

Università degli Studi di Salerno
Dipartimento di Scienze Aziendali - Management & Innovation System

Dottorato di Ricerca in Management and Information Technology
Curriculum Informatica, Sistemi Informativi e Tecnologie del Software
XVI Ciclo



Ph.D. Thesis

Information Visualization: from Petroglyphs to CoDe Graphs

Candidate

Dr. Paola de Roberto

Coordinator

Prof. Andrea De Lucia

Tutor

Prof. Genoveffa Tortora

2016 - 2017

*To my husband
and my children*

Abstract

Data visualization concerns the communication of data through visual representations and techniques. It aims at enhancing perception and support data-driven decision making so enabling insights otherwise hard to achieve. A good visualization of data makes it possible to identify patterns and enables better understanding of phenomena. In other words, data visualization is related to an innate human ability to quickly comprehend, discern and convert patterns into useful and usable information.

Humans have used visual graphical representations as early as 35.000 B.C., through cave drawings. Indeed, human ancestors already reasoned in terms of models or schemata: the visual representation of information is an ancient concept, as witnessed by the rock carvings found. Over the centuries, information visualization has evolved to take into account the changing human needs and its use has become more and more conscious. The first data visualization techniques have been developed to observe and represent physical quantities, geography and celestial positions. Successively, the combined use of euclidean geometry and algebra improved accuracy and complexity of information representation, in different fields, such as astronomy, physics and engineering. Finally, in the last century most modern forms of data representations were invented: starting from charts, histograms, and graphs up to high dimensional data, and dynamic and interactive visualizations of temporal data [41].

Nowadays, the huge amount of information enables more precise interpretation of phenomena so fostering the adoption of infographic techniques, in particular, for supporting managerial decision-making in the business area.

Actually, data visualization represents the heart of Business Analytics, which needs to

transform a large amount of complex data into comprehensible information for the most intuitive description of phenomena.

Information visualization uses visual reports which can be considered sentences of a proper visual language in a given specific domain. Then, there exists the need to examine and to address the main issues related to visual languages and information visualization in different domains: *i)* through the analysis and the interpretation of visual information; *ii)* by investigating the use of information modeling and the representation techniques needed to design and implement User-Centered decision support systems.

In the first part of the thesis, we propose a Visual Analytics system applied to the archaeological area; this system supports rupestrian archaeologists in the analysis and classification of thousands of reliefs and artifacts; each artifact contains different engravings, often called *petroglyphs*.

Rock art is a term coined in archeology for indicating any human made markings carved on natural stone [19,65]. The most part of the symbols concerning rock art are represented by *petroglyphs*, which were created by removing part of a rock surface by incising, picking, carving, and abrading. Although it is not possible to give certain interpretations to these petroglyphs, archaeologists have proposed many theories to explain their meaning, e.g., astronomical, cultural, or religious [105]. To this aim, we presents a visual analytics system, named *DARK*, for supporting rock art archaeologists in exploring repositories of rock art scenes each consisting of hundreds of petroglyphs carved by ancient people on rocks. With their increasing complexity, analyzing these repositories of heterogeneous information has become a major task and challenge for rock art archaeologists. *DARK* combines visualization techniques with fuzzy-based analysis of rock art scenes to infer information crucial for the correct interpretation of the scenes. Moreover, the *DARK* views allow archaeologists to validate their hypothesis against the information stored in the repository. In addition, we describe and detail the main features of the *PetroSketch* mobile application for supporting archaeologists in the classification and recognition of petroglyph symbols. *PetroSketch* is a virtual notebook enabling users to draw a petroglyph symbol on a white page, or by following the contour of a symbol captured with the camera, and to obtain its classification and the list of symbols more similar to it. The latter is performed by

a flexible image matching algorithm that measures the similarity between petroglyph by using a distance, derived from the image deformation model, which is computationally efficient and robust to local distortions.

In the second part of the thesis, we focus our attention on investigating the data visualization problem in the area of Business Analytics. In this context, the visual report of information extracted from data sources and in particular from data warehouses is faced. Visualization tools play a central role in contexts where information must be represented by preserving both the accuracy of data, and the complexity of relationships between data. Much attention has been paid to the problem of effective visualization of data in individual reports that usually are viewed through different types of standard graph (histogram, bar-plots, etc.) or in a tabular form. However, this kind of representation provides separate information items, but gives no support to visualize their relationships, which are the basis for most decision processes. Indeed, decision makers spend significant time and effort interpreting graphics derived from large multidimensional databases. Then, the choice of a graphical representation is critical whenever it is necessary to interpret data. Data are usually represented by several dashboard diagrams, such as histograms and pies, which do not highlight logical relationships among them.

