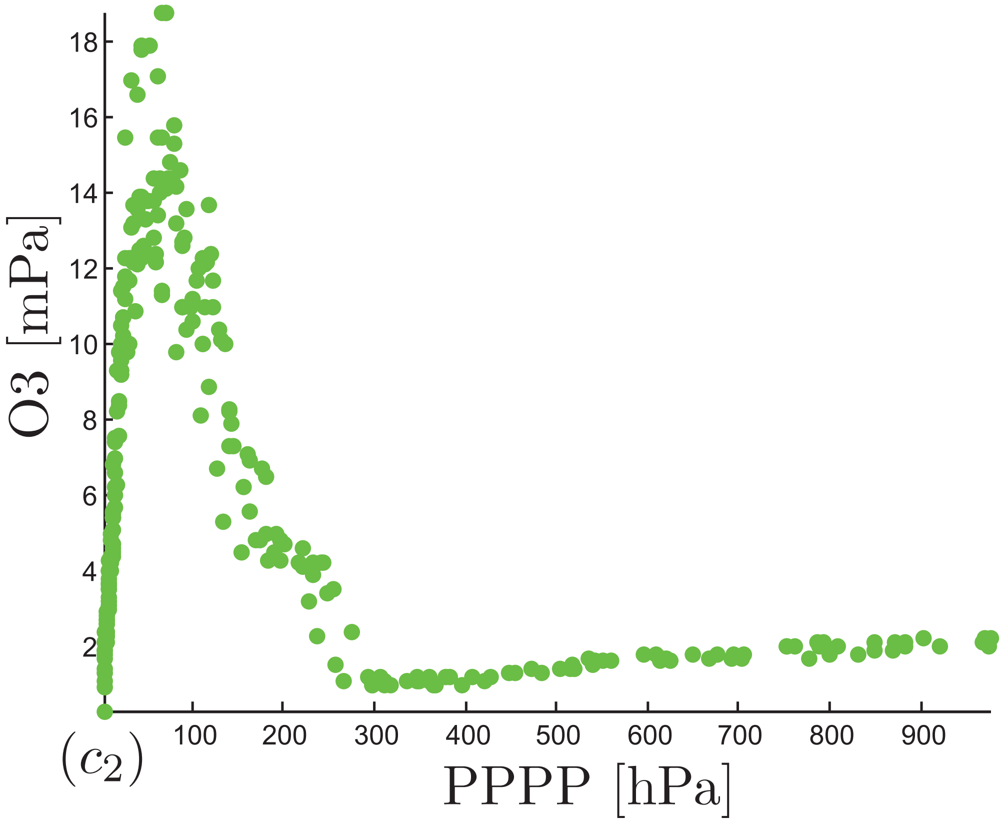


Figure 3.8: Feature Token: After extracting a feature vector from the data points and assigning them to the closest k-means centroid (c_2), we obtain the following feature tokens: *c2*, *PPPP_[hPa]*, *O3_[mPa]*, *PPPP_[mPa]_vs_O3_[mPa]* and *PPPP_[mPa]_vs_O3_[mPa]_c2*. The feature tokens can be efficiently indexed using inverted lists much like a regular search index for textual documents.

scatter-plot of column *a* versus column *b* was quantized to cluster id *c*, we would obtain the terms *a*, *b*, *c*, *a_b*, *a_b_c*. See Figure 3.8 for an actual example obtained in our test setup.

Step 4: Term Weighting After obtaining a set of terms for each document, we have to choose a weighting scheme for these terms to account for their respective relevance to the containing document for the topic modeling process. A straight forward scheme to measure the relevance of a term to a given document is *term frequency* – the number of occurrences of a term in a document. I use this approach for the terms obtained from annotated information (*a*,*b* and *a_b* above). For terms that include the

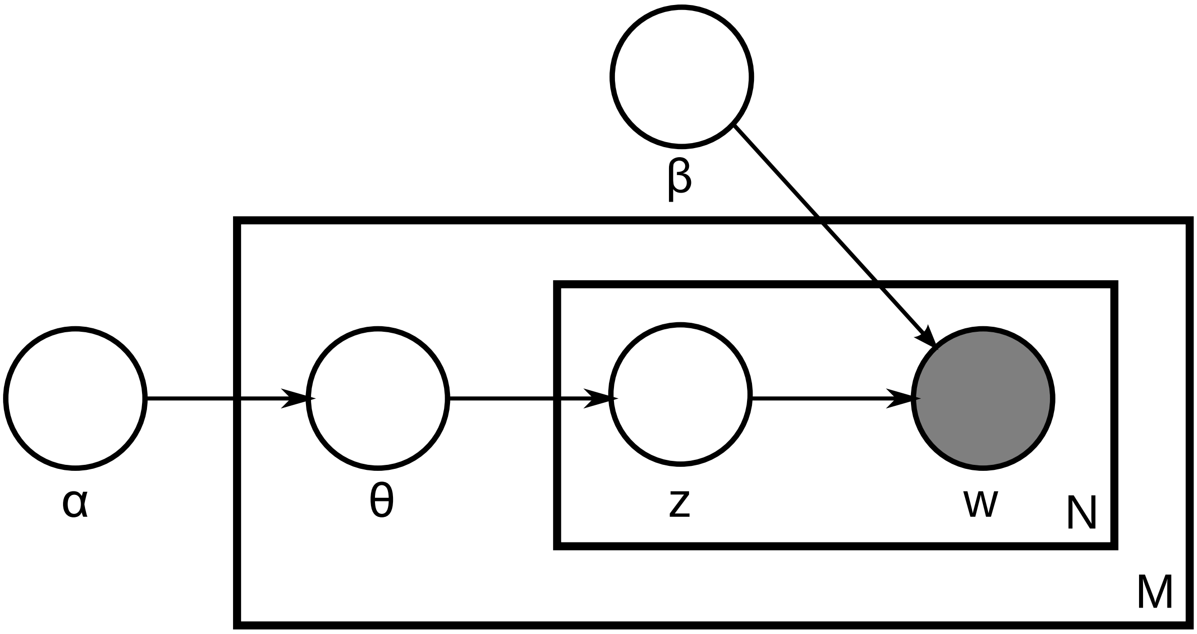


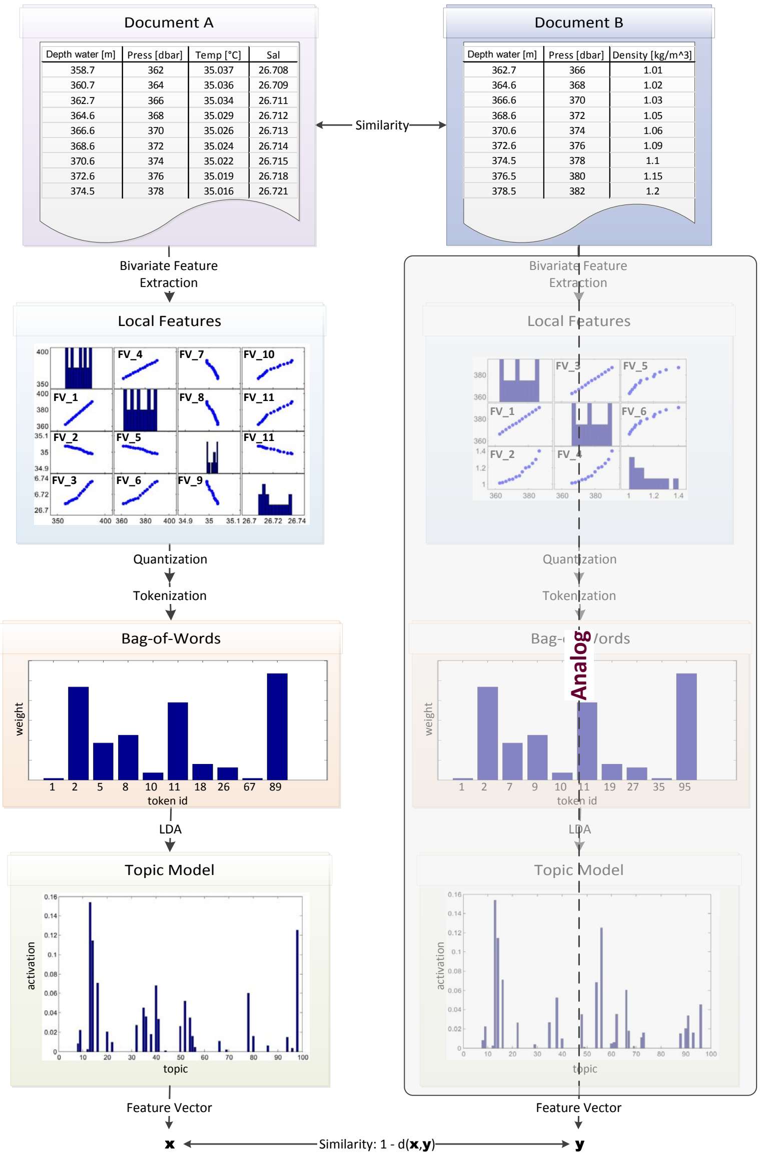
Figure 3.9: Latent Dirichlet Allocation in plate notation. Using my bag-of-words representation, we can observe each token w_{ij}. Each of these tokens describe a bivariate pattern occurring in the underlying multivariate data. The latent topic assignment $z_i j$ of token j in document i and the latent topic distribution θ_i of document i are chosen such as to best explain the observed tokens. Original image by Bkkbrad, released under cc-by-sa 3.0 [Cre].

feature pattern, I propose to use the distance to the closest cluster centroid in feature space as the relevance of those terms respectively (*c* and *a_b_c* above).

3.4.2 Topic Modeling

The goal of topic modeling is to extract a number of latent topics for a collection of documents and represent each document as a mixture of those topics. Topics are modeled in such a way to best account for the token-document distribution observed in the document collection. This representation effectively describes the data patterns occurring in a given document and is thus very suitable for similarity measurement and subsequently for nearest-neighbor retrieval.

Topic modeling is a generative learning process that models documents as a mixture of a small number of topics. Latent-Dirichlet-Allocation (LDA) is a popular topic model proposed by Blei et al. in 2003 [BNJ03], which is also used in this work. Please refer to section 2.3 Topic Modeling for more related work. In the scope of this work, I transferred the basic idea of topic modeling for the first time to the domain of multivariate research data [SvLS13a].



Document A

Depth water [m]	Press [dbar]	Temp [°C]	Sal
358.7	362	35.037	26.708
360.7	364	35.036	26.709
362.7	366	35.034	26.711
364.6	368	35.029	26.712
366.6	370	35.026	26.713
368.6	372	35.024	26.714
370.6	374	35.022	26.715
372.6	376	35.019	26.718
374.5	378	35.016	26.721

Document B

Depth water [m]	Press [dbar]	Density [kg/m^3]
362.7	366	1.01
364.6	368	1.02
366.6	370	1.03
368.6	372	1.05
370.6	374	1.06
372.6	376	1.09
374.5	378	1.1
376.5	380	1.15
378.5	382	1.2

Figure 3.10: Approach for topic modeling of multivariate research data documents. Our goal is to compute a similarity measure between two multivariate documents A and B using the following steps. (a) Extract $n(n-1)$ local, bivariate feature vectors, where n is the number of columns in each document. (b) Quantize these feature vectors by assigning the closest nearest neighbors of a k-means clustering and thus obtain a bag-of-words representation. (c) Learn k latent topics from the observed document-token distribution using LDA and thus (d) represent each document as a mixture of those topics in vector form. (e) Assess similarity via a suitable distance measure; use cosine similarity.

I use the bag-of-words representation for the multivariate data documents under concern to compute a topic model. Using my own implementation of the Latent-Dirichlet-Analysis, and the (normalized and rescaled) word histograms from our bag-of-words representation as input, I compute a topic model consisting of k^* topics[1]. The generative process is shown in Figure 3.9 in plate notation. In plate notation, iterative processes are illustrated with a surrounding box (or plate) with the number of iterations shown in the lower right. We can observe the tokens each document consists of using the histogram of the bag-of-words representation. Iteratively, the latent assignment of each token to one of the k^* topics and the topic distribution of each document are optimized, to best explain the observed tokens.

Concerning parametrization of the topic modeling, I optimized the number of topics k^*, by manually inspecting the resulting topics of the trained topic model. I paid particular attention to different sets of patterns often co-occurring in a number of documents to be the most important tokens for different topics. Finally I set $k^* = 100$. Setting k^* too high, leads to over-fitting in the learning step and causes the topic mixtures to differ significantly for every document. Choosing k^* too low leads to non-discriminative topic mixtures.

Upon completion of this step, each multivariate data document can now be represented with a k^*-dimensional feature vector containing the topic activations for the document.

3.4.3 Similarity Function

After applying the feature extraction as described in the previous sections, we obtain a mixture of topic activations as the representation of each document. This is essentially a probability density function (or, a histogram, normalized, such that its entries sum up to one). To compute similarity between two documents using their respective topic activations, I propose to use the cosine similarity, though other metrics can be used as well.

Given two multivariate documents A and B in their respective topic model representations $\vec{x}$ and $\vec{y}$, we can then compute their similarity via

$$\begin{aligned} s(\vec{x},\vec{y}) &= \frac{\vec{x}\cdot\vec{y}}{||\vec{x}||\,||\vec{y}||} \\ &= \frac{\sum_{i=1}^{k^*} x_i y_i}{\sqrt{\sum_{i=1}^{k^*}(x_i)^2}\cdot\sqrt{\sum_{i=1}^{k^*}(x_i)^2}}, \end{aligned}$$

[1] conventionally the number of topics is denoted with k. I use k^* instead, as k is already used to denote the number of clusters of the k-means algorithm in the previous step

where k^* is the number of topics modeled in the step before. As previously mentioned in Section 2.1 Distance Functions , the cosine similarity has the advantage of producing an output in the $[0,1]$ interval for non-negative vectors (which the topic activation vectors $\vec{x}$ and $\vec{y}$ are).

3.5 Retrieval Scheme

In the previous sections of this chapter I proposed the feature extraction process for retrieval in multivariate research data. The idea was to extract a set of feature vectors from all bivariate patterns occurring in multivariate data and convert these feature vectors to a bag-of-words representation for further indexing and topic modeling (see Figure 3.10 for an illustration).

This information retrieval approach for multivariate research data enables three kinds of querying tasks, that are explained subsequently:

- Ranked retrieval for one or more bivariate data patterns occurring in multivariate data-sets
- Aggregation and ranking of bivariate data patterns for retrieval suggestions and refinements
- Nearest-Neighbor retrieval for multivariate data-sets using query-by-example based on approximate feature space indexing

3.5.1 Ranked Retrieval Based on BM25

For ranked retrieval using one or more bivariate data patterns, I propose to use a popular ranking function from text retrieval called 'Okapi BM25' [MRS08] to compute the relevance of a multivariate data-set to the query terms (see Section 2.5 Textual Document Retrieval for related work).

The following ranking function is based on the BM25 definition by Manning [MRS08], and was adapted to the bag-of-words approach proposed in this thesis.

Let Q denote a query consisting of query-terms $q_1,\ldots,q_m$. Let D_i be the ith out of n documents, for which we want to compute its relevance to the query. Each query term, as well as each term of the document belongs to the feature vocabulary built using the bag-of-words approach. Each term describes a bivariate data pattern and can either be a unigram specifying one axis label or a pattern id, a bigram specifying one axis label and a pattern id or a complete trigram specifying both axis labels and their pattern id. Please refer to Figure 3.8 for an illustrated example. With these definitions at hand, the relevance of multivariate data document D_i to a query Q can be computed as:

$$\mathrm{sim}_{\mathrm{bm25}}(D_i,Q)=\sum_{j=1}^{m}\mathrm{IDF}(q_j)\cdot\frac{\mathrm{TF}(q_j,D_i)\cdot(\alpha+1)}{\mathrm{TF}(q_j,D_i)+\alpha\cdot(1-\beta+\beta\cdot\frac{|D_i|}{\frac{1}{n}\sum_{k=1}^{n}|D_k|})}. \tag{3.6}$$

Here $\mathrm{TF}(q_j, D_i)$ denotes the term frequency of token q_j in document D_i. The term frequency is 1, if q_j occurs in document D_i and the term describes axis labels (not a data pattern). One computes the term frequency as the inverse distance of the term to its nearest cluster centroid, if the term contains a data pattern id that occurs in document D_i. The term frequency is 0 if the term does not occur at all in document D_i. $|D_i|$ is the length (number of tokens) of document D_i. α and β are weight parameters to adjust the influence of the term frequency and the inverse document frequency respectively.

$\mathrm{IDF}(q_j)$ is the inverse document frequency of token q_j given by

$$\mathrm{IDF}(q_j) = \log \frac{n - n(q_j) + 0.5}{n(q_j) + 0.5}. \tag{3.7}$$

This gives a ranking function, that allows one to compute the relevance of every document in a collection to a given query and rank the documents accordingly.

3.5.2 Ranked Aggregation

Given an initial query, an important task is to compute the most important and most discriminative of terms in the result-set. These terms can be used to provide an information seeker with an overview of the result-set and they can serve as search suggestions and auto-completions (see Section 5.3).

To compute the most important terms, one aggregates and ranks all terms of a result-set according to their respective relevance to the query. Let Q be this initial query and R be the result-set of this query consisting of documents $D_1, \dots, D_k$. The following processing steps outline the algorithm used to aggregate and rank all terms in R.

- select k most relevant documents to query Q as result set R using BM25 retrieval
- group terms of all documents in R and sum up their respective relevance scores
- filter terms according to a partial search term (e.g., a partial axis label) the user supplied
- return the h terms with the highest score as the most important terms in the result set

This yields a set of h terms from the result-set R that can be used to further refine the search. In Section 5.3, I will show how to use these terms to visualize search suggestions for the information seeker.

3.5.3 Nearest-Neighbor Retrieval Based on FLANN

To allow for nearest-neighbor retrieval and indexing of multivariate research data, the approach I proposed in Subsection 3.4.2 consisted of extracting a feature vector from the bag-of-words representation of a multivariate document based on topic modeling.

Given such a high-dimensional feature vector (100 dimensions in this case) to represent each document, an efficient indexing method is required to allow for fast nearest-neighbor querying in a collection of such documents. Several techniques like space partitioning, clustering and dimension reduction can be applied to greatly reduce the time needed to compute the *k* nearest neighbors in feature space. For this purpose I use the *FLANN*[2] library [FLA] by Muja and Lowe [ML09]. This is a C++ library for performing fast approximate nearest neighbor queries in high-dimensional feature spaces. FLANN contains a collection of several indexing and retrieval algorithms that the authors found to be best performing for different kinds of data-sets. To decide which algorithm to use, FLANN also contains a system that analyzes the data to be indexed and then automatically decides which algorithm and which parametrization to use. One can influence this decision by giving the FLANN system a desired run-time and/or precision goal.

Since the nearest-neighbor index implemented in the scope of this thesis will be used for visual-interactive retrieval, fast retrieval times are required. Precision is of secondary importance, as the topic modeling approach is very robust to changes in the data it models. Given these goals, the FLANN index that was built for this purpose uses a priority search on hierarchical k-means trees. This results in a speedup over exhaustive linear search of about 100, while retaining a precision of about 70% compared to an exhaustive linear search.

3.6 Summary

Summarizing this chapter, I presented a novel approach for extracting features in multivariate data to enable content-based retrieval. This approach was motivated by computing features that work in a similar vein as the visual-analysis of multivariate data using the scatter-plot-matrix. To that end, a set of features was extracted from all bivariate patterns occurring in multivariate data. For retrieval purposes, these feature vectors were quantized to a bag-of-words representation that allows fast and efficient retrieval of multivariate data by querying for a set of bivariate patterns. A feature vector based on topic modeling of this bag-of-words representation was proposed to compute the similarity between entire multivariate data documents, e.g. for query-by-example and nearest-neighbor indexing purposes.

[2] Fast Library for Approximate Nearest Neighbors

4 Evaluation of Retrieval Performance

In this chapter I will present an evaluation of the different techniques and approaches presented so far for bivariate and multivariate data retrieval.

So far, apart from my research, no benchmarks to support a quantitative comparison of different retrieval techniques for bivariate and multivariate have been proposed. I attribute the absence of respective data retrieval benchmarks to the difficulty of defining *similarity* for bivariate or multivariate data. For data like text, images, audio, 3D models or video, the notion of similarity usually follows a straight-forward concept. For example, similar text documents can be about the same topic or similar images show similar scenery or objects. As such, annotations that describe these notions of similarity are relatively easy to create (no experts are needed) and are readily available on community sites, such as Flickr for image collections. For retrieval in bivariate or multivariate research data, such similarity concepts are not considered so far to judge relevance of retrieved data objects to a query. A meaningful quantitative evaluation for retrieval in research data collections however, requires meaningful annotations assigned by humans to the data objects, to allow construction of similarity classes or relevance judgments.