The CoDe (Complexity Design) language allows to organize visualizations, named CoDe Graphs, by composing and aggregating dashboard diagrams through graphical conceptual links. The CoDe-based graph composition modeling allows to visualize relationships between information in the same image following the definition of efficiency of a visualization given by Jacques Bertin [9]: *"The most efficient (graphic) construction are those in which any question, whatever its type and level, can be answered in a single instant of perception, that is, in a single image"*. This representation named CoDe model can be considered a high-level cognitive map of the complex information underlying the ground data. The choice of the final visualization layout in terms of standard graphs is left to a visualization interface which provides the necessary implementation constructs.

The application of the CoDe methodology impacts principally on the Business Analytics field where the knowledge, management and analysis of company data (e.g., sales, production, costs and profits) are fundamental requirements for a valid decision-making

system. In fact, in modern companies, strategic information is necessary to grant the solidity and survival of business companies. In addition, CoDe is an alternative graphic model for the representation of data that reduces the gap between the data warehouse expert and company management. Indeed, CoDe allows to select the information of interest, to model the relationships between them and to automatically generate the diagrams containing this information linked through the relationships identified. To this aim, we presents the generation process of CoDe Graphs, and after analyzing the state of the art concerning the evaluation of graphical representation comprehensibility, we first propose a classification of that evaluation approaches and then perform the evaluation of the comprehensibility of CoDe Graphs with respect to dashboard reports by means of a controlled experiment, involving 47 participants. Results show that CoDe Graphs reduces participants' effort, while improving effectiveness and efficiency when a comprehension task is performed, so witnessing the usefulness of the CoDe methodology.

Acknowledgements

Firstly, I would like to express my sincere gratitude to my tutor Prof. Genny Tortora for her continuous support, patience, motivation, and immense knowledge. Her guidance helped me pursue my research goals. I could not imagine having a better tutor and mentor for my Ph.D studies.

Moreover, I would like to thank Prof. Michele Risi for his precious assistance and expertise.

Last but not the least, I would like to thank my family for supporting me spiritually throughout this period of my life.

Contents

Abstract	v
Acknowledgements	ix
Contents	xi
1 Introduction	1
2 Visualization of Archaeological Information	7
2.1 Rock Art	9
2.2 Image Processing of Petroglyphs	9
3 A Visual Analytics System for Rock Art Knowledge Discovery	11
3.1 Digital Preservation of Rock Art	12
3.2 The DARK System	13
3.2.1 Structural and Semantic Abstraction through Bubble View	15
3.2.2 Scene Spatial Analysis using Ring View	16
4 Classification and Recognition of Petroglyphs	23
4.1 Rock Art	24
4.2 An Approach for Petroglyph Classification	25
4.2.1 Shape Normalization	26
4.2.2 Feature Set	26
4.2.3 Classification	27

4.2.4	Performance Optimization	28
4.3	PetroSketch	29
4.4	The Experiment	31
4.4.1	Settings	32
4.4.2	Results	32
4.4.3	Discussion	33
5	Visualizing Information in Data Warehouses Reports	35
5.1	Evaluation of Graphical Representations	37
6	Complex Graphs generated by the CoDe Methodology	43
6.1	Modeling and Generation of CoDe Graphs	43
6.1.1	Code Models and CoDe Graphs	44
6.1.2	Conceptual Organization of Report Visualization: the CoDe Paradigm	49
6.1.3	Generating CoDe Graphs	56
7	User Comprehension of CoDe Graph Reports	61
7.1	Definition and Planning of the Empirical Evaluation	61
7.1.1	Context Selection	62
7.1.2	Variable Selection	63
7.1.3	Hypothesis Formulation	65
7.1.4	Experiment Design, Material, and Procedure	66
7.1.5	Analysis Procedure	68
7.1.6	Threats to Validity	69
7.2	Results of the Empirical Evaluation	73
7.2.1	Hypotheses Testing	74
7.2.2	Analysis of Co-factors	75
7.2.3	Post-Experiment Results	75
7.2.4	Implications	76
8	Conclusions and Future Works	79

Chapter 1

Introduction

Data visualization concerns the communication of data through visual representations and techniques. It aims at enhancing perception and support data-driven decision making so enabling insights otherwise hard to achieve. A good visualization of data makes it possible to identify patterns and enables better understanding of phenomena. In other words, data visualization is related to an innate human ability to quickly comprehend, discern and convert patterns into useful and usable information.

Nowadays, the huge amount of information enables more precise interpretation of considered phenomena so fostering the adoption of infographic techniques, in particular, for supporting managerial decision-making in the business area. Moreover, data visualization represents the heart of Business Analytics, which needs to transform a large amount of complex data into comprehensible information for the most intuitive description of these phenomena.

Information visualization uses visual reports which can be considered sentences of a proper visual language in a given specific domain. Then, there exists the need to examine and to address the main issues related to visual languages and information visualization in different domains: *i)* through the analysis and the interpretation of visual information; *ii)* by investigating the use of information modeling and the representation techniques needed to design and implement User-Centered decision support systems.

In the first part of the thesis, we propose a Visual Analytics system applied to the archaeological area; this system supports rupestrian archaeologists in the analysis and

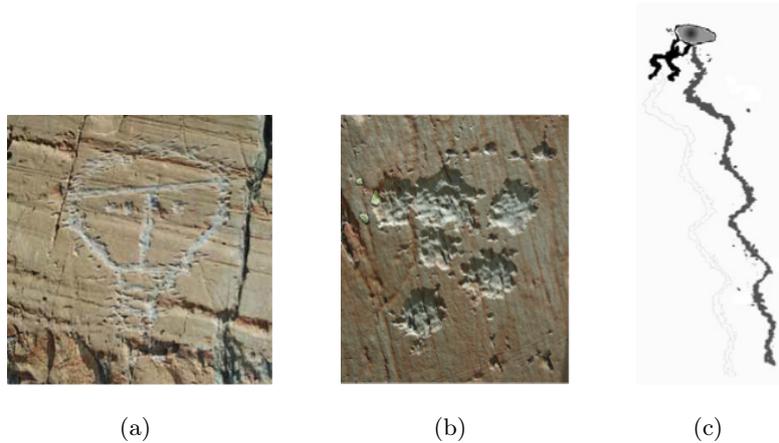


Figure 1.1: The picture of a petroglyph supposed to represent a Christ (a) [11], a picture interpreted as the stellar cluster of the Pleiades (b) [36], and the relief depicting priests making water spout from the rock (c) [27].

classification of thousands of reliefs and artifacts; each artifact contains different engravings, often called *petroglyphs*.

Rock art is a term coined in archaeology for indicating any human made markings carved on natural stone [19,65]. The most part of the symbols concerning rock art are represented by *petroglyphs*, which were created by removing part of a rock surface by incising, picking, carving, and abrading. Although it is not possible to give certain interpretations to these petroglyphs, archaeologists have proposed many theories to explain their meaning, e.g., astronomical, cultural, or religious [105]. For example, Figures 1.1(a) and 1.1(b) show the pictures of two petroglyphs interpreted as a Christ [11] and the stellar cluster of the Pleiades [36], respectively, while Figure 1.1(c) depicts a digitalized relief interpreted as a priest making water spout from the water basin [27].

Petroglyphs are engravings obtained by hitting or scraping stone surfaces with sharp tools made of stone or metal. These symbols were used by prehistoric people to communicate with their divinity or with other men. Basically, we find these carvings as a set of symbols called scenes, which depicts daily life situations. Therefore, the study of the petroglyphs is very important to gain new knowledge and awareness about the periods in which the petroglyphs were created.

Petroglyphs are prehistoric stone engravings unrevealing stories of ancient life and describing a conception of the world transmitted till today. Although they may seem

as durable as the rock they reside on, petroglyphs are inevitably deteriorated by natural causes as well as vandalism. As an example, some of them have been destroyed by tourists, while others are disappearing due to acid rain and sunshine. Hence, it is very important to preserve the petroglyphs trying to identify and archive them for future generations.

In the second part of the thesis, we focus our attention on investigating the data visualization problem in the area of Business Analytics. In this context, the visual report of information extracted from data sources and in particular from data warehouses is faced.

During the history, information visualization has evolved starting from visualization techniques developed to observe and represent physical quantities, geography and celestial positions up to the definition of dynamic and interactive techniques to visualize multi-dimensional data, abstract data, and temporal data [41]. To this aim, many techniques have been introduced and used in the centuries. The techniques, currently used, have their origin in the proposal of William Playfair. In 1786, he introduced the first line graph; since then bar, line and pie graphs (also called charts) have been largely adopted in day-by-day activities [78]. Such large use of graphical representations is motivated by the effective support they provide to a quick understanding of data and of the phenomena they describe. Indeed, they ease the communication process among managers and stakeholders, and support decision making [85].