Similar to benchmarking in multimedia retrieval, my goal is to assign data objects to similarity classes, based on meta-data annotations by experts. So far, such annotations were expensive to obtain, as experts are needed to manually annotate each data object. This prevented construction of a benchmark large enough. Due to recent efforts in the digital library community however, repositories offering expert-annotated research data became available. To construct such a benchmark, I use measurement data in the area of earth observation. Such data is available on a large scale in different repositories [PAN, Dat] and is annotated by experts. The annotations of the measurement data are done by the scientists who conducted the experiment according to pre-defined meta-data standards. The annotations describe the type of measurements and the experimental conditions under which they were obtained. I describe how to use this new data source to define similarity classes and subsequently construct a benchmark. As the automatically constructed benchmark data-set still suffers from a high intra-class divergence, meaning that objects within one class are often quite dissimilar, I propose to manually filter out the non-feasible classes from the benchmark data-set.

So the challenge in evaluating these techniques lies in the absence of standardized, established benchmarks. Hence, I will propose an approach to (semi-)automatically create suitable benchmark data-sets from publicly available data repositories in the following Section 4.1. Thereafter, these benchmark data-sets will be used to evaluate the proposed approaches for bivariate and multivariate retrieval.

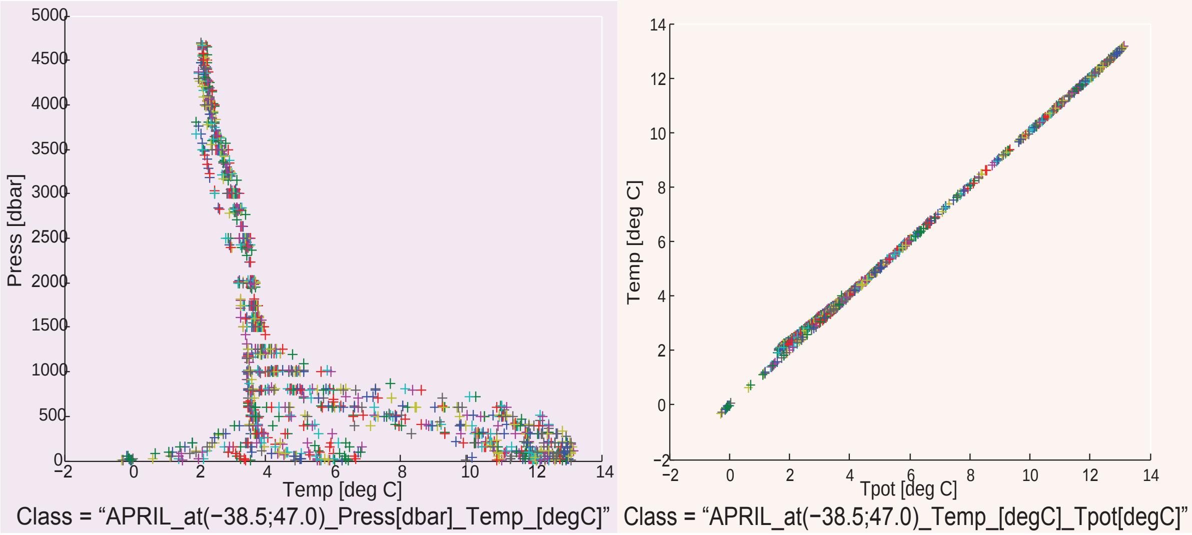


Figure 4.1: Exemplary data objects of the two largest similarity classes. For both classes (left and right), data points of all 58 objects are plotted into a single display in separate colors. The unique class labels consist of measurement type, location and time.

4.1 Constructing a Benchmark

In this section, I present my approach to construct a benchmark for retrieval in bivariate and multivariate data collections [SvLS12].

In the following subsections, I first describe the data source in detail and then cover the definition of similarity classes for the case of bivariate data retrieval. This approach is then extended to multivariate data in the subsequent section in a similar vein.

4.1.1 Data Source

For the construction of the benchmark data-set, I use earth observation data which is publicly available from the PANGAEA Data Library [DGR*02, PAN]. PANGAEA archives, publishes, and distributes geo-referenced primary research data in the domain of earth observation (water, sediment, ice, atmosphere) from scientists all over the world. It is operated by the Alfred-Wegener-Institute for Polar and Marine Research in Bremerhaven, and the Center for Marine Environmental Sciences in Bremen, Germany. Most of the data available can be publicly accessed via `http://www.pangaea.de` and can be downloaded under the Creative Commons Attribution License 3.0. Each file consists of a table of multivariate measurements, that include radiation levels, temperature progressions and ozone values, among many more. Each file available through PANGAEA is carefully annotated by the scientist who conducted the measurements. Quality control over this annotation process is taken care of

	Sum	Mean	Std	Median	Min	Max
objects: total	24,700	-	-	-	-	-
classes: total	1,608	-	-	-	-	-
objects: per class	-	15.36	10.9	11	5	58
data points: per class	-	657.98	1,043	319	51	9,770
data points correlation: avg per class	-	0.66	0.28	0.73	0.001	1.00

(a) Automatically constructed similarity classes

	Sum	Mean	Std	Median	Min	Max
objects: total	1,800	-	-	-	-	-
classes: total	90	-	-	-	-	-
objects: per class	-	20	-	-	-	-
data points: per class	-	4,077	1,409	3,155	2,121	6,711
data points correlation: avg per class	-	0.78	0.20	0.85	0.14	1.00

(b) Manually filtered similarity classes

Table 4.1: Statistics of proposed benchmark. Data point correlation is computed as Pearson's correlation coefficient for each bivariate data object and the absolute of the correlation is averaged over all objects of each class.

by the PANGAEA data curator. Most importantly for our purposes, these annotations include the type of measurement (standardized names along with base units for each measurement variable in the data table), as well as the experimental conditions (time and location) under which the measurement was conducted.

To construct a test data-set for bivariate data retrieval, I used 490 publicly available measurement files from PANGAEA. Each file contains multivariate measurements with 10 to 100 columns each. By extracting every pair-wise variable combination from each of these measurement files, I obtained 24,700 bivariate data objects which are used to form the test data set of the benchmark.

For construction of the test data-set for multivariate retrieval, I obtained approximately 10,000 publicly available measurement files from PANGAEA. From each of these multivariate data files, I extracted a feature vector based on my approach (see Chapter 3) and also defined similarity classes to allow for precision and recall evaluation for the multivariate retrieval case.

4.1.2 Automatic Similarity Classes

Given a collection of expert-annotated bivariate data, my goal is to algorithmically define similarity classes based on these annotations. Ultimately, each bivariate data object will be assigned to a single similarity class. As described above, the data that I consider for benchmark construction is geo-referenced, environmental data. To define a similarity class, I assume data measuring the same relationship (e.g., *Temperature [deg C] vs Pressure [dbar]*) at the same time of year (e.g., December) at a close-by location (e.g., *longitude* ≈ 24, *latitude* ≈ 12) to be similar. The annotations at hand contain labels for the measurement columns according to a controlled vocabulary. Thus, to compute a unique class identifier, the pair of annotated variable names is used directly. The month part of the time-stamp is by definition standardized and is extracted as well. The geo-reference is encoded as latitude ranging from -90°to 90°and longitude ranging from -180°to 180°. I discretize the location using a 6×12 grid as a coarse approximation of the different climate zones. By combining these three discrete values, we obtain a unique identifier for the sought after similarity classes. In total, 1,608 different similarity classes resulted from this algorithmical process.

The assumption for the definition of similarity classes is based on Tobler's first law of geography: "'Everything is related to everything else, but near things are more related than distant things"' [Tob70]. The decision of the discretization parameters (temporal and spatial resolution) to construct the similarity classes influences similarity of data within a class and similarity of data among classes, as neighboring data points may be assigned to different similarity classes if they are close to the discretization border. I discuss these two benchmark statistics in detail in Section 4.4.

Table 4.1a gives a detailed overview of the most important benchmark statistics. Particularly interesting is the high average data-point correlation per class, which indicates that objects exhibit some (linear) relationship, which can in principle be captured by feature extraction. Figure 4.1 shows data objects from the two largest similarity classes for illustration. We see that all objects within these two classes are numerically similar.

However looking at other similarity classes (see Figure 4.2), it quickly becomes obvious that there is a high divergence within several similarity classes. Thus, I propose to manually filter out those classes next.

4.1.3 Manual Filtering of Similarity Classes

By manually inspecting several of the automatically created similarity classes for benchmarking, it becomes clear that several are too dissimilar within a given class, and too similar across classes. Therefore, I propose to construct a second set of similarity classes manually judging the suitability of similarity classes. By visualizing and manually inspecting the classes generated automatically by my approach, I selected a reduced set of *benign* similarity classes. In particular, I filtered out classes whose objects did not exhibit a similar bivariate pattern. Examples of such classes are shown in Figure 4.2. Additionally, I filtered out classes whose patterns I saw before to reduce the similarity across different classes. Figure 4.4 shows a set of classes of which only the first one was selected. Figure 4.3 finally shows six of the 90 classes that remained after the manual filtering. The benchmark statistic of these filtered similarity classes are shown in Table 4.1b.

The precision and recall evaluation in Section 4.2 will show that the performances of the different techniques correlate between the filtered and unfiltered benchmark.

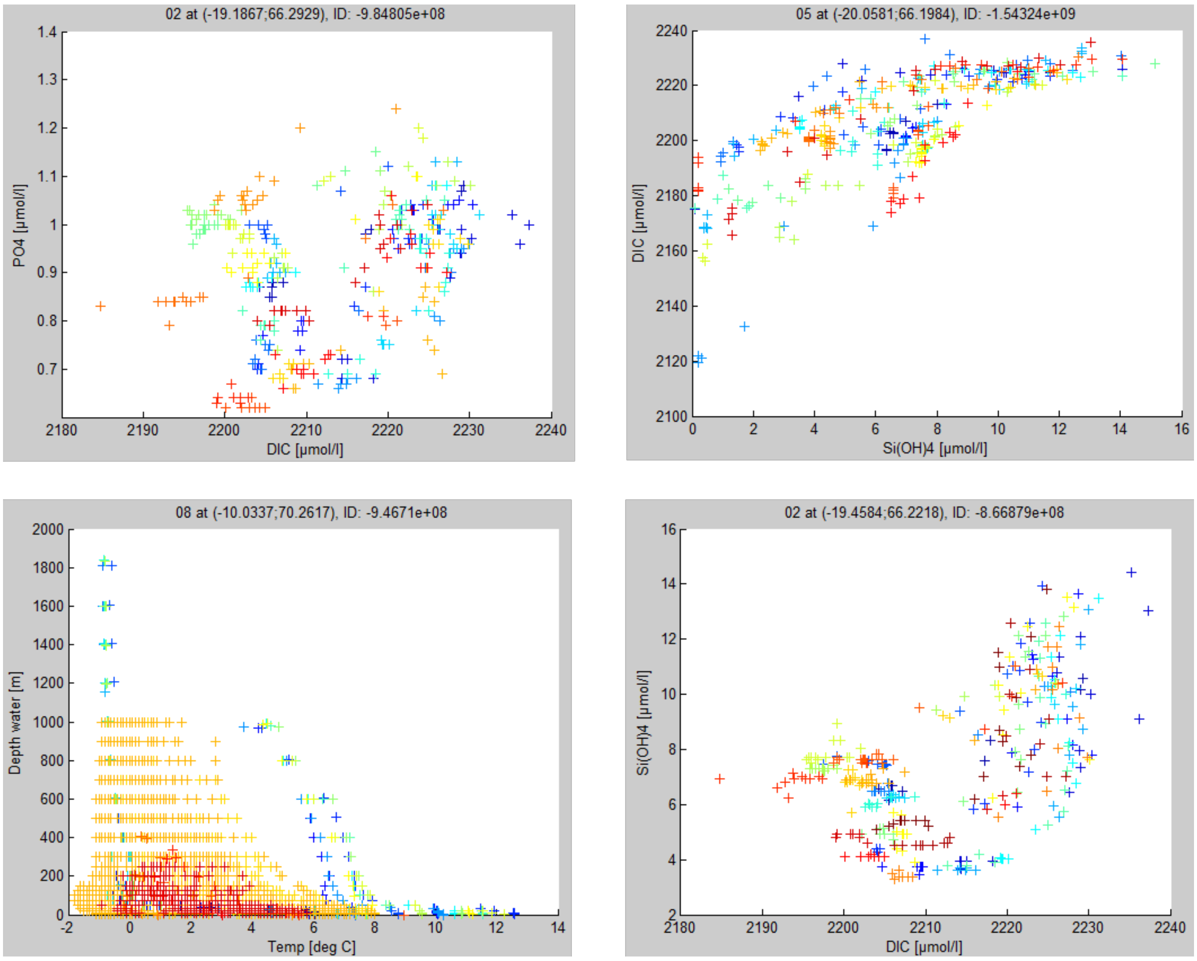


Figure 4.2: Examples of four bivariate similarity classes that were filtered out due to their high intra-class variance.

4.1.4 Extension to Multivariate Similarity Classes

To construct a benchmark data-set for multivariate data retrieval, we again need to define similarity classes, but this time for multivariate data objects. Analogously to my approach for automatically defining similarity classes for bivariate retrieval, I assign multivariate data documents to similarity classes in a similar way. Again the similarity class labels are constructed by using the annotated meta-data as a starting point.

- We quantize latitude and longitude as before in a 6 × 12 grid.
- The month of the measurement is used directly.

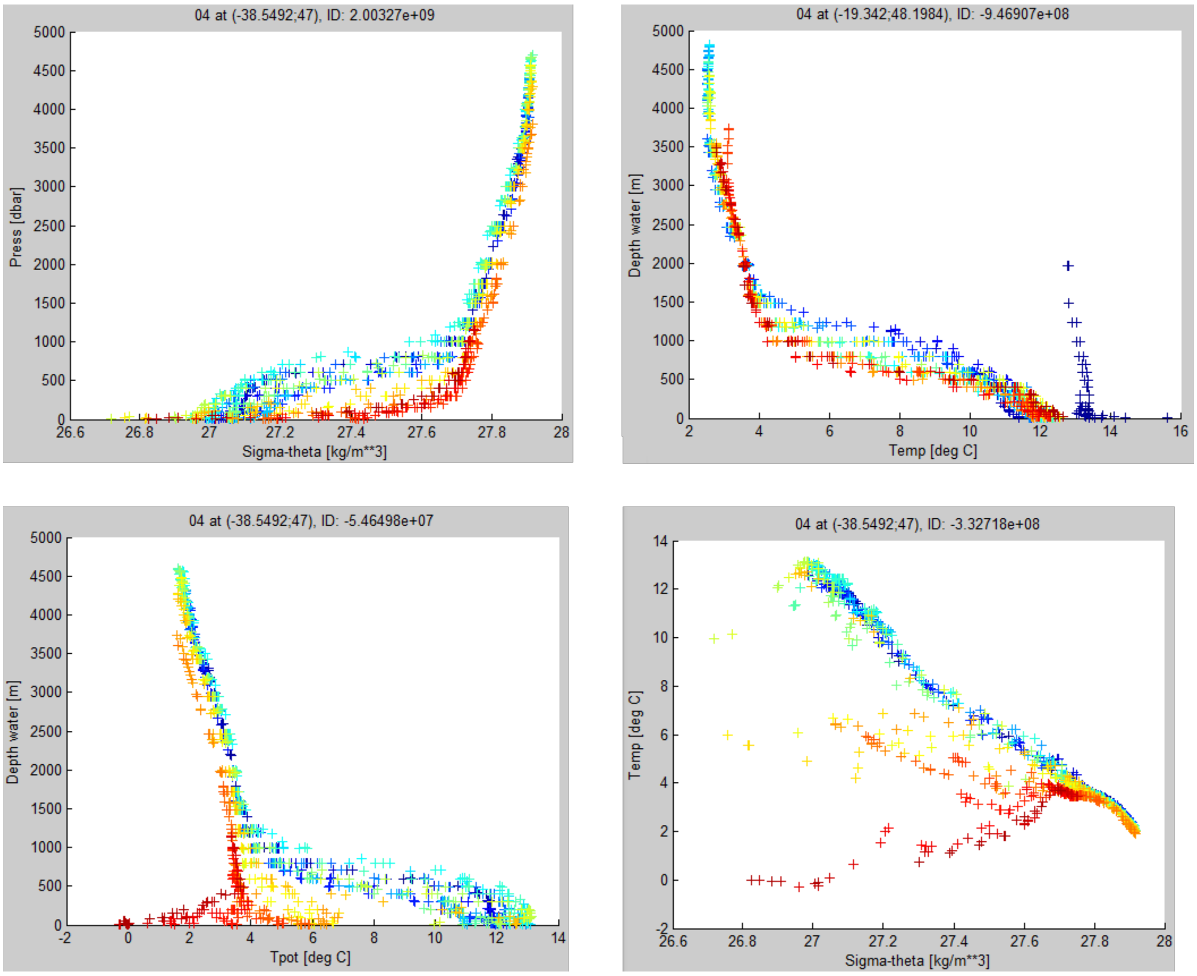
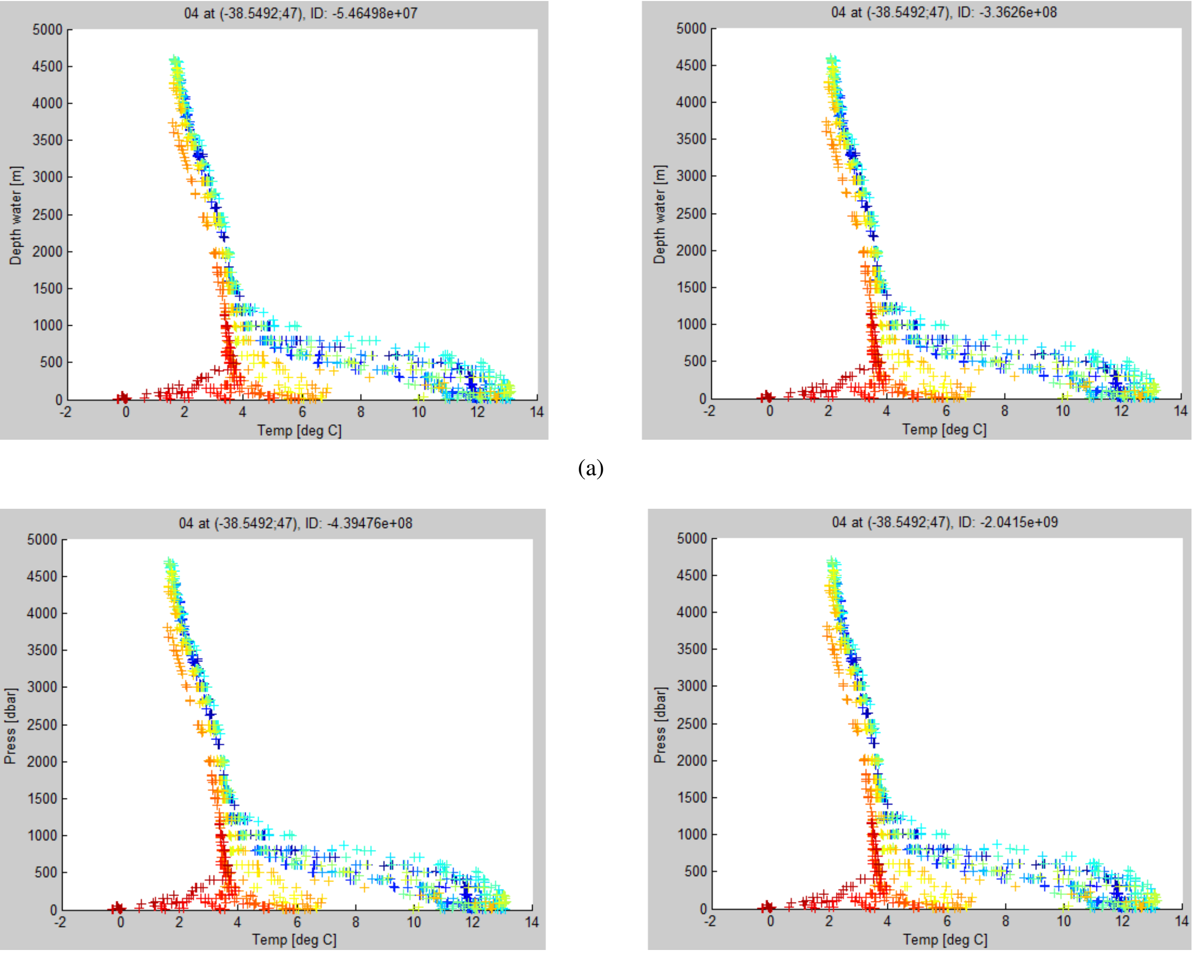


Figure 4.3: Examples of four bivaraite similarity classes that remained in the benchmark data-set as their intra-class variance is low.

- We order all measurement variables alphabetically and concatenate them. Thereby we obtain the same concatenation for multivariate documents that measure exactly the same variables, independent of the order of these variables.
- Finally all of these three categorical labels are concatenated and represented by computing a hash value thereof.

By applying the extraction scheme above in a fully automated way, I obtain 638 similarity classes with a total of 9,789 multivariate data objects assigned to them. Again, I manually inspected these similarity classes. This set of similarity classes exhibits a high intra-class variation (since measurements can vary significantly even when taken at a similar location and at a similar point in time) and a low inter-class



(a)

(b)

Figure 4.4: Example of merged Similarity Classes: In (a) and (b) we each see two similarity classes that are very similar. To increase inter-class variance of the benchmark data-set, these similarity classes were merged respectively.

variation (since measurements can be very similar, even when taken at a different location and at different point in time). To amend this problem the similarity classes are filtered and merged in a manual inspection step again, to ensure their suitability for benchmarking. This is accomplished by visualizing all multivariate data objects of one similarity class in a single scatter-plot-matrix using shadow plots. If all scatter-plots correspond approximately (determined visually, see Figure 4.5) the intra-class variation is low and the documents remain in the benchmark data-set. Otherwise, the similarity class and all associated documents are filtered out (see Figure 4.6). After obtaining 487 similarity classes in this

	Sum	Mean	Std	Median	Min	Max
objects: total	5,920	-	-	-	-	-
classes: total	376	-	-	-	-	-
objects: per class	-	15.7	3.8	16	10	20

Table 4.2: Statistics of the multivariate benchmark. 5,920 multivariate data objects are assigned to 376 different similarity classes.

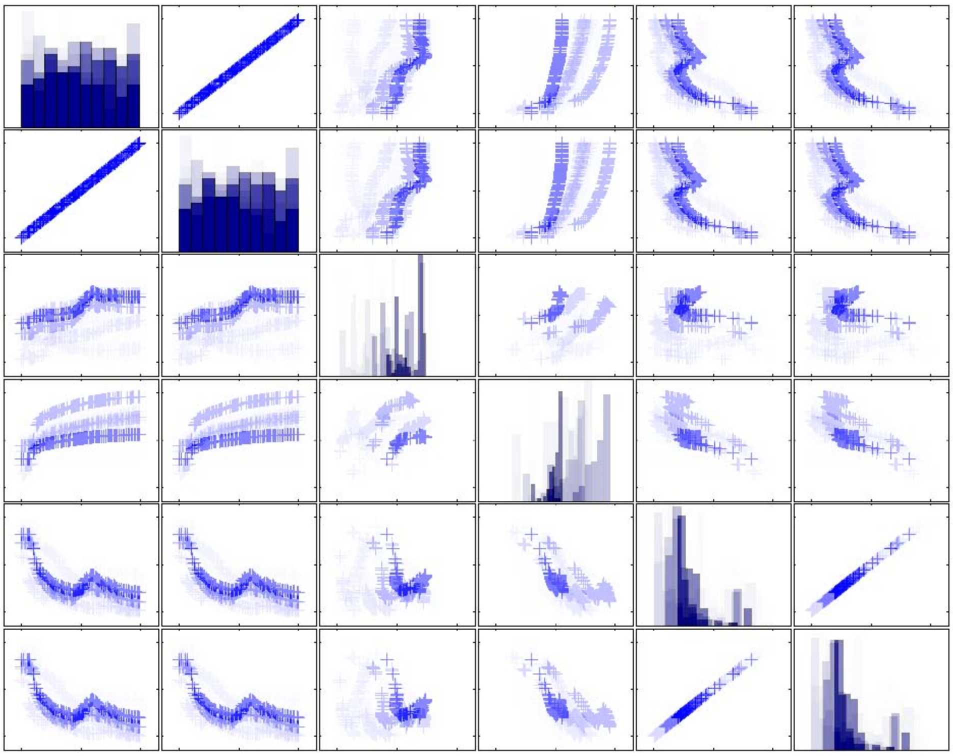


Figure 4.5: Example of a multivariate similarity class defined via meta-data that was kept in the benchmark data-set.

way, a second step is performed to increase inter-class variation. To this end, documents of different similarity classes that still measure the same variables (hence, they only differ in time and location) are compared pair-wise, by visualizing their respective scatter-plot matrices. If they correspond approximately the similarity classes are merged into one. After I completed this step, 376 similarity classes with a total of 5,920 multivariate data objects assigned to them were left (see Table 4.2).

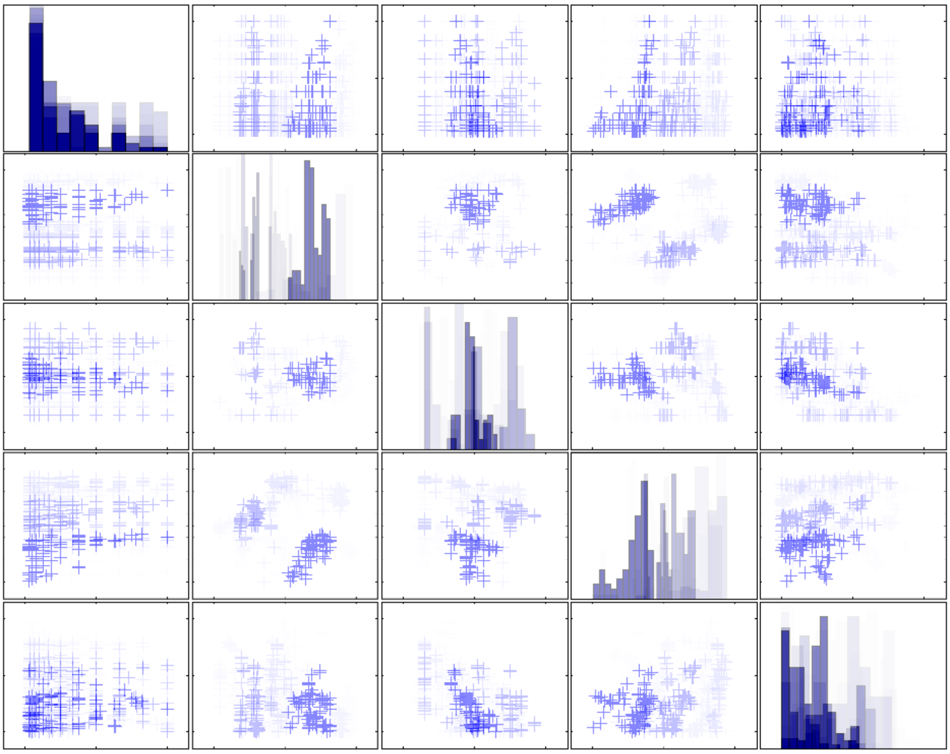


Figure 4.6: Example of a multivariate similarity class defined via meta-data that was removed from the benchmark data-set by the manual inspection.

This data-set can then be used to asses multivariate retrieval performance my measuring precision and recall. This is achieved by extracted feature vector representations of each multivariate data object and comparing the retrieval results with their respective similarity classes (see Section 4.3).

4.2 Evaluation of Bivariate Retrieval

For quantitative evaluation of effectiveness, I compared retrieval performance for each of the nine considered feature extraction algorithms (see Section 3.3.1) [SvLS12]. In particular, I use a query-by-example, leave-one-out evaluation. This means that each object is used as a query and precision and recall is computed for the ranking of all other objects in the data set. I compute r-precision (also known as first-tier precision or precision at r, see Section 2.4 Information Retrieval Metrics). To compute the r-precision, we retrieve $k-1$ objects from the data set for a given query, where k is the number of objects in the query's similarity class. Then the percentage of relevant objects within these $k-1$ retrieved objects is the r-precision. By this definition, precision at r is equal to recall at r.

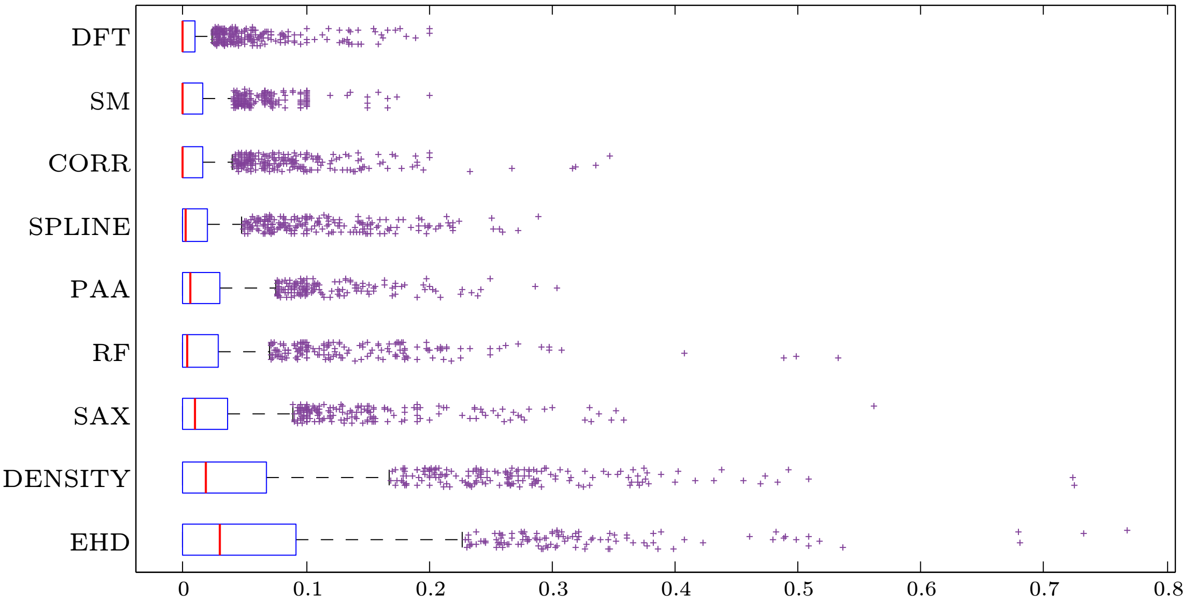


Figure 4.7: Box-plot of average *r*-precision for each studied descriptor for the benchmark data set with automatically created similarity classes. The box visualizes the 95% confidence interval around the mean. The red, vertical bar indicates the median and the scattered plus-signs show outliers. Note that the difference in retrieval precision between the techniques is significant, as randomized retrieval only reaches 0.00059 *r*-precision.

4.2.1 Results on Automatic Similarity Classes

Table 4.3, column μ_r^1 shows the average *r*-precision of the retrieval results when evaluating against the automatically defined similarity classes. Figure 4.7 shows some more detail of the results by visualizing the box-plot of the *r*-precision. We see that the average *r*-precision is between 1% and 7%, and that retrieval for several classes does not work at all (*r*-precision of zero). The difference in *r*-precision between the techniques is significant nonetheless, as an algorithm that randomly retrieves data objects only reaches an average *r*-precision of 0.059%. Particularly the image-based descriptors KDE (density) and EHD (shape) out-perform the other descriptors significantly. The only technique that performs below the two baseline techniques (CORR and SM) is the discrete Fourier transform based descriptor (DFT).

4.2.2 Results on Filtered Similarity Classes

When evaluating the techniques on the manually filtered benchmark set, overall performance increases, which of course was to be expected. These results are shown in Table 4.3, column μ_r^2 and Figure 4.8.

Descriptor	**Abbr.**	**Dim**	t_{sec}	μ_r^1 (%)	μ_r^2 (%)
Regressional Features	RF	43	2.43	2.5	20.8
Smoothing Splines	SPLINE	996	0.115	2.08	17.8
Discrete Fourier Transform	DFT	100	$0.1 \cdot 10^{-3}$	1.1	10.2
Piecewise Aggregate Approx.	PAA	100	$0.2 \cdot 10^{-3}$	2.3	15.5
Symbolic Aggregate Approx.	SAX	100	0.002	3.08	20.2
Kernel Density Estimate	KDE	1024	0.07	5.73	35.4
Edge Histogram Descriptor	EHD	80	0.26	**6.56**	**39.5**
Correlation Descriptor	CORR	3	0.028	1.7	17.2
L_2 of Resampled Data	L_2	100	0.009	1.35	9.1
Random Retrieval	-	-	-	0.059	0.7

Table 4.3: Overview of the considered feature extraction techniques for bivariate data. Average feature extraction time t_{sec} (lower is better) and average r-precision results indicate performance on the benchmark with automatically created similarity classes (μ_r^1) and manually filtered similarity classes (μ_r^2). The low r-precision of randomized retrieval shows the significance of changes in r-precision between the obtained results.

Evaluation was done in the same query-by-example, leave-one-out fashion as before. Overall, r-precision for every evaluated technique increases.

However, even with this manually filtered benchmark set, the ranking and the relative differences between the different retrieval techniques remain similar. Most importantly for the remainder of this thesis, the two best performing techniques are still the image based techniques (in particular the edge histogram descriptor) and they still out-perform other techniques significantly.

4.3 Evaluation of Multivariate Retrieval

To evaluate the performance of my approach for multivariate data retrieval, I conduct an precision-recall evaluation analogously to my evaluation for bivariate data retrieval. However, evaluating the performance of my approach for multivariate data retrieval is challenging, as there are no techniques to directly compare against.

Therefore, I chose to compare two variants of my feature extraction approach against a straightforward baseline technique. As a baseline, I compute a "column-label-descriptor" for each multivariate document as follows. A vocabulary of all column labels that occur in at least two different documents is built in an offline step. Then, a binary vector with length equal to the size of the vocabulary is extracted from each document. At position i, this vector is 1 if the document contains a measurement for variable i, otherwise it is 0. However, recalling the construction of the benchmark data-set, this label information was (among other meta-data) used to automatically construct the similarity classes

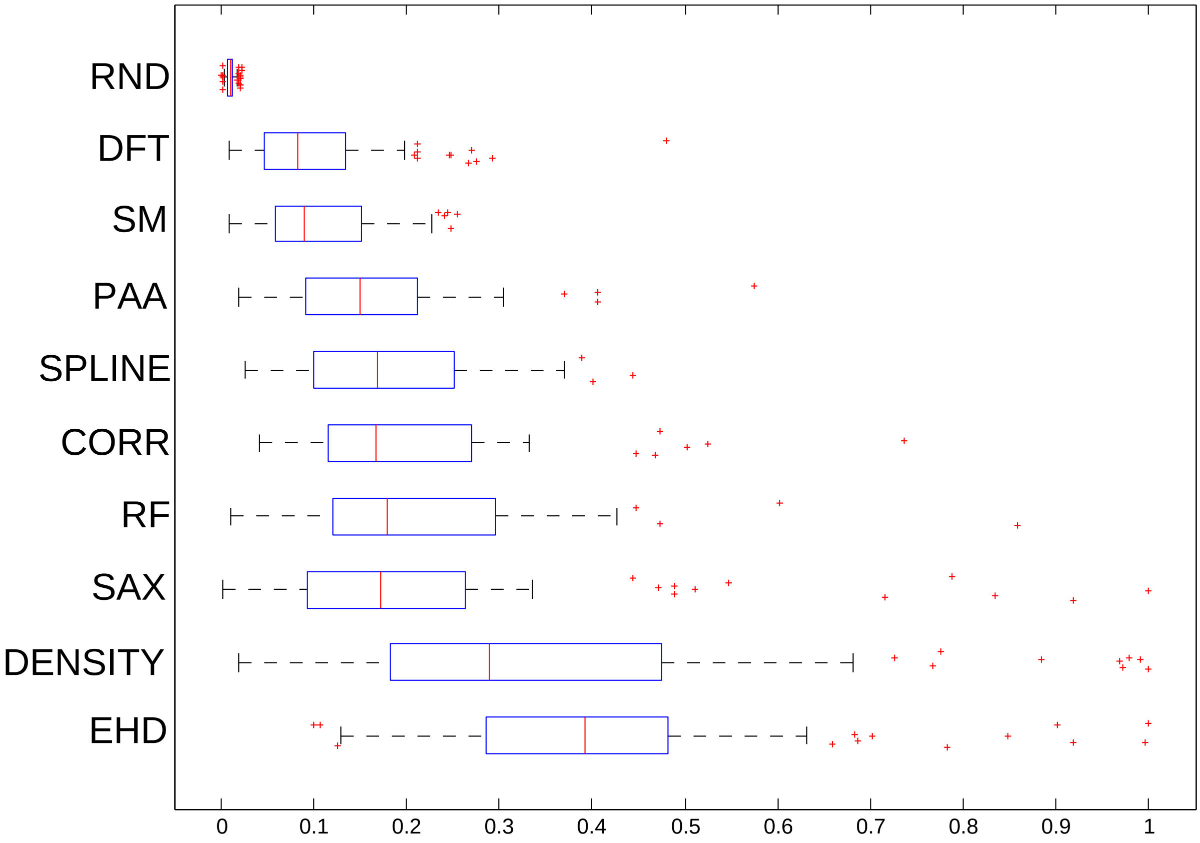
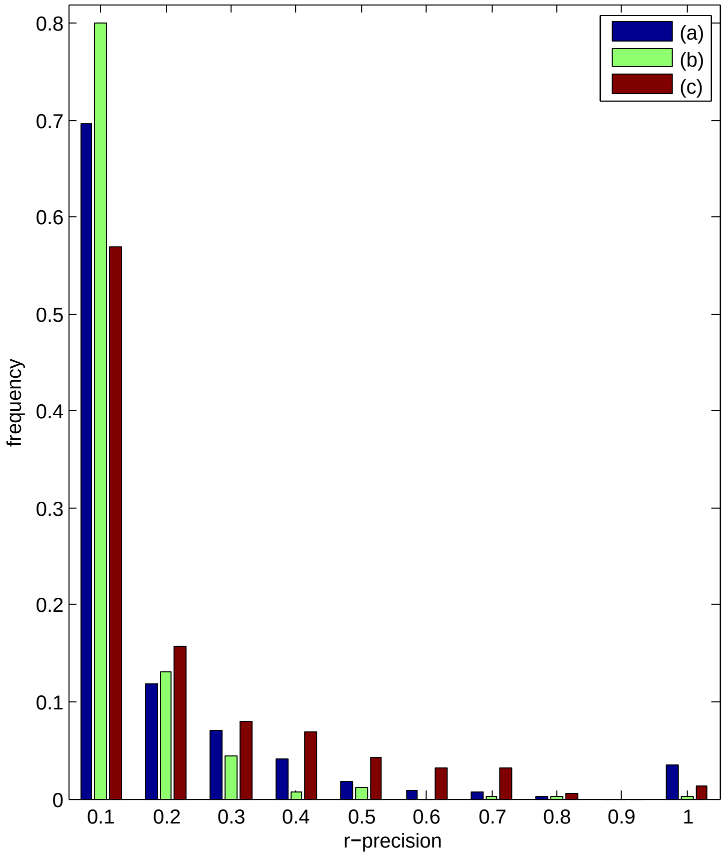


Figure 4.8: Box-plot of average r-precision for each studied descriptor for the filtered benchmark dataset. The box visualizes the 95% confidence interval around the mean. The red, vertical bar indicates the median and the scattered plus-signs show outliers.

in the first place. Since non-suitable classes were manually filtered out though, this problem is mostly amended, as the similarity classes were ultimately decided upon by human judgment.

This baseline technique is compared to two variants of my feature extraction technique. To this end, two different topic models are learned and used for feature extraction.

The first topic model is trained using *only* pattern-based feature tokens. According to the definition in Section 3.4 Multivariate Feature Extraction , these features only describe the bivariate pattern (e.g., a linear relationship) occurring between two columns of a multivariate data document, without also describing the columns themselves. Thus, these features can be computed even for multivariate data that is not annotated, as the column labels are not needed to train this topic model. Thus, a feature vector extracted this way from a multivariate data document only represents the patterns occurring in that document, but not the columns that actually exhibit these patterns.



Descriptor	**Dim**	μ_r **(%)**
(a) Column Label Descriptor	501	10.75
(b) Pattern Topic Descriptor	100	6.69
(c) Pattern+Column Topic Descriptor	100	16.26

Figure 4.9: r-Precision Evaluation of Multivariate Retrieval. (a) Baseline technique extracted from column labels; (b) topic descriptor inferred from data-terms only; (c) topic descriptor inferred from data-terms and column-label terms. We see that incorporating data-terms into the retrieval process significantly improves precision.

The second topic model is trained as proposed in Section 3.4. Thus, this topic model is based on the pattern-based terms as well as the meta-data based terms. So in comparison, features extracted this way do represent the patterns occurring in multivariate data and the columns exhibiting these patterns.

Comparing these three techniques allows us to quantify the added value of incorporating pattern-based features into the retrieval scheme. The baseline only uses annotations, the first topic model only uses patterns and the second topic model uses both.

Let us look at the results of the r-precision evaluation for these three techniques in Figure 4.9. We see that the topic modeling descriptor inferred only from the pattern terms in (b) reaches 6.69% r-precision, while the performance of the baseline technique in (a) is at 10.75% r-precision. Though the pattern-based topic descriptor does reach the performance level of the baseline technique, one should note that it can be computed without needing any kind of annotations and only consists of 100 dimensions versus the 501 dimensions of the baseline technique.

Next, let us look at the topic descriptor in (c) that uses the pattern-based terms and the column labels combined to learn a topic model. This descriptor outperforms the baseline technique significantly with a r-precision of 16.26%. It is quite interesting to note that the sum of the r-precisions of the baseline technique and the pattern-based topic descriptor almost equals the precision of the combined topic descriptor. This indicates the significant, potential gain in retrieval challenges for multivariate data by incorporating the data patterns themselves.

4.4 Benchmark Discussion

In this last section of this chapter, I will discuss several aspects of the proposed approach for benchmark construction in bivariate and multivariate data retrieval.