Several visualization tools, such as Spotfire [91] and Tableau [94], support the representation of data extracted by data-warehouses through different graphs that contribute to the description of a single phenomenon. The comprehension of the phenomenon in its completeness is left to the user that has to understand it by relating the information represented in each graph. Generally, managers acquire information in the form of a digital dashboard, which displays graphs of key performance indicators on a single screen [63]; they spend significant time and effort to make decisions, by interpreting graphics derived from large multidimensional databases.

Data can be reported by using interactive tools [38, 80, 86, 94] or through graphical printed reports [1, 83, 96, 104]), where *"the static representation shows the entire story and gives context"* [37]. Static representations highlight data relationships among charts. Following Jaques Bertin, one of the most influential theorists in the field of graphic design semiotics: *"the most efficient (graphic) constructions are those in which any question, whatever its type and level, can be answered in a single instant of perception, that is, in a*

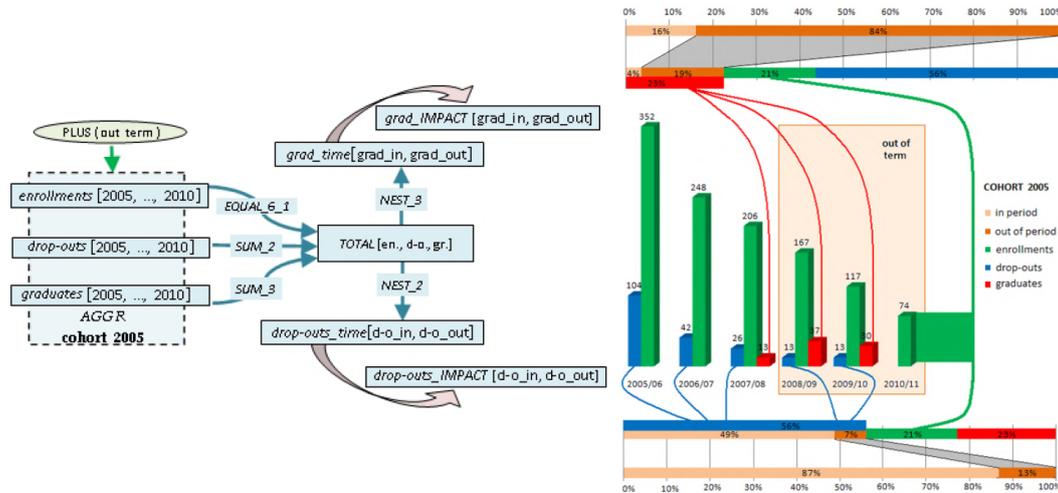


Figure 1.2: A CoDe model and the generated CoDe Graph.

single image." [10].

A graphical report could be designed based on metadata and independently of specific data instances and chart types, so that the execution of such a graphical model could generate the correct graphical instance, once specific data instances and chart types have been selected. As data change, the same graphical model can be re-executed so generating an updated report with the same semantics on the new data.

The Complexity Design (CoDe) methodology provides a technique to model graphical reports on data extracted by a data-warehouse [82,83]. This graphical model, named CoDe model, enables users to describe the semantics of a graphical representation independently of specific instances of data, similarly to programs, which implement algorithms (independently of the specific data provided as input). The execution of a CoDe model could generate the correct graphical instance, once specific data instances and chart types have been selected. As data change, the same CoDe model can be re-executed so generating an updated report with the same semantics on the new data.

Graphical representations involving one or more graph types (e.g., pie-chart, bar-chart or histogram) can be obtained: CoDe models enable users to generate CoDe Graphs, which integrate different dashboard charts with conceptual links, as shown in Figure 1.2.

The CoDe Graph Generator tool interprets a CoDe model and automatically generates the corresponding CoDe Graph.

In the following, we provide a detailed description of the thesis content and how it has been organized into two main parts. In the first part, the chapters are organized as follows:

In Chapter 1, we describe the main contexts related to the research activities. In particular, we outlined the principal aspects concerning the Rock art and petroglyphs, and successively data visualization and report modeling and generation.

In Chapter 2, we outline related work concerning visualization of archaeological information and image processing of Petroglyphs.