4.4.1 Bivariate Retrieval

For construction of the benchmark, I proposed a new approach to define similarity among bivariate data objects, by using meta-data annotations by experts in research data repositories. After assigning objects to similarity classes, an interesting questions is how well retrieval works in correlation with numerically dissimilar objects within each class (intra-class divergence) and dissimilar objects among classes (inter-class divergence).

Looking at the 2D probability density of the bivariate data objects, I compute intra-class divergence as the Euclidean distance between each data object's individual 2D probability density and the 2D probability density of all objects in the corresponding class. One can visualize this divergence as the difference of the scatter-plot density of a single data object versus the density of the scatter-plot of all bivariate data objects at once (visualized in Figure 4.1).

To judge inter-class divergence, we select all objects of two random classes and again compute the distance of the 2D probability density. This time, we compute the distance between the distribution of all the data points of these two random classes and the data point distribution of each individual object. By repeating this experiment until convergence for each class, we get an average distance of objects in different similarity classes – thus the inter-class divergence.

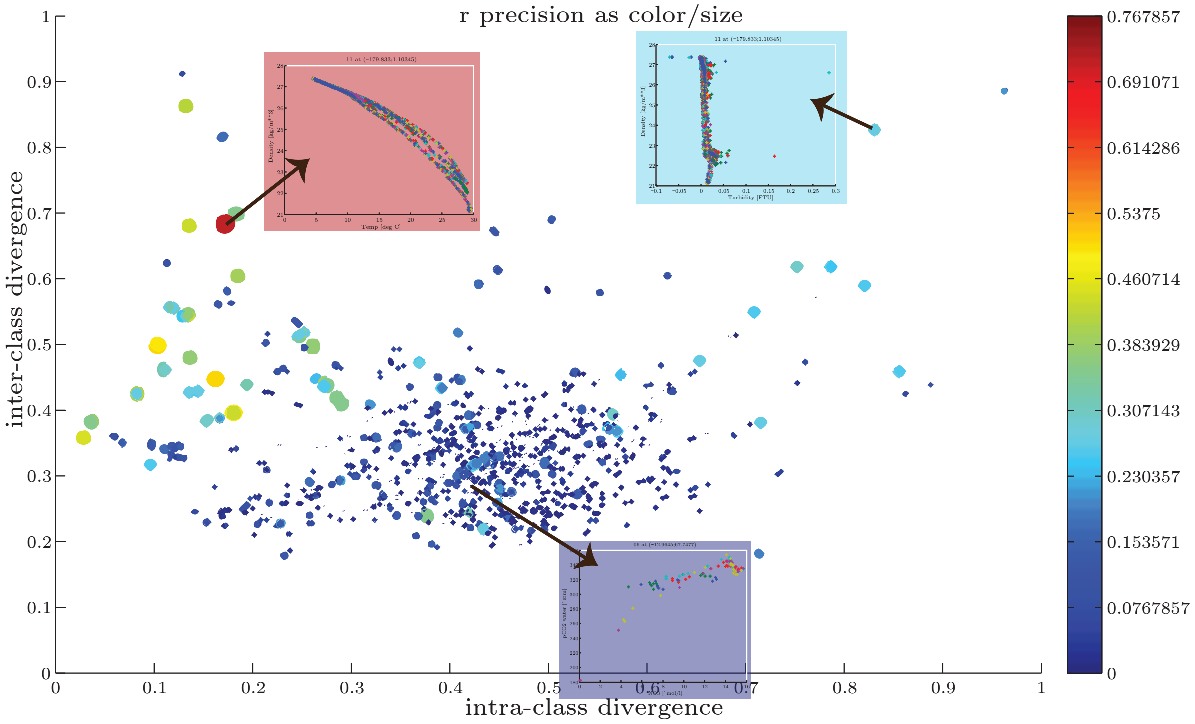


Figure 4.10: Intra-class divergence plotted versus inter-class divergence for each similarity class in the benchmark. Color and size of data-points redundantly visualize mean average *r*-precision for each class. Data-point color in the small sub-images show class-membership.

Let us explore the *r*-precision of individual similarity classes and their respective inter- and intra-class divergence in detail. Figure 4.10 shows an overview of this relationship. As expected, we see that well performing classes (big, non-blue points) are primarily located in the upper left corner and thus exhibit a low intra-class divergence and a high inter-class divergence. However there are a few well performing classes in the upper right corner. These classes exhibit a very high intra-class divergence (which makes retrieval difficult), but at the same time a high inter-class divergence. This indicates that for bivariate retrieval, numerical discrimination from other classes is more important for good retrieval results than numerical similarity within a class.

4.4.2 Multivariate Retrieval

Next, let us look at some qualitative retrieval results for multivariate data retrieval. In Figure 4.11 we see the retrieval results using an object from the best performing similarity class as the query. As expected, all scatter-plots visualized using shadow drawing are very similar to each other. In Figure 4.12 on the other hand, we see the retrieval results using an object from the worst performing similarity class. Even though the similarity between the scatter-plots is not as high, they are still quite similar, considering the data document used as the query object in this case yielded an exact 0.0 r-precision on the benchmark. This indicates that there is a semantic gap between the way I defined the similarity classes based on the annotated meta-data and the actual data-patterns. As my approach for multivariate data retrieval does not try to model any of the (application specific) semantics I had in mind while creating the benchmark (like location or time), it can only retrieve data that exhibits similar patterns.

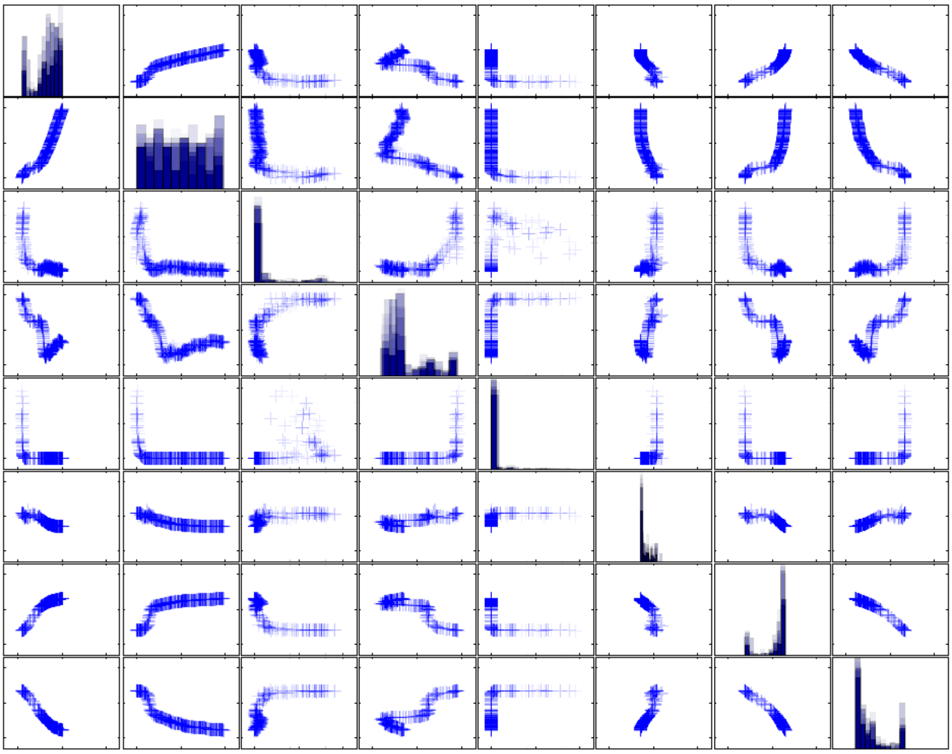


Figure 4.11: Qualitative results using a multivariate data document from one of the best performing similarity classes. The query object and its ten nearest neighbors are plotted in a single scatter-plot matrix as shadow-plots in shades of blue. Here, those scatter-plot matrices are nearly identical to each other.

4.5 Summary

In this chapter I evaluated the retrieval performance of my approach with respect to two different aspects. First, I compared nine different feature extraction techniques for bivariate retrieval. The best performing technique was an image-processing based technique I developed. The basic idea was to

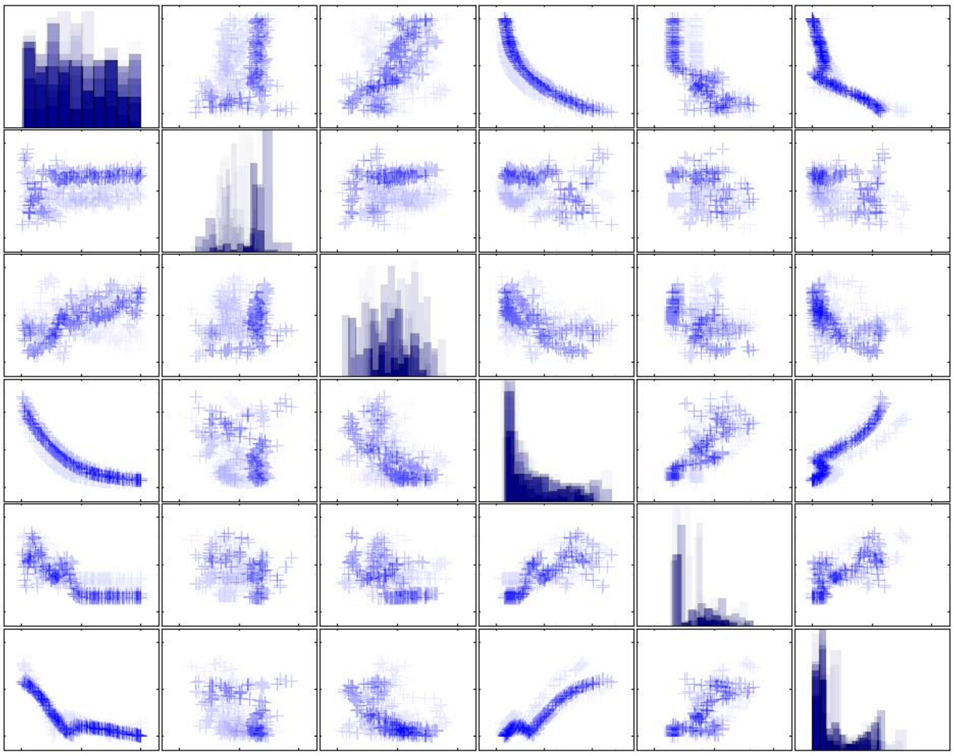


Figure 4.12: Qualitative results using a multivariate data document from one of the worst performing similarity classes. The query object and its ten nearest neighbors are plotted in a single scatter-plot matrix as shadow-plots in shades of blue. The scatter-plot matrices of these documents vary significantly.

estimate and plot the density of the data's scatter-plot and then describe this shape of this density as an edge-histogram. Secondly, I compared my approach for multivariate data retrieval to a baseline technique, which relies on meta-data alone. I was able to show that my technique, which includes content-based features on top of meta-data based features, significantly outperforms the baseline technique.

5 Visual-Interactive Retrieval

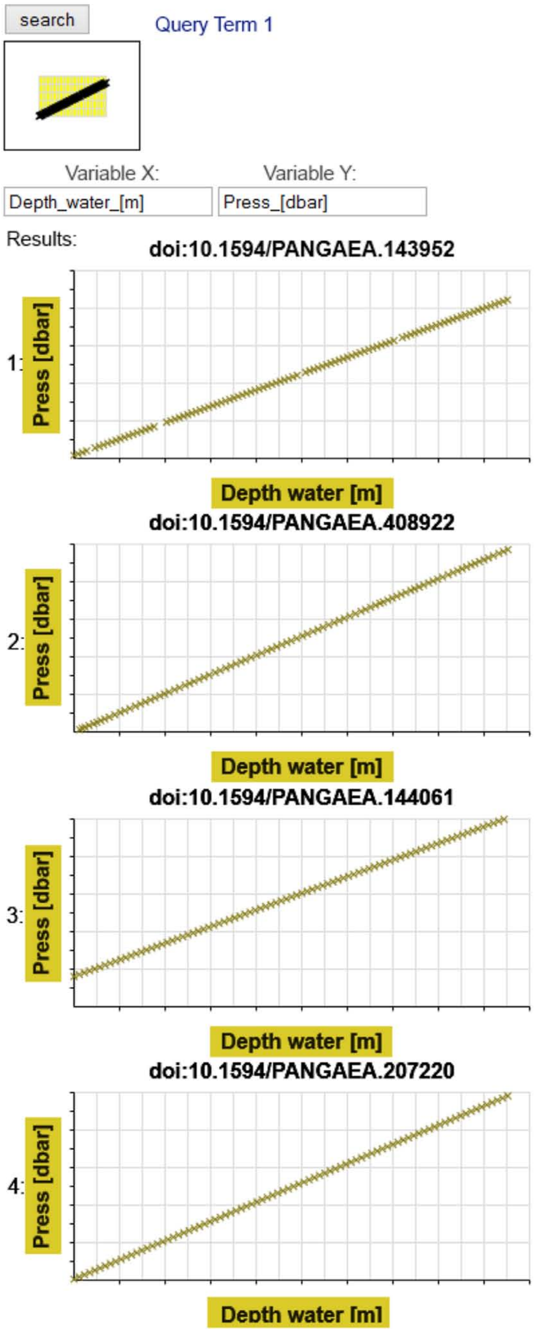
In the previous chapters 3 and 4 of this thesis, I described and evaluated my algorithmic approach for retrieval of multivariate research data. One aspect I have not consider at so far, is how to support information seekers in making use of this algorithmic approach.

Little research focused on *interactive methods* which use such similarity functions to help the user with the query formulation process based on data content. Such methods include highlighting of results to show why a document was retrieved, as well as search suggestions to provide the user with an overview of meaningful terms she can search for next. These functions are typically located on the front-end of a visual-interactive retrieval system, but require indexing structures in the back-end to be efficient.

In this chapter, I will present a novel approach for providing the user with interactive search suggestions and result highlighting when querying multivariate data [SvLS13b]. Such visual-interactive tools are already successfully used in textual search engines and yield similar advantages to users querying non-textual research data documents. In particular, users have certain expectations how retrieval systems should work and what query options they should provide [KH12]. Search suggestions provide information seekers with an overview of (often complex) data patterns and variable names to initially search for or to auto-complete or refine their search. Result highlighting shows the information seeker, which part of a document matched her query and thus explains *why* it was retrieved.

Akin to search suggestions on popular web- or e-commerce search engines, I present the user with search suggestions and completions, based on her partial query as it is being entered. Figure 5.1 shows an example of this suggestion-approach. Furthermore, I can provide the user with instant search results and also highlight those parts of a retrieved multivariate data-set, that correspond to the user's query. Similar to paragraph highlighting in text retrieval, I propose to show those scatter-plots of a retrieved data-set, that contain parts of the query (e.g., a textual match on the axis label or a match of a particular scatter-plot pattern) and to highlight these parts. That way a user can see why a particular document was retrieved in the first place, and quickly skim through the results to find the data-sets she is most interested in. Another example in Figure 5.2 shows a result list and the highlighted scatter-plots.

To allow for this kind of visual-interactive querying in multivariate data, I developed the novel indexing method based on a bag-of-words approach as described in Section 3 Approach to Multivariate Research Data Retrieval .



Variable X: Depth_water_[m]

Variable Y: Press_[dbar]

Figure 5.1: An example query using the *PANGAEA* data repository, searching for a linear relationship between water depth and water pressure. This query pattern (variable x, variable y, curve form) we searched for is highlighted in each retrieved data-set.

5.1 Instant Results

Providing results quickly to an information seeker's query is very advantageous. It speeds up the search process in general, and also allows the user to avoid over-specifying her query, by providing instant results while the query is being formulated. To allow for "instantaneous" search results (in general, less than 300ms), I use the bag-of-words index described in Chapter 3 for retrieval. Recall that an information seeker can search for arbitrary meta-data, column labels and data patterns such as a specific relationship of *a* versus *b* with pattern *c*. Figure 5.1 shows an example search query. In this example, I searched for a complete tri-gram by specifying both axis labels (water depth versus pressure) and a data pattern (a linear relationship between those two variables). For ranking I use a modified BM25 approach as described in Subsection 3.5.1.

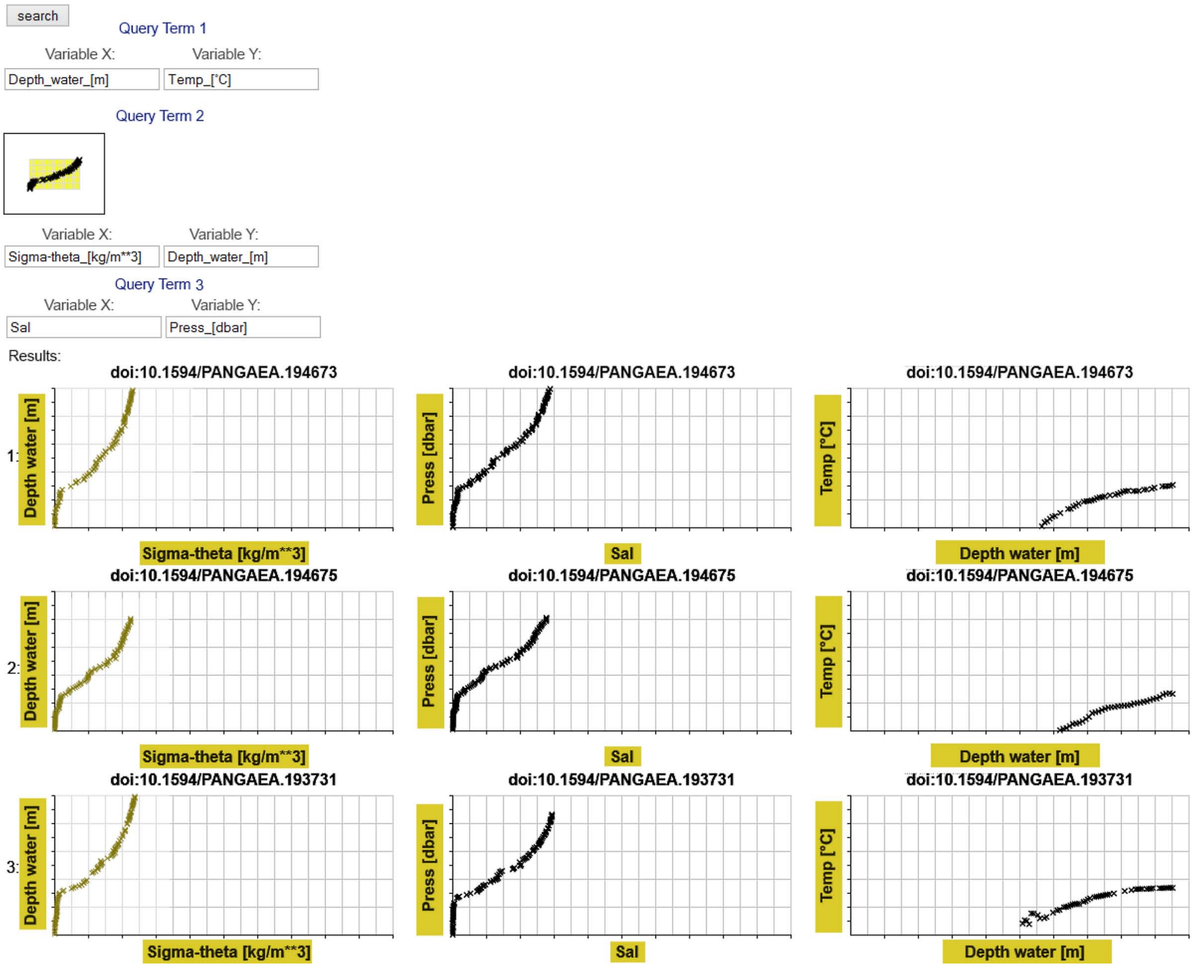
As with any full-text index that is based on inverted lists, we can efficiently combine several search terms by intersecting the associated lists. Thus, the default behavior of my approach is to look for all search terms, and only return those documents, that contain every search term and rank them according to their aggregated term weight.

Due to the full-text-like indexing of the bag-of-words approach, the system is able to perform search queries in less than 300 milliseconds, which is generally accepted as “instantaneous” in retrieval applications. Thus, while the user is still formulating her query, we provide her with immediate results as this has been shown to speed up the retrieval process.

As long as the full-text-index and the primary key index of the database fully reside in the system's main memory, the look-up part of the query time is independent of the number of documents and is dominated by the time required to read the result data from the hard disk.

5.2 Result Highlighting

Highlighting of search results is very important to explain to the user, why a particular document is being returned. For text retrieval, highlighting the search terms and showing a few surrounding sentences is a very suitable way to do so. I adapt this to the retrieval scenario at hand. For each retrieved multivariate document, I want to explain to the information seeker why this particular document was retrieved. For this kind of document, merely showing the title or the identifier of the document is too obscure for this purpose. On the other hand, visualizing the complete scatter-plot-matrix and highlighting individual scatter-plots takes up too much screen-space, considering scatter-plot-matrices can be huge and that we would need to visualize one for each retrieved document. Thus, I propose to filter the scatter-plot matrices for individual scatter-plots that contain at least one of the query terms and then show a fixed number of these individual scatter-plots sequentially. These scatter-plots visualize those bivariate patterns in each document that (partially) matched the user's query. Furthermore, I highlight those scatter-plots by coloring the axis labels and / or the data-points, depending on their match with the search term. Figure 5.2 shows an example query, where scatter-plots of each returned document highlight the user's query matches.



search

Query Term 1

Variable X: Depth_water_[m]

Variable Y: Temp_[°C]

Query Term 2

Variable X: Sigma-theta_[kg/m**3]

Variable Y: Depth_water_[m]

Query Term 3

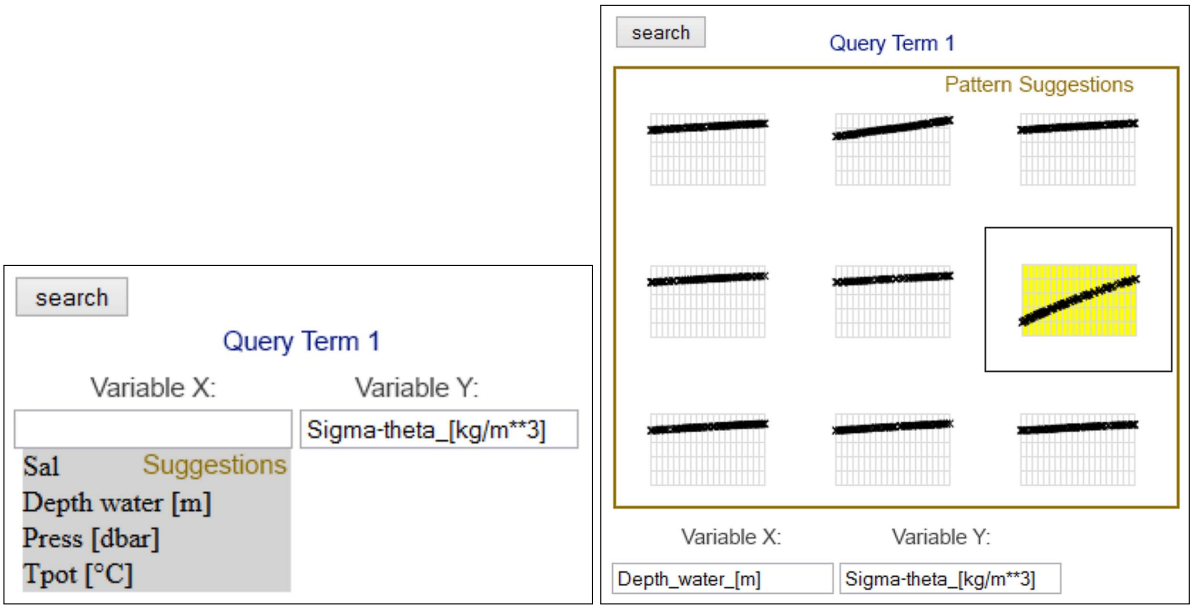
Variable X: Sal

Variable Y: Press_[dbar]

Results:

1:
doi:10.1594/PANGAEA.194673
Depth water [m]
Sigma-theta [kg/m**3]
doi:10.1594/PANGAEA.194673
Press [dbar]
Sal
doi:10.1594/PANGAEA.194673
Temp [°C]
Depth water [m]
2:
doi:10.1594/PANGAEA.194675
Depth water [m]
Sigma-theta [kg/m**3]
doi:10.1594/PANGAEA.194675
Press [dbar]
Sal
doi:10.1594/PANGAEA.194675
Temp [°C]
Depth water [m]
3:
doi:10.1594/PANGAEA.193731
Depth water [m]
Sigma-theta [kg/m**3]
doi:10.1594/PANGAEA.193731
Press [dbar]
Sal
doi:10.1594/PANGAEA.193731
Temp [°C]
Depth water [m]

Figure 5.2: Search Result Highlighting on our front-end: For each of the retrieved multivariate documents, the scatter-plots that contain the search terms are highlighted by coloring the axis labels and/or the data points in yellow. I searched for multivariate documents that contain a sigmoid like relationship between water density (sigma-theta) and water depth, as well as an arbitrary relationship of water depth versus temperature and salinity versus pressure. Note that the retrieved documents contain all of these search terms; the system highlights the results accordingly to show the user, why these documents were retrieved.



Variable X:
Variable Y: Sigma-theta_[kg/m**3]

Variable X: Depth_water_[m]
Variable Y: Sigma-theta_[kg/m**3]

(a)

(b)

Figure 5.3: Search Suggestions: (a) After specifying one axis label for the search term, the system presents search terms for the other axis. In this example I specified water density as the y axis. The system suggests to search for salinity, water depth, pressure or temperature – precisely those four variables that water density is functionally dependent of [Ste04]. (b) After selecting one of the suggested x-axis labels, the system suggests search terms for the curve progression by visualizing small scatter-plots to the user.

5.3 Search Suggestions

Search suggestions enable two core benefits for information seekers. First, they provide an overview of search terms to refine an initial query with and formulate a more-precise query, while making sure that the query is not over-specified and can still retrieve matches. Secondly, search suggestions serve as auto-completions that allow users to formulate their desired query faster and with less typing errors. Search suggestions and auto-completions gained particular popularity due to their introduction into Google's and Amazon's search front-ends. Since then, search suggestions have also become central to the user's expectations (or mental model) how a search engine works [KH12]. As such, failure to provide the user with this functionality often leads to queries with no results due to the search for non-existent patters. In other cases, users do not have a precise pattern in mind to search for (or to continue / refine their search with). In these cases, search suggestions provide the user with a much needed overview of patterns she can search for next.

To provide search suggestions for retrieval of multivariate data, recall that my approach for aggregating terms in a given result-set works as follows (compare Subsection 3.5.2):

- retrieve d documents that contain all query terms the user searched for so far
- sum up the ranking scores for all tri-gram terms in the result set
- restrict possible tri-grams according to a partial search term (e.g., a partial axis label) the user supplied
- return the h tri-grams with the highest score as search suggestions and visualize them accordingly

Such an aggregation function yields a list of tri-gam terms, that each contain both axis labels and the data pattern of the bivariate data pattern they describe. Thus, I can visualize a set of suggestions from these terms by showing a textual pop-up of potential x- and y-axis labels, as well as a set of scatter-plots as small preview images (see Figure 5.3). This allows users to quickly select columns and patterns they are interested in.

The parameter d, the number of documents retrieved to select search suggestions from, influences the quality of the suggestions (higher d is better) but also the computational cost of the suggestions (lower d is better). I found $d = 50$ to be a good compromise between run-time and search suggestion accuracy.

5.4 Nearest-Neighbor Recommendations

Using nearest neighbor retrieval to query and explore large document collections via pairwise document similarity is a well known approach to discover documents of interest. Particularly in e-commerce, recommender systems try to solve this problem, by recommending similar documents to a given one, presenting the user with alternative or complementary products. Though in e-commerce these systems usually rely on collaborative filtering and not on data content similarity, I use this retrieval paradigm to explore multivariate data collections. To this end, I implemented a "Related Documents" approach that presents the k nearest neighbors to the document we are currently viewing (thus, the four most similar documents with respect to my approach). The features used for the nearest-neighbor retrieval are based on topic modeling (see Section 3.4.2), the indexing method is based on FLANN as described in Section 3.5.3.

Once the k nearest neighbors are retrieved, the task is to visualize them in an intuitive way to the information seeker. To this end, I chose to render a large, single scatter-plot-matrix that contains the scatter-plots of all k nearest neighbors. Each scatter-plot itself is rendered using shadow-drawing. This allows the information seeker to visually inspect the similarities of the neighboring documents and select documents of potential interest. Figure 5.4 shows a (small) visualization of several multivariate documents in a single scatter-plot-matrix.

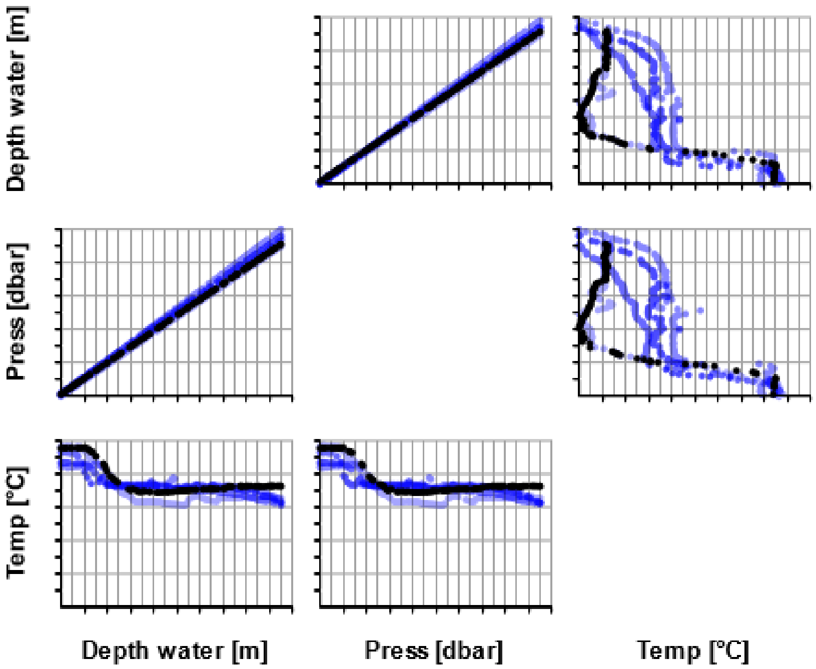


Figure 5.4: Example visualization of several multivariate documents in a single scatter-plot-matrix using shadow-plots.

5.5 Summary

In this chapter I proposed several visual-interactive techniques for retrieval in multivariate research data based on the feature extraction approach of this thesis. Result highlighting explains to an information seekers, why certain documents were retrieved by presenting the context. For multivariate data, I proposed to meet this challenge by visualizing all the scatter-plots that (partially) contain one of the terms the seeker queried for. Search suggestions provide the information seeker with an overview of patterns to refine her search with. They are computed by aggregating and ranking all the terms of a result-set according to their respective relevance to the query. Finally, a visual-interactive interface for nearest-neighbor browsing was presented, that allows an information seeker to explore collections of multivariate-data by exploiting the similarities that exist between these documents.

6 Use-Case

In this chapter I present a use-case of retrieval in multivariate research data using my approach. The objective of this chapter is to qualitatively explore some of the benefits my approach can provide to information seekers.

As a use-case, I chose the domain of physical oceanographic research. This is a branch of the earth or environmental sciences that is concerned with physical processes in the ocean [Ste04]. Physical oceanography is a very data intensive science (see Section 2.9.1). Measurements are obtained via ships, observatory stations, buoys and other maritime sensorics and are globally connected to earth observations networks. Data libraries such as PANGAEA [PAN] or DataONE [Dat] make these measurement data-sets publicly available to enhance research, exchange, re-use and long-term availability. These circumstances make oceanography a good candidate to benefit from novel retrieval techniques.

In related work, content-based retrieval for time-oriented research data in PANGAEA was useful to scientists using such data repositories to fulfill their information needs [BBF*10]. As my approach generalizes from time-series data to multivariate data, a similar practical benefit for this data domain can be expected. In the following I will describe the data source and how one can employ my approach to retrieve data from PANGAEA.

6.1 Data Source

As a use-case for my approach to multivariate research data retrieval I consider real-world data from the *PANGAEA* Data Library [DGR*02,PAN], which I also used for evaluation in Chapter 4. *PANGAEA* is a digital library for the environmental sciences. It archives, publishes, and distributes geo-referenced primary research data from scientists all over the world. It is operated by the Alfred-Wegener-Institute for Polar and Marine Research in Bremerhaven, and the Center for Marine Environmental Sciences in Bremen, Germany.

For my experimental setup, I acquired every document that is currently available under the Creative Commons Attribution License 3.0 from `http://www.pangaea.de`. Each document is uniquely identified with a digital object identifier (DOI) and consists of a table of multivariate measurements. These measurements include radiation levels, temperature progressions and ozone values, among many more. Each document available at *PANGAEA* is carefully annotated by the scientist who conducted the measurements. A data curator controls the quality of this annotation process. These meta-data annotations include standardized parameter names, base units as well as the time and the geo-location of the measurements.

I obtained 98,416 such documents in total. These multivariate documents contain between 3 and 100 columns, and approximately 220 rows on average. The raw uncompressed data of these documents requires approximately 35 GB of disk space.

Following my indexing approach described in Chapter 3, I computed and indexed approximately 2.5 million terms. This requires about 2 GB of RAM to keep the index fully within the main memory of my test setup. Additionally to allow for nearest-neighbor retrieval, I indexed the topic feature vectors extracted from each multivariate document in a FLANN index, which requires about 500 MB of RAM.

6.2 Oceanography

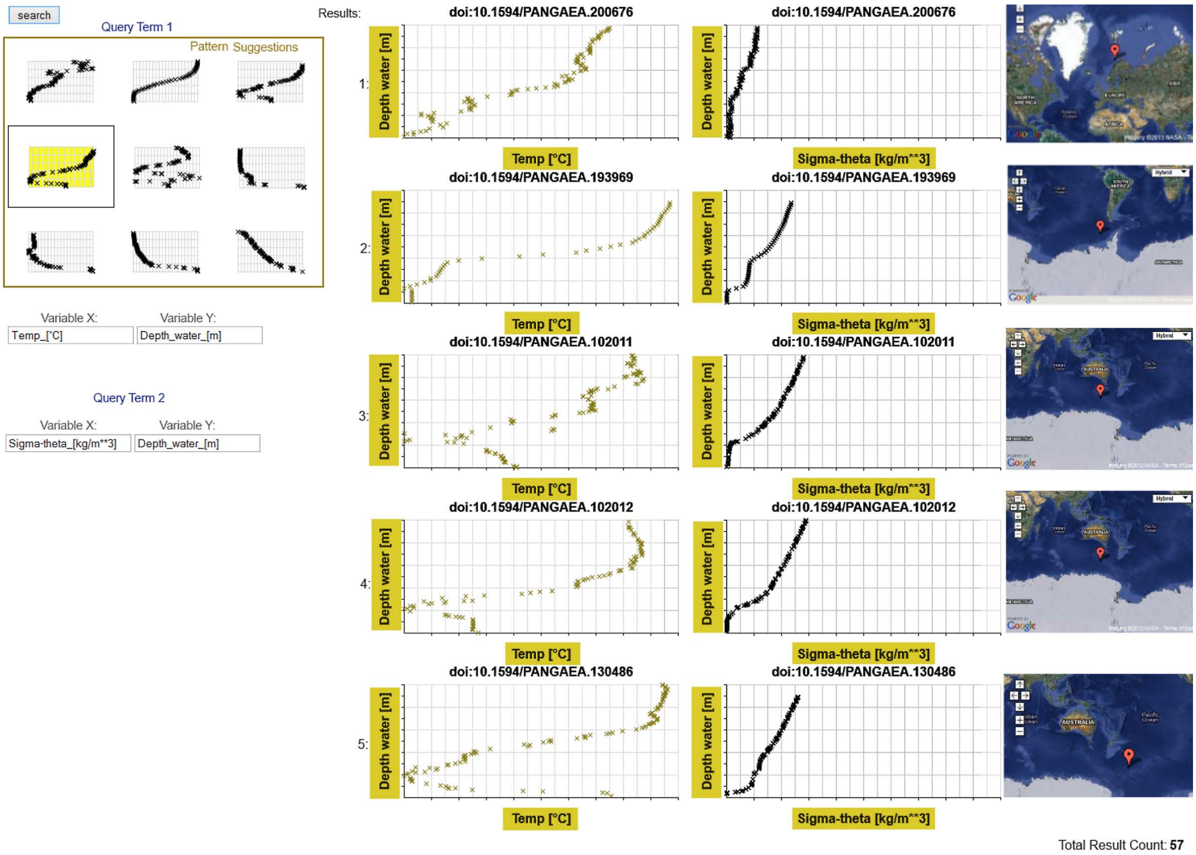
Oceanography is a branch of the environmental sciences that is concerned with physical processes in the ocean by measuring important properties via ships, observatory stations, buoys and other maritime sensorics. Many of these observation devices are connected to global earth observation networks that make these measurement data-sets publicly available. Finding data-sets relevant to an information seeker is a major challenge in this area and will serve as a use-case here.

6.2.1 Querying

In this first subsection, I query the wealth of oceanographic data indexed with my approach for a case study. Assuming no prior knowledge of the contents of this data repository, I look for data-sets that show similar measurement patterns as part of an exploratory search process. It can be the basis to hypothesize about the reason for the observed similarities.

First, I enter two intuitive variables, namely water temperature and water depth. Since I do not know what kind of pattern a measurement between these two variables should look like, I let the system provide an overview of important patterns (see upper left part of Figure 6.1). Initially I assumed temperature to either drop or rise with water depth (depending on whether the environment is warm or cold). However, I was surprised the system suggested a pattern that indicates a drop in temperature until a certain water depth, and then an increase in temperature as we go deeper (see scatter-plot "Suggestions" in upper left of Figure 6.1). This effect of discovering something unexpected, yet interesting is often referred to as *serendipity*. Thus, I selected and searched for this pattern. Moreover, I want the result-set to also contain measurements of water density versus water depth, to see if these will be similar to each other as well. The result of this query can be seen in Figure 6.1.

The result set that was retrieved did exhibit the pattern I queried for and was highlighted accordingly. On top of that, the relationship of density versus depth was also similar. Looking at the locations where those measurements were taken, I was initially surprised to see such different locations as the Norwegian sea and the Antarctic southern ocean. However, looking into some details about the Norwegian sea did reveal that it is a maritime subarctic climatic zone just like the Antarctic southern ocean, explaining the similarity of the measurement patterns.


search
Query Term 1
Pattern Suggestions
Results:
1:
2:
3:
4:
5:
doi:10.1594/PANGAEA.200676
doi:10.1594/PANGAEA.193969
doi:10.1594/PANGAEA.102011
doi:10.1594/PANGAEA.102012
doi:10.1594/PANGAEA.130486
Depth water [m]
Temp [°C]
Sigma-theta [kg/m**3]

Query Term 1

Variable X: Temp_[°C]

Variable Y: Depth_water_[m]

Query Term 2

Variable X: Sigma-theta_[kg/m**3]

Variable Y: Depth_water_[m]

Total Result Count: **57**

Figure 6.1: Case Study: I queried for a specific pattern between temperature and water depth to see whether the documents containing this pattern were measured at locations with a similar maritime climate. The first document is situated in the Norwegian sea, while documents 2 to 5 were obtained in the Antarctic southern ocean. Both regions have a maritime subarctic climatic zone, explaining why the same pattern was found there. Map Data is attributed to Google Maps.

Further refinement of the search by selecting a specific pattern for the relationship between water density and water depth as well, did reduce the result set to a more homogeneous region as I expected at that point (see Figure 6.2). Using that query, all retrieved documents were measured in the Antarctic southern ocean, approximately at the longitude of New Zealand (169°).

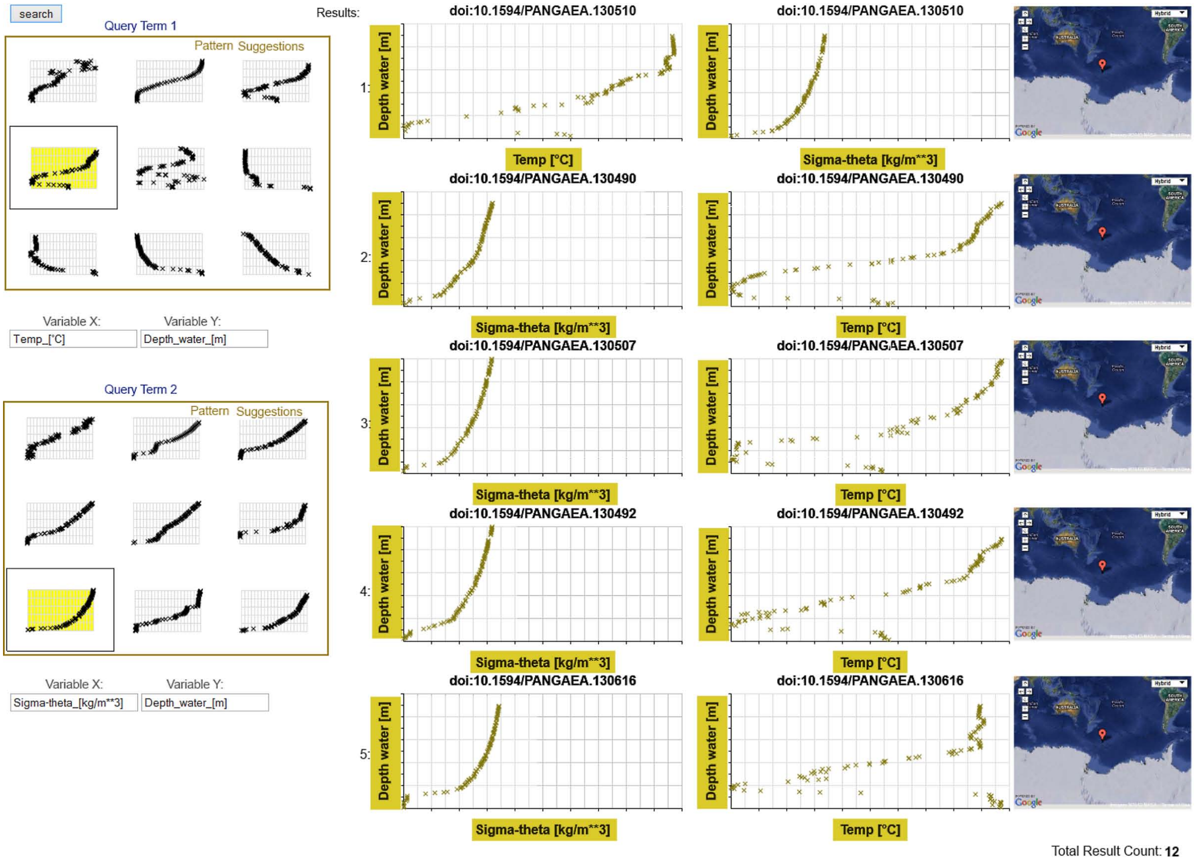


Figure 6.2: Case Study: I refined the original search query (see Figure 6.1) by also selecting a specific relationship between water density and water depth that was suggested by system. We see that this combination of patterns only occurs in documents that originate from the Antarctic southern ocean at longitude 169°. Map Data is attributed to Google Maps.