In Chapter 3, we discuss the importance of visual analytics systems and the interactive visual analysis process in the Rock art domain. In particular, we present a visual analytics system, named DARK, for supporting rock art archaeologists in exploring repositories of rock art scenes each consisting of hundreds of petroglyphs carved by ancient people on rocks. DARK combines visualization techniques with fuzzy-based analysis of rock art scenes to infer information crucial for the correct interpretation of the scenes.

In Chapter 4, we propose and describe a petroglyph classification methodology based on the Image Distorsion Model, and then we introduce the tool PetroSketch for supporting archaeologists in the classification and recognition of petroglyph symbols. PetroSketch is a virtual notebook enabling users to draw a petroglyph symbol on a white page, or by following the contour of a symbol captured with the camera, and to obtain its classification and the list of symbols more similar to it. The latter is performed by a flexible image matching algorithm that measures the similarity between petroglyph by using a distance, derived from the image deformation model, which is computationally efficient and robust to local distortions.

The second part of the thesis is organized as follows:

In Chapter 5, we discuss the related work about the representation of information extracted from data warehouses and then the evaluation techniques adopted for the comprehension of graphical reports. To this aim, we analyzed state of art approaches concerning the evaluation of graph representation in terms of comprehension. We extracted and then classified them in order to evaluate the comprehensibility of CoDe Graphs with respect to dashboard reports.

In Chapter 6, we outline the methodology based on the Complexity Design (CoDe) modeling to conceptually organize visualizations of data warehouse reports. In particular, the CoDe methodology provides a technique to model graphical reports on data extracted

by a data-warehouse. Indeed, a CoDe model enables users to describe the semantics of a graphical representation by abstracting from the specific instances of data, similarly to how a program abstracts from a specific assignment of values to the input data. Moreover, we illustrate the syntax and semantics of the CoDe visual language. In particular, in Section 6.1, we describe the main concepts of CoDe Graphs and present the CoDe Graph Generator tool for generating them.

In Chapter 7, we discuss the evaluation of CoDe Graphs comprehension through an empirical evaluation. In particular, we detail the design and planning of the controlled experiment in Section 7.1. To this aim, we assessed user comprehension by evaluating the Effectiveness (i.e., the correctness and completeness in the comprehension of a phenomenon), the Effort (i.e., the time required to comprehend the phenomenon), and the Efficiency (i.e., the ability to effectively comprehend the phenomenon without wasting time). Moreover, in Section 7.2, we summarize the experiment results and discuss their practical implications, together with possible threats that may affect the validity of our study.

Final remarks and future research directions about information visualization are discussed in Chapter 8.

Chapter 2

Visualization of Archaeological Information

Cultural heritage studies are basically analytical in nature and involve several disciplines, each providing a different contribution. Although the systematic study and analysis is a desirable requirement, in practice most of them are subjective and based on personal observations and impressions. In the recent years some works have been carried out to automate the analysis and reasoning activities. As an example, the site geographical representation can assist archaeologists to interpret data and formulate new theories [21]. GIS tools have been equipped with increasing capabilities and employed for decision support applications [101], analytical and modeling applications [28, 62], and even, used in archaeological research for analysis and reasoning activities [68]. The GIS spatial analytical application presented in [106] allows domain experts to investigate the potential extent of a habitat/environment through the analysis of a series of maps, graphs, and tabular data.

The geovisual analytics environment presented in [50] exploits the space time cube to investigate the relationships between sites and artifacts discovered at various sites in order to understand the interaction between cultures. The space time cube is a GIS based implementation of Hägerstrand's original Spacetime Aquarium, which allows to represent the three main components of spatio-temporal data, namely when, where, and what. The space time cube visualizes the space time archaeological data and provides interactive filtering and sorting functions that can be applied to clarify patterns and relationships hidden in the data. They also present the preliminary results on the development of functions for archaeological investigation within a geovisual analytics environment.

The Cyber-Archaeology project is focused on developing technologies and tools for the documentation of cultural heritage [52]. A central component of Cyber-Archaeology is

a visual analytics system that supports the collaborative analysis of multispectral data spanning the broad scales of time (temporal) and space (spatial). This system can also be used to guide the knowledge discovery process and unlock the underlying meaning of an artifact, while determining if the artifact needs to be monitored or requires any type of intervention.

Inkpen *et al.* proposed a protocol based on GIS to monitor stone degradation [54]. The proposed system integrates images from different time periods and different sites to yield useful information.

In [25] visual analytics has been used to facilitate the interpretation of multi-temporal thermographic imagery for the purpose of restoration of cultural heritage. The proposed visual environment allows to explore thermographic data from the unifying spatiotemporal perspective aiming to detect spatial and spatiotemporal patterns that could provide information about the structure and the level of decay of the material.

Vis4Heritage is a visual analytics framework for discovering wall painting degradation patterns [109]. The framework provides users with a set of analytic and hypothesis verification tools to support the multi-resolution degradation analysis.

In the context of the IndianaMAS project, Deufemia *et al.* [33] presented Indiana Finder, a visual analytic system supporting the interpretation of new archaeological findings, the detection of interpretation anomalies, and the discovery of new insights. The approach introduced a data view based on ring charts and an interpretation summary view based on 3D maps.

To the best of our knowledge, no specific approach has been developed for supporting archaeologists to discover new relationships between petroglyphs. The most related idea is proposed in [100] where authors introduced ArchMatrix, a visual interactive system that supports archaeologists in archiving, managing and studying the findings collected during archaeological excavations. The system implements the Harris Matrix method, used to describe the position of stratigraphic units, and provides advanced information retrieval strategies through the use of a graph database.

The approach proposed in this thesis differs from the previous analytics systems both in the managed information, i.e., the relationships between interpretations and spatial relationships between the petroglyphs in the scenes, and how to represent them. Indeed,

the proposed data views are able to summarize this information allowing archaeologists to rapidly infer new ones.

2.1 Rock Art

Petroglyphs are images created by removing part of a rock surface by incising, picking, carving, or abrading, as a form of rock art. This art can be found in many cultures around the world and at many times. Many theories have been advanced and advocated to explain their purpose, depending on their location, age, and the type of image. Some petroglyphs are thought to be astronomical markers, maps, or other forms of symbolic communication, including a form of "prewriting". The form of petroglyphs is described by a variety of terms in the archaeological literature. One of the most common forms of rock art around the world is the anthropomorphic depiction. They are pictures that resemble humans, but sometimes can represent something else, such as the personification of a spirit or other nonliving thing. Other common images are animals, weapons, and tools.

Mont Bego, in the extreme south-east of France, is the most important alpine rock art site. Archaeologists consider this place as an incredibly valuable source of knowledge, due to the up to 40,000 figurative petroglyphs and 60,000 non-figurative petroglyphs [27]. The figurative petroglyphs represent corniculates, harnesses, daggers, halberds, axes, reticulates, rectangular or oval shaded zones, and anthropomorphous figures. Between 1898 and 1910, Clarence Bicknell realized up to 13,000 drawings and reliefs, part of which were published in [12]. Bicknell identified seven types of figures taking a natural history approach: horned figures (mainly oxen), ploughs, weapons and tools, men, huts and properties, skins and geometrical forms [18]. From 1967 Henry de Lumley is in charge of performing research on the site.

2.2 Image Processing of Petroglyphs

The symbol recognition problem is one of the most studied and analyzed research topic in the field of the image processing [67, 97]. But surprisingly, the study of the rock art was only minimally touched by these investigations. Probably, this is due to some unique properties of the petroglyphs (e.g., different petroglyphs may share more or less the same patterns while being different), which make them unsuitable for recognition tasks.

The work presented in [90] aimed to catalogue petroglyphs in terms of lengths of parts of animal bodies, and relations among petroglyphs of several regions. In [95] Takaki *et al.* proposed new methods to characterize shapes of the petroglyphs and the properties of the group they belong to. In particular, they first extract the skeleton of the petroglyph by applying different image processing algorithms, then the structures are expressed through elementary symbols in order to allow a quantitative comparison. The properties of petroglyph groups are expressed by statistics of simple quantities, such as the numbers of animals and men.

Recently, Zhu *et al.* applied a distance measure based on the Generalized Hough Transform to find meaningful motifs within large collections of rock art images [111]. They also proposed a tool called PetroAnnotator, which allows human volunteers to "help" computer algorithms to segment and annotate petroglyphs [110]. Finally, in [87] Seidl and Breiteneder proposed an pixel-wise classification for rock art image segmentation and presented some preliminary results.

Chapter 3

A Visual Analytics System for Rock Art Knowledge Discovery

Rock art is a term coined in archeology for indicating any human made markings carved on natural stone [19,65]. The most part of the symbols concerning rock art are represented by *petroglyphs*, which were created by removing part of a rock surface by incising, picking, carving, and abrading. Although it is not possible to give certain interpretations to these petroglyphs, archaeologists have proposed many theories to explain their meaning, e.g., astronomical, cultural, or religious [105].