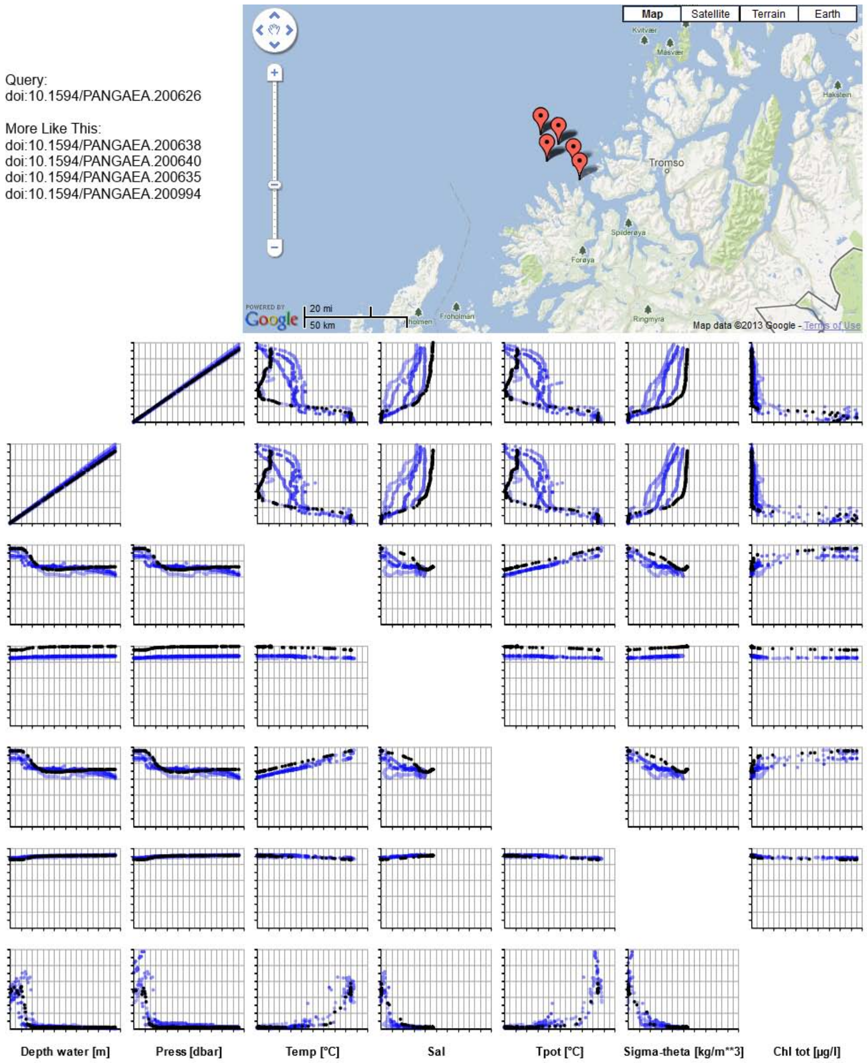
6.2.2 Browsing

In this second subsection of my use-case, I want to browse and explore the oceanographic documents using the nearest-neighbor recommendations I proposed in Section 5.4. Selecting one of the documents I found during the query use-case in the previous subsection (measurements obtained in the Norwegian sea), the systems retrieves and visualizes the four most similar documents to this query. Figure 6.3 shows a screen-shot of this approach in action. The result-set is visualized by a single scatter-plot-matrix that contains all scatter-plots of all retrieved documents. Each individual scatter-plot is rendered using shadow-drawing. This means that the opaque data-points are present in every document of the result-set, while the translucent data-points are only present in one or a few of the documents. Additionally, the geo-location where the documents were originally measured is visualized in a single map-view.

We can see that the four nearest neighbors to the query document all contain the same dimensions (although the search was not restricted to that), and that the scatter-plot-matrices of all retrieved documents are very similar to the scatter-plot-matrix of the query object. This indicates that all five documents describe a similar climate pattern. Furthermore, we see that all documents were obtained in the Norwegian Sea near the coast of Tromso, and as such, were indeed measured in the same maritime climate zone.

6.3 Summary

This use-case in physical oceanography showed the practical applicability and scalability of the proposed approach for retrieval in multivariate data. By exploring and searching for data-sets via specific patterns, a scientist in this domain can find measurement data that support or contradict a given hypothesis (e.g., "the relationship between water depth and water pressure is linear"). Furthermore, the proposed approach allows for nearest-neighbor retrieval of multivariate documents. This can be used for browsing and exploring a collection of multivariate data, as well as querying-by-example. To this end, scientists can provide a multivariate data-set they are interested in (e.g., because they obtained it themselves) and retrieve a ranked list of similar multivariate data-sets. In this way, it is possible to quickly find out if (and where) someone else obtained similar results before or if their results are novel.



Query:
doi:10.1594/PANGAEA.200626

More Like This:
doi:10.1594/PANGAEA.200638
doi:10.1594/PANGAEA.200640
doi:10.1594/PANGAEA.200635
doi:10.1594/PANGAEA.200994

Figure 6.3: My approach for similarity measurement of multivariate data documents in action. The scatter-plot-matrices of the query document and result documents are visualized. The query is drawn in opaque black, the retrieval results are drawn in translucent blue to show the deviation of the retrieval results to the query. One can see that all plots are very similar, indicating that all five documents describe a similar climate pattern. Additionally, all documents were measured in the Norwegian Sea near the coast of Tromso. One can continue the data exploration by selecting one of the retrieved documents as the next query object to retrieve its four nearest neighbors. Map Data is attributed to Google Maps.

7 Conclusions and Future Work

To conclude my thesis "Information Retrieval for Multivariate Research Data Repositories" I will summarize and highlight its main scientific contributions in the following section. Finally, I would like to present some preliminary ideas for future work where retrieval of research data plays an important role.

7.1 Conclusions

Scientific research and industrial applications are becoming more and more data-intensive and data-driven. Finding and subsequently analyzing relevant data is an increasingly decisive factor in these disciplines. Research data represents a valuable asset and if made accessible in a transparent and user friendly way, can improve the scientific and industrial processes as a whole. Encompassing digital library support for research data is therefore highly desirable.

In this thesis, I presented a body of work that dealt with novel methods for information retrieval for multivariate research data. In particular, I focused my studies on extending established, meta-data based retrieval approaches with techniques for content-based access to retrieve multivariate research data. My research in this area and my approach are motivated by certain information needs which cannot (or only insufficiently) be fulfilled by using textual, annotation-based access to research data. While it is perfectly viable to retrieve annotated data by searching for specific titles, authors, geo-locations or experimental parameters, it is not viable to search for data *patterns* themselves: Consider an information seeker looking for data documents that are *similar* to an example data document she measured herself, or an information seeker looking for a document that contains one or more specific data patterns. These are the driving challenges motivating my approach and in this conclusion I want to summarize the contributions I made towards solving these challenges and discuss the advantages and benefits resulting from my research.

1st Contribution To extend annotation-based access to research to allow for content-based querying, my first major contribution was to develop and present an approach that extracts descriptors from multivariate data documents. To achieve that, I proposed the extraction of bivariate feature vectors from each pair-wise combination of the columns in the multivariate data as a first step. This was motivated by the widely-used scatter-plot-matrix – a visual analytics tool that plots each pair-wise combination of columns to enable the analysis of multivariate data by users. This first step allowed the retrieval of multivariate documents, by extracting the set of all bivariate feature vectors from each of them. That way, I enabled an information seeker to specify one or more bivariate patterns to search for.

2nd Contribution I conducted research towards answering the query I presented as a motivational example: Given a complete multivariate data document as an example query, retrieve its nearest-neighbors from a collection of such documents. To that end, my second major contribution consists of developing a novel measure for computing the similarity between multivariate research data documents based on topic modeling. By learning a specialized topic model for research data based on the documents' bag-of-words representation (see 4th contribution), it was possible to infer a compact vector consisting of the topic activations for each document. By computing the similarity between these topic activation, nearest-neighbor retrieval as well as distance-based indexing of multivariate data documents was enabled.

3rd Contribution I constructed a benchmark for retrieval in bivariate and multivariate data collections. For bivariate data, I extensively evaluated nine different feature extraction algorithms. The method for benchmark-construction was based on meta-data annotations by domain experts. I used publicly available earth observation data for this purpose, by defining similarity classes based on annotations including type, location and time of measurement. Several of the evaluated feature extraction approaches were novel techniques developed by myself for this thesis. In particular, I developed the feature extraction technique which yielded the best retrieval results. The idea of that technique is to first estimate the density of bivariate data using a Gaussian kernel and then to extract an edge-histogram descriptor from the rendering of the resulting density estimate. For quantitatively evaluating retrieval in multivariate data, I compared my topic modeling based approach to a baseline technique which would extract features from multivariate data documents based on meta-data alone. My approach significantly outperformed this baseline technique.

4th Contribution After enabling bivariate data retrieval and multivariate data retrieval by comparing sets of feature vectors, a major challenge that was not addressed so far, were visual-interactive tools that would enable and support information seekers in formulating such queries. So as a fourth major contribution, I developed such visual-interactive tools based on a novel approach for content-based indexing of multivariate data by using a bag-of-words approach. The basic idea was to quantize the extracted bivariate feature patterns, and thus obtain a bag-of-words for each document which can be efficiently indexed using inverted lists. This efficient indexing technique allowed me to develop visual-interactive query modalities that were not available for this kind of document before. In particular and most importantly, I provided the information seeker with result highlighting and search suggestions. This provided advantages similar to such techniques for retrieval in the textual domain which information seekers are used to. Result highlighting – filtering and highlighting the scatter-plots that partially matched the user's query – explains why a document was retrieved. Search suggestions provide an overview of scatter-plot patterns to search for and refine a query with. This speeds up the retrieval process and reduces zero-result queries. I showed the applicability and scalability of my approach by indexing the complete collection of multivariate research data that is publicly available from a data library for the environmental sciences.

Summary My research in the scope of this thesis was concerned with the challenge of retrieving relevant documents from a collection of multivariate data documents. To improve upon the state-of-the-art, which mostly relied on annotation-based access, I developed and evaluated novel content-based retrieval techniques for multivariate research data repositories. My contributions extend the accessibility to such data repositories and improve the supply of information in the area of research data. To conclude the main part of this thesis, I would like to answer the question I posed as a challenge in the very beginning according to my results:

"In a collection of multivariate data documents, how can I find the documents I am looking for?"
"By being enabled to query and explore for data patterns on top of meta-data annotations."

7.2 Future Work

For quite some time, research *papers* have included a lot more than mere text written on paper. Scientific work includes empirical studies, measurement data, images, audio, video, plots and renderings, mathematical expressions and of course the full-text itself. Along with this content, papers are usually annotated for indexing with controlled meta-data like authors, keywords, topic classification and publication year and venue. Today, as all of that content is either digitized (empirical data, e.g. in social science studies comes to mind) or *digitally born* (like digital images, Word or TeX texts, mathematical expressions, measurement data), a more encompassing definition of research paper might be *digital research object*.

Bechhofer et al. [BDRG*10] define research objects as "self-contained units of knowledge" that support "publication, sharing and reuse". In particular, research objects encompass "semantically rich aggregations of resources, that can possess some scientific intent or support some research objective".

In an interesting outlook, David De Roure forecasts the "demise of the paper" [DR13]. He outlines several reasons why a paper will not be the suitable medium for scientific work in the future. These encompass the inability to include the evidence in the paper, the inability to reconstruct an experiment or re-obtain the results based on the paper alone and that any research records need to be machine readable to support automation and curation.

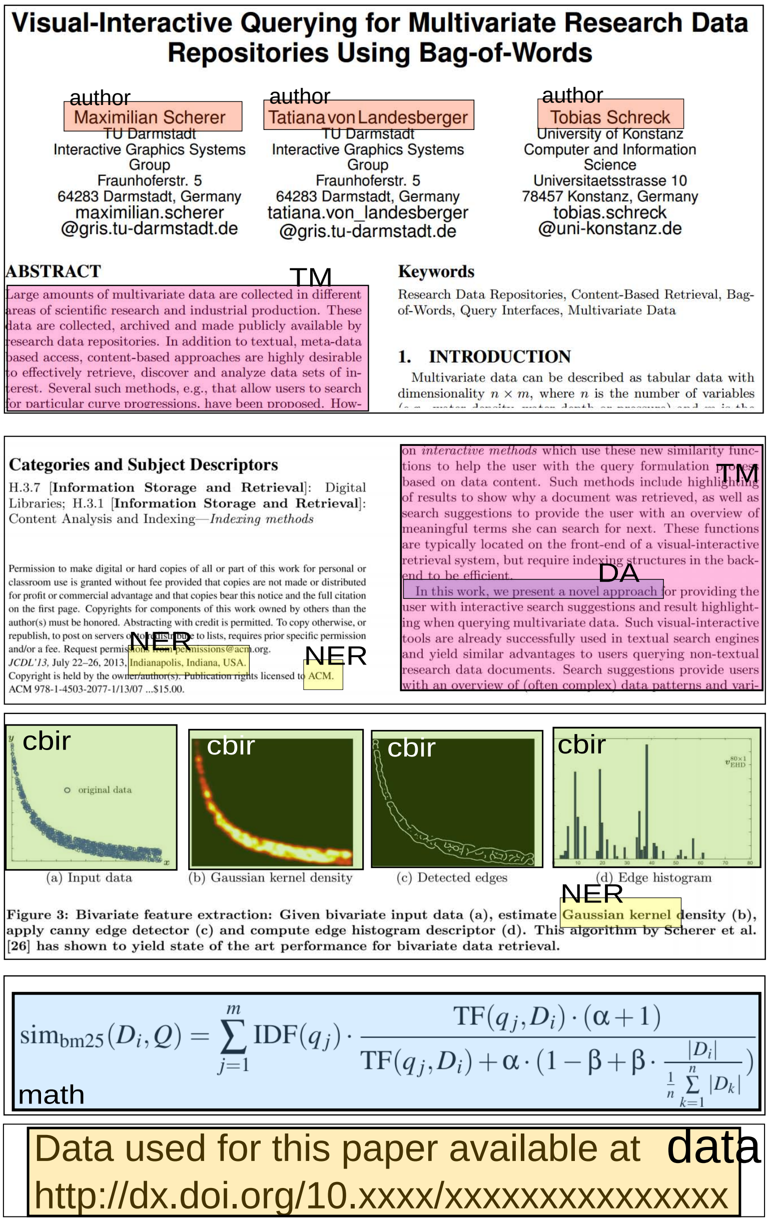
Recent advances in information retrieval for data domains like images, audio, video, math, meta-data and natural language texts together with the body of work about retrieval for multivariate research data presented in this thesis, will allow us to further develop the concept and use of the digital research object to tackle some of the future challenges outlined above. In particular these advances in information retrieval can support us in making research records machine readable for automatic indexing. Furthermore they support the re-useability and verifiability of research records by cross-linking and inter-linking scientific texts, data-sets, scientists, meta-data, multimedia content and mathematical expressions.

7.2.1 Information Retrieval for Digital Research Objects

Computing content-based descriptors from all the components a digital research objects consists of, allows us to index and relate research records well beyond established techniques like co-author and citation networks, or mere keyword or full-text search in the research object's textual components. Figure 7.1 outlines my first ideas how we can leverage the information contained in a research object's text, images, meta-data and associated data-sets.

We can analyze the full-text with techniques like topic modeling [BNJ03], named entity recognition [NS07] or discourse analysis [JM09], to automatically extract useful information for curation and indexing of a research record. Using algorithms from content-based image retrieval, we can extract descriptors from all the images contained in a research object [DJLW08]. This would allow to automatically compute relationships between papers, if they contain similar figures. In a similar vein, we can extract descriptors from the mathematical expressions found in more technical papers [YA09,HGL*13]. Again this enables us to mine for similarities between papers based on similar or equivalent mathematical expressions they contain. Finally the techniques I proposed in this thesis for analysis of multivariate data, along with related techniques for time-series data or categorical data, can be used to extract descriptors from the data-sets a research object uses. This allows for retrieving relations between scientific work, based on the similarities between data-sets they use or obtain. This is particular important for data intensive sciences like earth observation or biology, where lots of new experiments are conducted, but finding related work with similar experiments or with similar results is hard without appropriate data mining techniques.

The analysis of citation and co-author graphs also greatly benefits from the information that can be mined from digital research objects [SK13]. Analyzing citation and co-author connections between papers with respect to their actual content has already proven fruitful in identifying and forecasting important and influential literature [YHT*12]. A similar approach was taken by Gollapalli et al. to define a similarity measure between different researchers [GMG12]. Such first results show the potential of combining network-based and content-based descriptors. Encompassing, in-depth descriptors of scientific work as proposed here, will help improving the discovery and semi-automatic triage of scientific literature with respect to a user's information need.



Visual-Interactive Querying for Multivariate Research Data Repositories Using Bag-of-Words

Maximilian Scherer
TU Darmstadt
Interactive Graphics Systems Group
Fraunhoferstr. 5
64283 Darmstadt, Germany
maximilian.scherer
@gris.tu-darmstadt.de

Tatiana von Landesberger
TU Darmstadt
Interactive Graphics Systems Group
Fraunhoferstr. 5
64283 Darmstadt, Germany
tatiana.von_landesberger
@gris.tu-darmstadt.de

Tobias Schreck
University of Konstanz
Computer and Information Science
Universitaetsstrasse 10
78457 Konstanz, Germany
tobias.schreck
@uni-konstanz.de

ABSTRACT

Large amounts of multivariate data are collected in different areas of scientific research and industrial production. These data are collected, archived and made publicly available by research data repositories. In addition to textual, meta-data based access, content-based approaches are highly desirable to effectively retrieve, discover and analyze data sets of interest. Several such methods, e.g., that allow users to search for particular curve progressions, have been proposed. How-

Keywords

Research Data Repositories, Content-Based Retrieval, Bag-of-Words, Query Interfaces, Multivariate Data

1. INTRODUCTION

Multivariate data can be described as tabular data with dimensionality $n \times m$, where n is the number of variables

Categories and Subject Descriptors

H.3.7 [**Information Storage and Retrieval**]: Digital Libraries; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing methods*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
JCDL'13, July 22–26, 2013, Indianapolis, Indiana, USA.
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-2077-1/13/07 ...$15.00.

on *interactive methods* which use these new similarity functions to help the user with the query formulation process based on data content. Such methods include highlighting of results to show why a document was retrieved, as well as search suggestions to provide the user with an overview of meaningful terms she can search for next. These functions are typically located on the front-end of a visual-interactive retrieval system, but require indexing structures in the back-end to be efficient.

In this work, we present a novel approach for providing the user with interactive search suggestions and result highlighting when querying multivariate data. Such visual-interactive tools are already successfully used in textual search engines and yield similar advantages to users querying non-textual research data documents. Search suggestions provide users with an overview of (often complex) data patterns and vari-

(a) Input data (b) Gaussian kernel density (c) Detected edges (d) Edge histogram

Figure 3: Bivariate feature extraction: Given bivariate input data (a), estimate Gaussian kernel density (b), apply canny edge detector (c) and compute edge histogram descriptor (d). This algorithm by Scherer et al. [26] has shown to yield state of the art performance for bivariate data retrieval.

$$\mathrm{sim}_{\mathrm{bm25}}(D_i,Q) = \sum_{j=1}^{m} \mathrm{IDF}(q_j) \cdot \frac{\mathrm{TF}(q_j,D_i)\cdot(\alpha+1)}{\mathrm{TF}(q_j,D_i)+\alpha\cdot(1-\beta+\beta\cdot\frac{|D_i|}{\frac{1}{n}\sum_{k=1}^{n}|D_k|})}$$

Data used for this paper available at
http://dx.doi.org/10.xxxx/xxxxxxxxxxxxxxx

Figure 7.1: Concepts for Digital Research Objects to leverage the information contained in a research object's text, images, meta-data and associated data-sets. Techniques like topic modeling (TM), named entity recognition (NER), discourse analysis (DA), content-based image retrieval, mathematical information retrieval (math) and research data retrieval (data) can be applied here.