In order to digitally preserve, study, and interpret these artifacts the archaeologists have created repositories containing heterogeneous informations, like pictures, 3D images, textual descriptions, GPS coordinates, black and white reliefs, and so on. The exponential growth of these repositories and the high dimensionality of the stored data have made their analysis a major task and challenge for rock art archaeologists. Even though in the recent years several image recognition approaches have been proposed for automating the segmentation and classification of petroglyphs [32, 87, 88, 111], few work has been done to automate the analysis of these repositories in order to support archaeologists in their investigative tasks.

In this chapter, we introduce novel abstraction techniques for ontology-based, interactive visual analysis of large repositories of petroglyphs. These abstraction techniques are based on structural and semantic information and allow users to easily discover new information about petroglyph shapes and relationships. The structural information includes the shape of the carved symbols and the spatial relationships between them. The latter represent a valuable information for the correct interpretation of petroglyphs since

many correlations exist between the spatial relationships of the carved symbols and their interpretations. The semantic information is extracted from the interpretations given by archaeologists and organized according to an ontology. We also present a new data view that uses fuzzy-spatial relationships to visually summarize the scene topological structure. This data view can be used to study petroglyph correlations, infer new information useful to correctly interpret the scenes, and validate new hypothesis.

A prototype system, named DARK (Discovery of Ancient Rock art Knowledge) [29], has been implemented as a component of IndianaMAS, a framework for the digital protection and conservation of rock art sites [72, 73]. DARK supports archaeologists in their investigation activities by performing analytics on the dataset and providing appropriate visualization and interaction techniques. The prototype has been evaluated using data from the database containing information about the rock carving reliefs of Mount Bego.

3.1 Digital Preservation of Rock Art

Petroglyphs are among the oldest form of art known to humans. A petroglyph consists of a single carved symbol as in Figure 1.1(a), or can be the composition of two or more symbols as in Figure 3.1(a) up to several hundreds. Usually the archaeologists working on a rock art site collect petroglyphs, classify them based on their shape, and define dictionaries. Moreover, since petroglyphs were used to convey messages and ideas, archaeologists try to give interpretations to the compositions. The latter are called *scenes* in the rest of the thesis.

The scenes of petroglyphs can be composed of groups of animals, weapons, and so on, and they can be interpreted as religious beliefs, aesthetic concepts, warfare, and modes of life. Although in some cases the symbols composing a scene seem to be arranged without an apparent order, in many others, repetitive logical relationships can be recognized [11, 27]. Such recurrent combinations are called *pattern*. As an example, Figure 3.1(b) depicts a petroglyph interpreted as the mother goddess located above a bull. By analyzing the Mount Bego's petroglyphs, archaeologists have observed that this motif frequently occur in scenes and have interpreted this recurrent pattern as the *birth* [27].

To promote the awareness and the preservation of rock art and to support archaeologists in their investigation activities we are developing a platform, named *IndianaMAS* [72], that

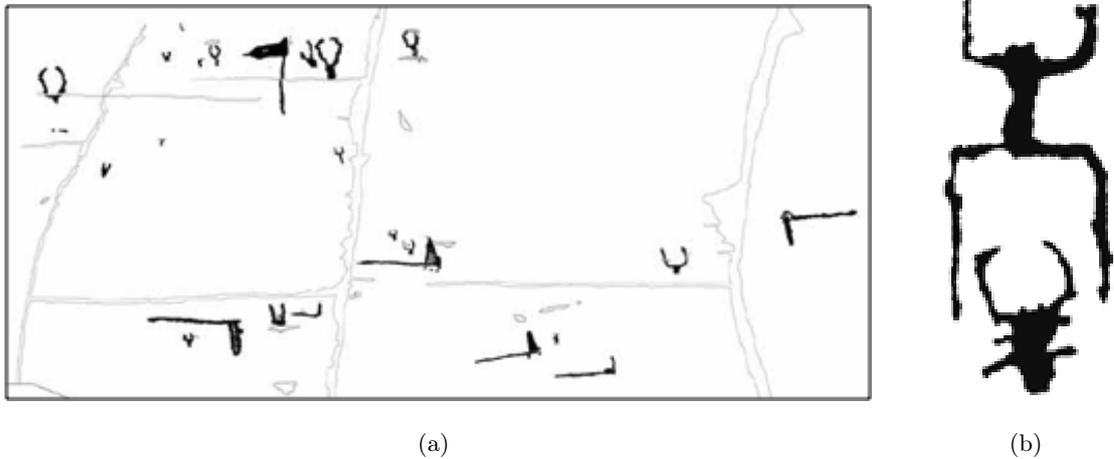


Figure 3.1: A scene composed of several corniculates and halberds (a) [11], the pattern mother goddess giving birth to the bull (b) [27].