A Publications

A.1 Directly Related to this Thesis

This thesis is partially based on the following of my own publications (citations are included in the corresponding sections). The following publications are ordered chronologically.

1. "Retrieval and Exploratory Search in Multivariate Research Data Repositories Using Regressional Features" [SBS11]. I presented this publication at the 11th Joint Conference on Digital Libraries (JCDL) on June 16th, 2011 in Ottawa, Canada.
2. "A Benchmark for Content-Based Retrieval in Bivariate Data Collections" [SvLS12]. I presented this publication at the 2nd Conference on Theory and Practice of Digital Libraries (TPDL, formerly ECDL) on September 25th, 2012 in Paphos, Cyprus.
3. "Visual-Interactive Querying for Multivariate Research Data Repositories Using Bag-of-Words" [SvLS13b]. I presented this publication at the 13th Joint Conference on Digital Libraries (JCDL) on July 24th, 2013 in Indianapolis, Indiana, USA.
4. "Topic Modeling for Search and Exploration in Multivariate Research Data Repositories" [SvLS13a]. I presented this publication at the 3rd Conference on Theory and Practice of Digital Libraries (TPDL, formerly ECDL) on September 24th, 2013 in Valletta, Malta.

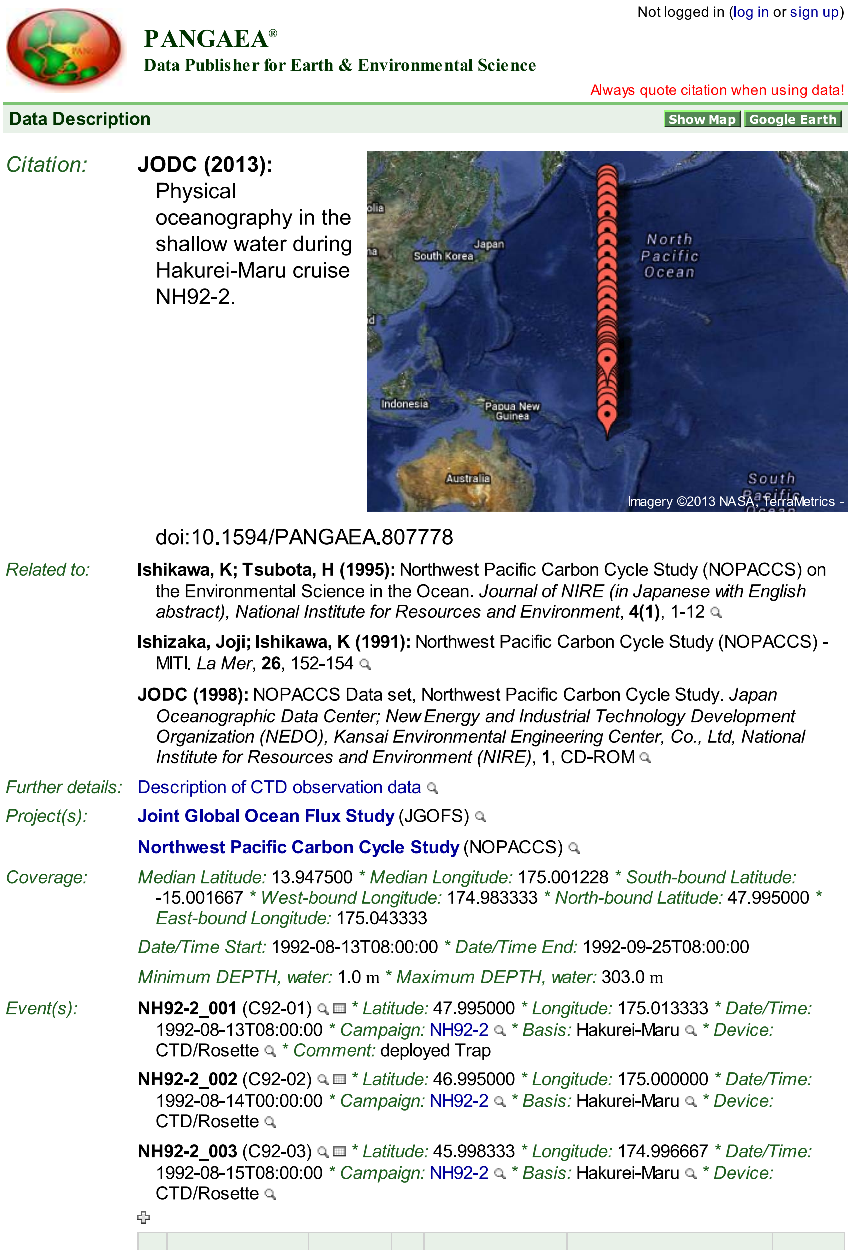
A.2 Not Directly Related to this Thesis

The following is a list of publications I have co-authored that are not directly related to this thesis. These publications are also listed in chronological order.

1. "Running on Optical Rails: Theory, Implementation and Testing of Omnidirectional View-based Point-To-Point Navigation" [DFL*08]. This was published while I was writing my diploma thesis at Goethe-University Frankfurt.
2. "Histograms of Oriented Gradients for 3D Object Retrieval" [SWS10]. I presented this publication at the 18th International Conference on Computer Graphics, Visualization and Computer Vision (WSCG) on February 3rd, 2010 in Pilsen, Czech Republic.
3. "The PROBADO Project - Approach and Lessons Learned in Building a Digital Library System for Heterogeneous Non-textual Documents" [BBC*10]. I presented this publication at the 14th European Conference on Research and Advanced Technology for Digital Libraries (ECDL) on September 9th, 2010 in Glasgow, UK.

4. "Sketch-Based 3D Model Retrieval Using Diffusion Tensor Fields of Suggestive Contours" [YSSK10]. I presented this publication at the 18th International Conference on Multimedia (ACM MM) on October 26th, 2010 in Florence, Italy.
5. "Graph-based combinations of fragment descriptors for improved 3D Object Retrieval" [SSW*12].
6. "STELA: Sketch-Based 3D Model Retrieval Using a Structure-Based Local Approach" [SBSS11].
7. "Sketch-Based 3D Model Retrieval Using Keyshapes for Global and Local Representation" [SBS*12].
8. "TimeSeriesPaths: Projection-Based Explorative Analysis of Multivariate Time Series Data" [BWS*12].
9. "Guided Discovery of Interesting Relationships Between Time Series Clusters and Metadata Properties" [BRS*12b].
10. "Content-Based Layouts for Exploratory Metadata Search in Scientific Research Data" [BRS*12a].

B Example: Multivariate Research Data



Not logged in (log in or sign up)

PANGAEA®
Data Publisher for Earth & Environmental Science

Always quote citation when using data!

Data Description Show Map Google Earth

Citation: **JODC (2013):** Physical oceanography in the shallow water during Hakurei-Maru cruise NH92-2.

doi:10.1594/PANGAEA.807778

Related to: **Ishikawa, K; Tsubota, H (1995):** Northwest Pacific Carbon Cycle Study (NOPACCS) on the Environmental Science in the Ocean. *Journal of NIRE (in Japanese with English abstract), National Institute for Resources and Environment*, **4(1)**, 1-12

Ishizaka, Joji; Ishikawa, K (1991): Northwest Pacific Carbon Cycle Study (NOPACCS) - MITI. *La Mer*, **26**, 152-154

JODC (1998): NOPACCS Data set, Northwest Pacific Carbon Cycle Study. *Japan Oceanographic Data Center; New Energy and Industrial Technology Development Organization (NEDO), Kansai Environmental Engineering Center, Co., Ltd, National Institute for Resources and Environment (NIRE)*, **1**, CD-ROM

Further details: Description of CTD observation data

Project(s): **Joint Global Ocean Flux Study** (JGOFS)

Northwest Pacific Carbon Cycle Study (NOPACCS)

Coverage: *Median Latitude:* 13.947500 * *Median Longitude:* 175.001228 * *South-bound Latitude:* -15.001667 * *West-bound Longitude:* 174.983333 * *North-bound Latitude:* 47.995000 * *East-bound Longitude:* 175.043333

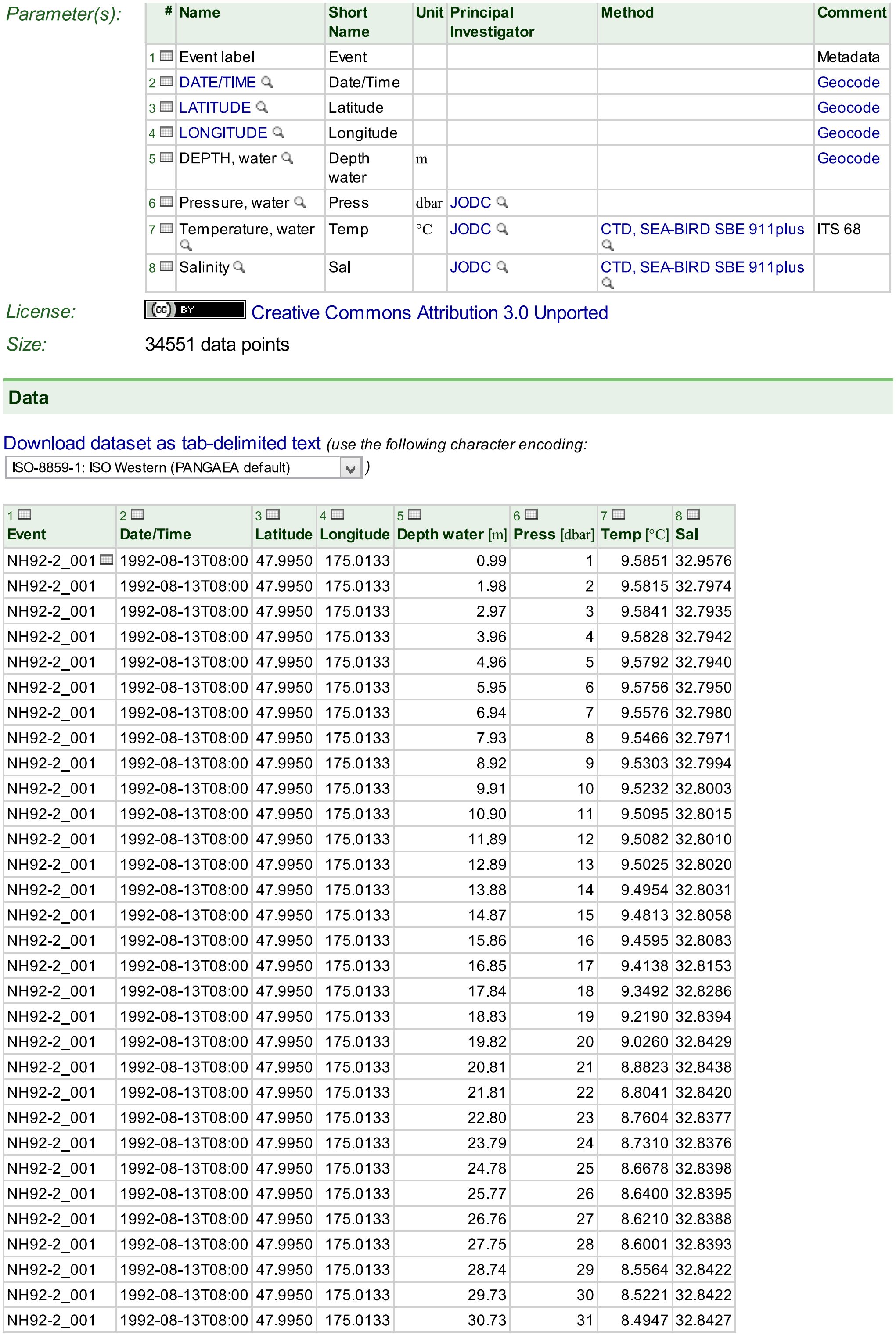
Date/Time Start: 1992-08-13T08:00:00 * *Date/Time End:* 1992-09-25T08:00:00

Minimum DEPTH, water: 1.0 m * *Maximum DEPTH, water:* 303.0 m

Event(s): **NH92-2_001** (C92-01) * *Latitude:* 47.995000 * *Longitude:* 175.013333 * *Date/Time:* 1992-08-13T08:00:00 * *Campaign:* NH92-2 * *Basis:* Hakurei-Maru * *Device:* CTD/Rosette * *Comment:* deployed Trap

NH92-2_002 (C92-02) * *Latitude:* 46.995000 * *Longitude:* 175.000000 * *Date/Time:* 1992-08-14T00:00:00 * *Campaign:* NH92-2 * *Basis:* Hakurei-Maru * *Device:* CTD/Rosette

NH92-2_003 (C92-03) * *Latitude:* 45.998333 * *Longitude:* 174.996667 * *Date/Time:* 1992-08-15T08:00:00 * *Campaign:* NH92-2 * *Basis:* Hakurei-Maru * *Device:* CTD/Rosette



Parameter(s):

#	Name	Short Name	Unit	Principal Investigator	Method	Comment
1	Event label	Event				Metadata
2	DATE/TIME	Date/Time				Geocode
3	LATITUDE	Latitude				Geocode
4	LONGITUDE	Longitude				Geocode
5	DEPTH, water	Depth water	m			Geocode
6	Pressure, water	Press	dbar	JODC		
7	Temperature, water	Temp	°C	JODC	CTD, SEA-BIRD SBE 911plus	ITS 68
8	Salinity	Sal		JODC	CTD, SEA-BIRD SBE 911plus	

License: (cc) BY Creative Commons Attribution 3.0 Unported

Size: 34551 data points

Data

Download dataset as tab-delimited text *(use the following character encoding:* ISO-8859-1: ISO Western (PANGAEA default) *)*

1 Event	2 Date/Time	3 Latitude	4 Longitude	5 Depth water [m]	6 Press [dbar]	7 Temp [°C]	8 Sal
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	0.99	1	9.5851	32.9576
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	1.98	2	9.5815	32.7974
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	2.97	3	9.5841	32.7935
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	3.96	4	9.5828	32.7942
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	4.96	5	9.5792	32.7940
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	5.95	6	9.5756	32.7950
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	6.94	7	9.5576	32.7980
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	7.93	8	9.5466	32.7971
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	8.92	9	9.5303	32.7994
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	9.91	10	9.5232	32.8003
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	10.90	11	9.5095	32.8015
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	11.89	12	9.5082	32.8010
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	12.89	13	9.5025	32.8020
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	13.88	14	9.4954	32.8031
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	14.87	15	9.4813	32.8058
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	15.86	16	9.4595	32.8083
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	16.85	17	9.4138	32.8153
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	17.84	18	9.3492	32.8286
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	18.83	19	9.2190	32.8394
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	19.82	20	9.0260	32.8429
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	20.81	21	8.8823	32.8438
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	21.81	22	8.8041	32.8420
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	22.80	23	8.7604	32.8377
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	23.79	24	8.7310	32.8376
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	24.78	25	8.6678	32.8398
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	25.77	26	8.6400	32.8395
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	26.76	27	8.6210	32.8388
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	27.75	28	8.6001	32.8393
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	28.74	29	8.5564	32.8422
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	29.73	30	8.5221	32.8422
NH92-2_001	1992-08-13T08:00	47.9950	175.0133	30.73	31	8.4947	32.8427

[ITI13]

C Curriculum Vitæ

Personal Data

Name	Maximilian Scherer
Birth Date	June 19th, 1984 in Offenbach, Germany
Family status	Unmarried
Nationality	German

Education

2009 – today PhD Student at Interactive Graphics Systems Group, Technische Universität Darmstadt, Germany, Focus: Information Retrieval for Multivariate Data

2003 – 2008 Study in computer science and graduation (diploma) at Goethe University in Frankfurt am Main, Germany, Focus: Image Processing

Grants & Scholarships

2012	SoftwareCampus "dataVIAS"
2011	EXIST Gründerstipendium "Subares"

Work Experience

2012 – today Co-Founder of Subares GmbH, EXIST spin-off from TU Darmstadt

2009 – today Researcher, Interactive Graphics Systems Group, Technische Universität Darmstadt, Germany.

2005 – 2009 Research Assistant, VSI, Goethe University Frankfurt, Germany

Darmstadt, 30.09.2013

List of Figures

Bibliography

[AFS93] AGRAWAL R., FALOUTSOS C., SWAMI A.: Efficient similarity search in sequence databases. In *Foundations of Data Organization and Algorithms*, Lomet D., (Ed.), vol. 730 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 1993, pp. 69–84. 17, 19, 33

[AMST11] AIGNER W., MIKSCH S., SCHUMANN H., TOMINSKI C.: *Visualization of Time-Oriented Data*. Springer-Verlag New York Inc, 2011. 19

[BBC*10] BERNDT R., BLÜMEL I., CLAUSEN M., DAMM D., DIET J., FELLNER D. W., FREMEREY C., KLEIN R., KRAHL F., SCHERER M., SCHRECK T., SENS I., THOMAS V.: The probado project - approach and lessons learned in building a digital library system for heterogeneous non-textual documents. In *14th European Conference on Digital Libraries - ECDL* (2010), pp. 376–383. 21, 85

[BBF*10] BERNARD J., BRASE J., FELLNER D. W., KOEPLER O., KOHLHAMMER J., RUPPERT T., SCHRECK T., SENS I.: A visual digital library approach for time-oriented scientific primary data. *Int. J. on Digital Libraries 11*, 2 (2010), 111–123. 21, 73

[BDRG*10] BECHHOFER S., DE ROURE D., GAMBLE M., GOBLE C., BUCHAN I.: Research objects: Towards exchange and reuse of digital knowledge. *Nature Precedings* (2010). 81

[Ber06] BERKHIN P.: A survey of clustering data mining techniques. *Grouping Multidimensional Data* (2006), 25–71. 8

[BGK10] BOTEV Z., GROTOWSKI J., KROESE D.: Kernel density estimation via diffusion. *Annals of Statistics 38*, 5 (2010), 2916–2957. 34, 36

[BLBS11] BERNARD J., LANDESBERGER T. V., BREMM S., SCHRECK T.: Multi-scale visual quality assessment for cluster analysis with self-organizing maps. In *Visualization and Data Analysis 2011* (2011), Proceedings of SPIE; 7868, The Society for Imaging Science and Technology (IS&T) and The International Society for Optical Engineering (SPIE), SPIE Press, Bellingham, pp. 78680N–1–78680N–12. 9

[Ble12] BLEI D. M.: Probabilistic topic models. *Commun. ACM 55*, 4 (Apr. 2012), 77–84. 9, 10

[BM04] BERRY L., MUNZNER T.: Binx: Dynamic exploration of time series datasets across aggregation levels. In *Proceedings of the IEEE Symposium on Information Visualization, Poster Compendium* (Washington, DC, USA, 2004), INFOVIS ’04, IEEE Computer Society, pp. 5–6. 19

[BNJ03] Blei D., Ng A., Jordan M.: Latent dirichlet allocation. *the Journal of machine Learning research 3* (2003), 993–1022. 9, 40, 82

[BRS*12a] Bernard J., Ruppert T., Scherer M., Kohlhammer J., Schreck T.: Content-based layouts for exploratory metadata search in scientific research data. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries* (New York, NY, USA, 2012), JCDL '12, ACM, pp. 139–148. 19, 21, 86

[BRS*12b] Bernard J., Ruppert T., Scherer M., Schreck T., Kohlhammer J.: Guided discovery of interesting relationships between time series clusters and metadata properties. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies* (New York, NY, USA, 2012), i-KNOW '12, ACM, pp. 22:1–22:8. 86

[BWS*12] Bernard J., Wilhelm N., Scherer M., May T., Schreck T.: Timeseriespaths: Projection-based explorative analysis of multivariate time series data. *Journal of WSCG 20*, 2 (2012), 97–106. 19, 86

[BYRN99] Baeza-Yates R. A., Ribeiro-Neto B.: *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999. 11

[Cha07] Cha S.-H.: Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences 1*, 4 (2007), 300–307. 7

[Cle85] Cleveland W. S.: *The Elements of Graphing Data*. Hobart Press, 1985. 19, 28

[CM84] Cleveland W. S., McGill R.: The many faces of a scatterplot. *Journal of the American Statistical Association 79*, 388 (1984), 807–822. 26

[CMV*12] Cook R., Michener W. K., Vieglais D. A., Budden A. E., Koskela R. J.: Dataone: A distributed environmental and earth science data network supporting the full data life cycle. In *EGU General Assembly Conference Abstracts* (2012), Abbasi A., Giesen N., (Eds.), EGU General Assembly Conference Abstracts. 21

[CP02] Castelli D., Pagano P.: Opendlib: A digital library service system. In *Research and Advanced Technology for Digital Libraries* (2002), Agosti M., Thanos C., (Eds.), vol. 2458 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 292–308. 19

[Cre] Creative Commons – Attribution-ShareAlike 3.0 Unported:. http://creativecommons.org/licenses/by-sa/3.0/. 40

[Dat] DataONE – Data Observation Network for Earth:. http://www.dataone.org/. 20, 47, 73

[DFL*08] Dederscheck D., Friedrich H., Lenhart C., Penc J., Rosert E., Scherer M., Mester R.: Running on Optical Rails: Theory, Implementation and Testing of Omnidirectional View-based Point-To-Point Navigation. In *ICCV 2008 - The 8th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras - OMNIVIS*

(2008). 85

[DGR*02] DIEPENBROEK M., GROBE H., REINKE M., SCHINDLER U., SCHLITZER R., SIEGER R., WEFER G.: Pangaea–an information system for environmental sciences. *Computers & Geosciences 28*, 10 (2002), 1201–1210. 48, 73

[DJLW08] DATTA R., JOSHI D., LI J., WANG J. Z.: Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv. 40*, 2 (May 2008), 5:1–5:60. 14, 16, 21, 82

[DKN08] DESELAERS T., KEYSERS D., NEY H.: Features for image retrieval: an experimental comparison. *Information Retrieval 11*, 2 (2008), 77–107. 16

[DLW05] DATTA R., LI J., WANG J. Z.: Content-based image retrieval: approaches and trends of the new age. In *In Proceedings ACM International Workshop on Multimedia Information Retrieval* (2005), ACM Press, pp. 253–262. 15

[DR13] DE ROURE D.: Pages of history. *Beyond the PDF 2. Amsterdam, Netherlands* (2013). 81

[Dry] DRYAD DIGITAL REPOSITORY FOR DATA UNDERLYING PUBLISHED WORKS:. http://www.datadryad.org/. 20, 21

[DTS*08] DING H., TRAJCEVSKI G., SCHEUERMANN P., WANG X., KEOGH E.: Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment 1*, 2 (2008), 1542–1552. 18

[EHBA10] EITZ M., HILDEBRAND K., BOUBEKEUR T., ALEXA M.: An evaluation of descriptors for large-scale image retrieval from sketched feature lines. *Computers & Graphics 34*, 5 (2010), 482 – 498. 16, 35

[ELI] ELIXIR EUROPEAN LIFE SCIENCES INFRASTRUCTURE FOR BIOLOGICAL INFORMATION:. http://www.elixir-europe.org/. 21

[ERB*12] EITZ M., RICHTER R., BOUBEKEUR T., HILDEBRAND K., ALEXA M.: Sketch-based shape retrieval. *ACM Trans. Graph. (Proc. SIGGRAPH) 31*, 4 (2012), 31:1–31:10. 11, 16, 17

[FLA] FLANN – FAST LIBRARY FOR APPROXIMATE NEAREST NEIGHBORS:. http://www.cs.ubc.ca/ mariusm/index.php/FLANN/FLANN. 45

[FMM*06] FOX P., MCGUINNESS D., MIDDLETON D., CINQUINI L., DARNELL J. A., GARCIA J., WEST P., BENEDICT J., SOLOMON S.: Semantically-enabled large-scale science data repositories. In *The Semantic Web-ISWC 2006*. Springer, 2006, pp. 792–805. 21

[GEOa] GEO – GROUP ON EARTH OBSERVATIONS:. http://www.earthobservations.org/index.shtml. 22

[GEOb] GEO – GROUP ON EARTH OBSERVATIONS: GEOSS 10-Year Implementation Plan. http://www.earthobservations.org/documents/10-Year accessed online 20.08.2013. 23

[GMG12] GOLLAPALLI S. D., MITRA P., GILES C. L.: Similar researcher search in academic environments. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries* (New York, NY, USA, 2012), JCDL '12, ACM, pp. 167–170. 82

[GWCS09] GREENBERG J., WHITE H. C., CARRIER S., SCHERLE R.: A metadata best practice for a scientific data repository. *Journal of Library Metadata 9*, 3-4 (2009), 194–212. 21

[Hea09] HEARST M. A.: *Search User Interfaces*, 1 ed. Cambridge University Press, 2009. 19

[HGL*13] HU X., GAO L., LIN X., TANG Z., LIN X., BAKER J. B.: Wikimirs: a mathematical information retrieval system for wikipedia. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries* (New York, NY, USA, 2013), JCDL '13, ACM, pp. 11–20. 82

[HXL*11] HU W., XIE N., LI L., ZENG X., MAYBANK S.: A survey on visual content-based video indexing and retrieval. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 41*, 6 (2011), 797–819. 14

[Ima] IMAGE CLEF:. http://www.imageclef.org/. 16

[ITI13] ISHIKAWA K., TSUBOTA H., ISHIZAKA J.: Physical oceanography in the shallow water during hakurei-maru cruise nh92-2. In *Publishing Network for Geoscientific & Environmental Data, JODC*. http://dx.doi.org/10.1594/PANGAEA.807778, 2013. 88

[JDS10] JÉGOU H., DOUZE M., SCHMID C.: Improving bag-of-features for large scale image search. *International Journal of Computer Vision 87*, 3 (2010), 316–336. 10, 16

[JM09] JURAFSKY D., MARTIN J. H.: Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition (2nd ed). 82

[JMF99] JAIN A., MURTY M., FLYNN P.: Data clustering: a review. *ACM computing surveys (CSUR) 31*, 3 (1999), 264–323. 8

[KAF*08] KEIM D., ANDRIENKO G., FEKETE J.-D., GÖRG C., KOHLHAMMER J., MELANÇON G.: *Visual Analytics: Definition, Process, and Challenges*, vol. 4950 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2008. 18

[KCPM01] KEOGH E., CHAKRABARTI K., PAZZANI M., MEHROTRA S.: Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and Information Systems 3*, 3 (2001), 263–286. 17, 19, 33

[KH12] KHOO M., HALL C.: What would 'google' do? users' mental models of a digital library search engine. In *Theory and Practice of Digital Libraries* (2012), Zaphiris P., Buchanan G., Rasmussen E., Loizides F., (Eds.), vol. 7489 of *Lecture Notes in Computer Science*, Springer, pp. 1–12. 65, 69

[KK03] KEOGH E., KASETTY S.: On the need for time series data mining benchmarks: A survey and empirical demonstration. *Data Mining and Knowledge Discovery 7*, 4 (2003), 349–371. 18

[KLF05] KEOGH E., LIN J., FU A.: Hot sax: Efficiently finding the most unusual time series subsequence. In *IEEE International Conference on Data Mining* (2005), pp. 226–233. 17, 19, 33

[KNS*00] Kosugi N., Nishihara Y., Sakata T., Yamamuro M., Kushima K.: A practical query-by-humming system for a large music database. In *Proceedings of the eighth ACM international conference on Multimedia* (New York, NY, USA, 2000), MULTIMEDIA '00, ACM, pp. 333–342. 16

[Koh82] Kohonen T.: Self-organized formation of topologically correct feature maps. *Biological Cybernetics 43*, 1 (1982), 59–69. 8

[KS02] Kasabov N., Song Q.: Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *Fuzzy Systems, IEEE Transactions on 10*, 2 (2002), 144–154. 19

[KTB12] Köhncke B., Tönnies S., Balke W.-T.: Catching the drift – indexing implicit knowledge in chemical digital libraries. In *Theory and Practice of Digital Libraries* (2012), Zaphiris P., Buchanan G., Rasmussen E., Loizides F., (Eds.), vol. 7489 of *Lecture Notes in Computer Science*, Springer, pp. 383–395. 21

[Kum10] Kumar V.: Discovery of patterns in global earth science data using data mining. In *Advances in Knowledge Discovery and Data Mining*, Zaki M., Yu J., Ravindran B., Pudi V., (Eds.), vol. 6118 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 2–2. 23

[KZH*11] Keogh E., Zhu Q., Hu B., Hao Y., Xi X., Wei L., Ratanamahatana C.: The ucr time series classification/clustering homepage. http://www.cs.ucr.edu/ eamonn/time_series_data/, 2011. 18

[LCSS98] La Cascia M., Sethi S., Sclaroff S.: Combining textual and visual cues for content-based image retrieval on the world wide web. In *Content-Based Access of Image and Video Libraries, 1998. Proceedings. IEEE Workshop on* (1998), IEEE, pp. 24–28. 16

[LJW*07] Lv Q., Josephson W., Wang Z., Charikar M., Li K.: Multi-probe lsh: efficient indexing for high-dimensional similarity search. In *Proceedings of the 33rd international conference on Very large data bases* (2007), VLDB '07, VLDB Endowment, pp. 950–961. 16

[LKL11] Lin J., Khade R., Li Y.: Rotation-invariant similarity in time series using bag-of-patterns representation. *J. of Intelligent Information Systems* (2011), 1–29. 11, 17, 18

[LKLC03] Lin J., Keogh E., Lonardi S., Chiu B.: A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery* (New York, NY, USA, 2003), DMKD '03, ACM, pp. 2–11. 17

[LLE00] Latecki L. J., Lakämper R., Eckhardt U.: Shape descriptors for non-rigid shapes with a single closed contour. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (2000), pp. 424–429. 16

[Low04] Lowe D. G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision 60*, 2 (Nov. 2004), 91–110. 10

[LPSW06] LAGOZE C., PAYETTE S., SHIN E., WILPER C.: Fedora: an architecture for complex objects and their relationships. *Int. J. Digit. Libr. 6* (2006), 124–138. 19

[LSDJ06] LEW M., SEBE N., DJERABA C., JAIN R.: Content-based multimedia information retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP) 2*, 1 (2006), 1–19. 10, 14, 16

[LW08] LI J., WANG J. Z.: Real-time computerized annotation of pictures. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 30*, 6 (2008), 985–1002. 16

[Lyn09] LYNCH C. A.: Jim gray's fourth paradigm and the construction of the scientific record. In *The Fourth Paradigm*, Hey T., Tansley S., Tolle K. M., (Eds.). Microsoft Research, 2009, pp. 177–183. 20

[LYRL04] LEWIS D. D., YANG Y., ROSE T. G., LI F.: Rcv1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res. 5* (Dec. 2004), 361–397. 14

[Mac67] MACQUEEN J. B.: Some Methods for classification and analysis of multivariate observations. In *Procedings of the Fifth Berkeley Symposium on Math, Statistics, and Probability* (1967), vol. 1, University of California Press, pp. 281–297. 9

[MH10] MARCIAL L. H., HEMMINGER B. M.: Scientific data repositories on the web: An initial survey. *Journal of the American Society for Information Science and Technology 61*, 10 (2010), 2029–2048. 21, 22

[ML09] MUJA M., LOWE D. G.: Fast approximate nearest neighbors with automatic algorithm configuration. In *In VISAPP International Conference on Computer Vision Theory and Applications* (2009), pp. 331–340. 16, 45

[MMP02] MÜLLER H., MARCH S., PUN T.: The truth about corel - evaluation in image retrieval. In *Proceedings of The Challenge of Image and Video Retrieval (CIVR)* (2002), pp. 38–49. 16

[MRS08] MANNING C. D., RAGHAVAN P., SCHÜTZE H.: *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008. 11, 14, 43

[NS07] NADEAU D., SEKINE S.: A survey of named entity recognition and classification. *Lingvisticae Investigationes 30*, 1 (2007), 3–26. 82

[NTC] NTCIR – EVALUATION OF INFORMATION ACCESS TECHNOLOGIES:. http://research.nii.ac.jp/ntcir/. 14

[ORN] ORNL DAAC:. http://daac.ornl.gov/. 20

[PAN] PANGAEA – PUBLISHING NETWORK FOR GEOSCIENTIFIC & ENVIRONMENTAL DATA:. http://www.pangaea.de/. 20, 21, 47, 48, 73

[PJW00] PARK D., JEON Y., WON C.: Efficient use of local edge histogram descriptor. In *Proceedings of the 2000 ACM workshops on Multimedia* (2000), ACM, pp. 51–54. 35

[Psy] PSYCHDATA NATIONAL REPOSITORY FOR PSYCHOLOGICAL RESEARCH DATA:. http://psychdata.zpid.de/. 21

[RBPK12] ROWLEY-BROOKE R., PITIÉ F., KOKARAM A.: A ground truth bleed-through document image database. In *Theory and Practice of Digital Libraries* (2012), Zaphiris P., Buchanan G., Rasmussen E., Loizides F., (Eds.), vol. 7489 of *Lecture Notes in Computer Science*, Springer, pp. 185–196. 21

[Rei67] REINSCH C.: Smoothing by spline functions. *Numerische Mathematik 10*, 3 (1967), 177–183. 30

[RHG08] RILEY M., HEINEN E., GHOSH J.: A text retrieval approach to content-based audio retrieval. In *Int. Symp. on Music Information Retrieval (ISMIR)* (2008), pp. 295–300. 10, 16

[RS05] RAMSAY J. O., SILVERMAN B. W.: *Functional Data Analysis*, 2nd ed. Springer Series in Statistics. Springer, June 2005. 30

[RZ09] ROBERTSON S., ZARAGOZA H.: The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr. 3*, 4 (Apr. 2009), 333–389. 13

[SB88] SALTON G., BUCKLEY C.: Term-weighting approaches in automatic text retrieval. *Information processing & management 24*, 5 (1988), 513–523. 13

[SB05] SHIRAHATTI N. V., BARNARD K.: Evaluating image retrieval. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01* (Washington, DC, USA, 2005), CVPR '05, IEEE Computer Society, pp. 955–961. 11

[SBS11] SCHERER M., BERNARD J., SCHRECK T.: Retrieval and exploratory search in multivariate research data repositories using regressional features. In *Proceeding of the 11th annual international ACM/IEEE joint conference on Digital libraries* (New York, NY, USA, 2011), JCDL '11, ACM, pp. 363–372. 21, 30, 85

[SBS*12] SAAVEDRA J. M., BUSTOS B., SCHRECK T., YOON S. M., SCHERER M.: Sketch-based 3D Model Retrieval using Keyshapes for Global and Local Representation. In *Eurographics Workshop on 3D Object Retrieval* (2012), Eurographics Association, pp. 47–50. 86

[SBSS11] SAAVEDRA J. M., BUSTOS B., SCHERER M., SCHRECK T.: Stela: sketch-based 3d model retrieval using a structure-based local approach. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval* (New York, NY, USA, 2011), ICMR '11, ACM, pp. 26:1–26:8. 86

[Sch80] SCHERER MICHAEL: Einfluß der normalleitenden Matrix auf die Stromverteilung und die Stromtragfähigkeit in technischen Supraleitern. *Dissertation, Kernforschungszentrum Karlsruhe* (1980). 5

[Shn96] SHNEIDERMAN B.: The eyes have it: A task by data type taxonomy for information visualizations. In *IEEE Visual Languages* (College Park, Maryland 20742, U.S.A., 1996), no. UMCP-CSD CS-TR-3665, pp. 336–343. 18

[SHR] SHREC – SHAPE RETRIEVAL CONTEST:. http://www.aimatshape.net/event/SHREC/. 16

[Sil85] SILVERMAN B.: Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society. Series B (Methodological) 47*, 1 (1985), 1–52. 30

[SK13] SUGIYAMA K., KAN M.-Y.: Exploiting potential citation papers in scholarly paper recommendation. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries* (New York, NY, USA, 2013), JCDL '13, ACM, pp. 153–162. 82

[SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Shape Modeling Applications* (2004), IEEE, pp. 167–178. 16

[SoEO10] SECRETARIAT G., ON EARTH OBSERVATIONS G.: *Crafting Geoinformation: The Art and Science of Earth Observation*. Group on Earth Observations, 2010. 22

[SSW*12] SCHRECK T., SCHERER M., WALTER M., BUSTOS B., YOON S. M., KUIJPER A.: Graph-based combinations of fragment descriptors for improved 3d object retrieval. In *MMSys* (2012), pp. 23–28. 86

[Ste56] STEINHAUS H.: Sur la division des corps matériels en parties. *Bull. Acad. Polon. Sci. Cl. III. 4* (1956), 801–804. 9

[Ste04] STEWART R.: *Introduction to physical oceanography*. Texas A & M University, 2004. 69, 73

[SvLS12] SCHERER M., VON LANDESBERGER T., SCHRECK T.: A benchmark for content-based retrieval in bivariate data collections. In *Proceedings of the Second international conference on Theory and Practice of Digital Libraries* (Berlin, Heidelberg, 2012), TPDL'12, Springer-Verlag, pp. 286–297. 28, 36, 37, 48, 55, 85

[SvLS13a] SCHERER M., VON LANDESBERGER T., SCHRECK T.: Topic modeling for search and exploration in multivariate research data repositories. In *Proceedings of the Third international conference on Theory and Practice of Digital Libraries* (2013), TPDL'13, pp. 370–373. 40, 85

[SvLS13b] SCHERER M., VON LANDESBERGER T., SCHRECK T.: Visual-interactive querying for multivariate research data repositories using bag-of-words. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries* (New York, NY, USA, 2013), JCDL '13, ACM, pp. 285–294. 65, 85

[SWS*00] SMEULDERS A. W., WORRING M., SANTINI S., GUPTA A., JAIN R.: Content-based image retrieval at the end of the early years. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 22*, 12 (2000), 1349–1380. 15

[SWS10] SCHERER M., WALTER M., SCHRECK T.: Histograms of oriented gradients for 3d object retrieval. In *WSCG 2010, 18th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision. Full Papers Proceeding* (2010), pp. 41–48. 85

[SZ03] SIVIC J., ZISSERMAN A.: Video google: A text retrieval approach to object matching in videos. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on* (2003), IEEE, pp. 1470–1477. 10

[The] THE CLEF INITIATIVE – CONFERENCE AND LABS OF THE EVALUATION FORUM:. http://www.clef-initiative.eu. 14

[Tob70] TOBLER W.: A computer movie simulating urban growth in the detroit region. *Economic geography 46* (1970), 234–240. 50

[TPG13] TUAROB S., POUCHARD L. C., GILES C. L.: Automatic tag recommendation for metadata annotation using probabilistic topic modeling. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries* (New York, NY, USA, 2013), JCDL '13, ACM, pp. 239–248. 21, 23

[TPN*12] TUAROB S., POUCHARD L. C., NOY N., HORSBURGH J. S., PALANISAMY G.: Onemercury: Towards automatic annotation of environmental science metadata. In *Proceedings of the 2nd International Workshop on Linked Science* (2012), LISC '12. 23

[TRE] TREC – TEXT RETRIEVAL CHALLENGE:. http://trec.nist.gov. 14

[TV08] TANGELDER J. W., VELTKAMP R. C.: A survey of content based 3d shape retrieval methods. *Multimedia tools and applications 39*, 3 (2008), 441–471. 14

[TWH01] TIBSHIRANI R., WALTHER G., HASTIE T.: Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63*, 2 (2001), 411–423. 9

[TWV05] TYPKE R., WIERING F., VELTKAMP R. C.: A survey of music information retrieval systems. In *ISMIR* (2005), pp. 153–160. 14

[WAG05] WILKINSON L., ANAND A., GROSSMAN R.: Graph-theoretic scagnostics. In *IEEE Symposium on Information Visualization* (2005), pp. 21–29. 19

[WCR*11] WONG B., CHOUDHURY S., ROONEY C., CHEN R., XU K.: Invisque: technology and methodologies for interactive information visualization and analytics in large library collections. *Research and Advanced Technology for Digital Libraries* (2011), 227–235. 19

[WMBB00] WITTEN I. H., MCNAB R. J., BODDIE S. J., BAINBRIDGE D.: Greenstone: A comprehensive open-source digital library software system. In *Proceedings of the Fifth ACM International Conference on Digital Libraries* (2000), pp. 113–121. 19

[WR09] WHITE R. W., ROTH R. A.: *Exploratory Search: Beyond the Query-Response Paradigm*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers, 2009. 19

[WSJ*12] WONG P. C., SHEN H.-W., JOHNSON C., CHEN C., ROSS R. B.: The top 10 challenges in extreme-scale visual analytics. *Computer Graphics and Applications, IEEE 32*, 4 (2012), 63–67. 19

[XW*05] XU R., WUNSCH D., ET AL.: Survey of clustering algorithms. *Neural Networks, IEEE Transactions on 16*, 3 (2005), 645–678. 8, 10, 37

[YA09] YOKOI K., AIZAWA A.: An approach to similarity search for mathematical expressions using mathml. *Towards a Digital Mathematics Library. Grand Bend, Ontario, Canada, July 8-9th, 2009* (2009), 27–35. 82

[YF00] YI B., FALOUTSOS C.: Fast time sequence indexing for arbitrary Lp norms. In *Proceedings of the 26th International Conference on Very Large Data Bases* (2000), pp. 385–394. 17, 19, 33

[YHT*12] YAN R., HUANG C., TANG J., ZHANG Y., LI X.: To better stand on the shoulder of giants. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries* (New York, NY, USA, 2012), JCDL '12, ACM, pp. 51–60. 82

[YJHN07] YANG J., JIANG Y., HAUPTMANN A., NGO C.: Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the international workshop on Workshop on multimedia information retrieval* (2007), ACM, pp. 197–206. 9

[YSSK10] YOON S. M., SCHERER M., SCHRECK T., KUIJPER A.: Sketch-based 3d model retrieval using diffusion tensor fields of suggestive contours. In *Proceedings of the international conference on Multimedia* (New York, NY, USA, 2010), MM '10, ACM, pp. 193–200. 15, 16, 86