integrates and complements the techniques usually adopted to preserve cultural heritage sites. The platform exploits ontologies to provide a shared and human-readable representation of the domain [43], intelligent software agents to analyze the digital objects analysis and perform reasoning and comparison activities over them [56], and standard tools and technologies for Digital Libraries to manage and share digital objects [13]. IndianaMAS enables the digital preservation of all kinds of available data about rock carvings, such as images, geographical objects, textual descriptions of the scenes. It provides the means to organize and structure such data in a standard way and supplies domain experts with facilities for issuing complex queries on the data repositories and making assumptions about the way of life of ancient people.

The database of petroglyphs consists of textual interpretation descriptions, ontological information, geographical coordinates, pictures and drawings of the reliefs. The ontological information is a list of concepts that summarizes the interpretations given by archaeologists. Figure 3.2 shows a part of the ontology we defined for the rock art archeology domain.

3.2 The DARK System

As said above, the recurrent combinations of petroglyph symbols are very important for the correct interpretation of scenes. However, their identification is a very challenging

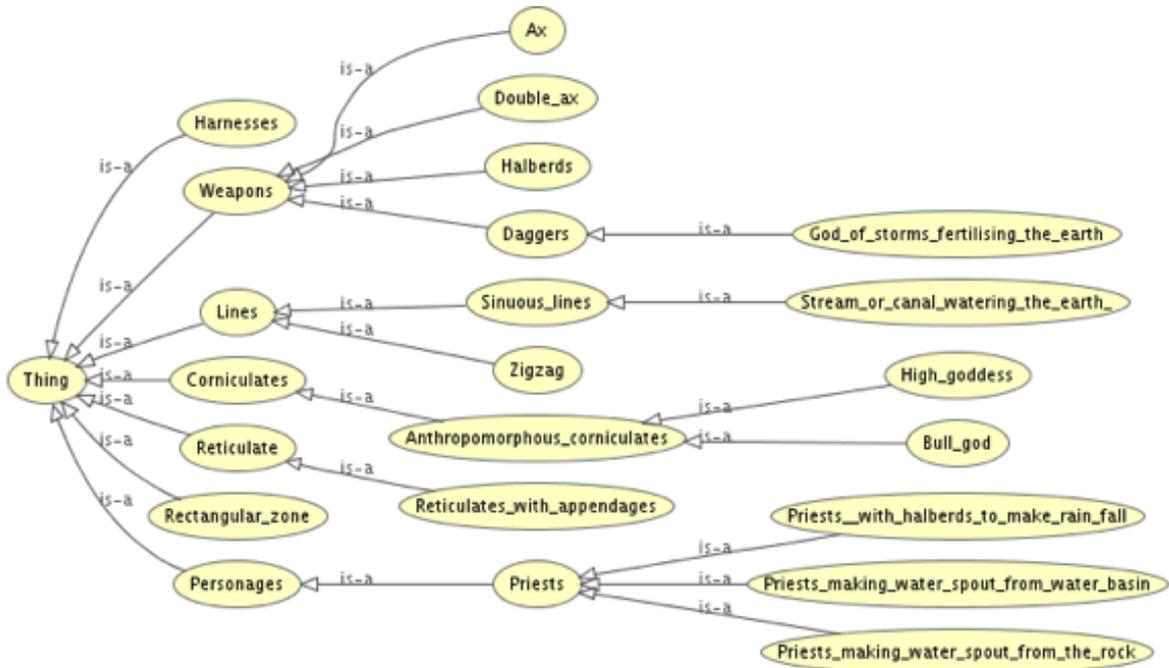


Figure 3.2: Part of the ontology defined for the rock art archeology domain.

task mainly due to the size and complexity of data stored in the repositories. So far, archaeologists have identified these patterns in an empirical way. In this section we present the DARK system whose aim is to support archaeologists in the detection and analysis of these patterns in a systematic way.

DARK is a visual analytics tool designed to take advantage of semantic information (i.e., ontological concepts) associated to the interpreted petroglyphs. It allows users to more easily isolate petroglyph symbols and their relationships for inferring and validating new patterns. The tool performs both a structural abstraction by clustering the petroglyphs with similar shapes and a semantic abstraction by exploiting the ontological information associated to the scenes. Moreover, to visually represent and summarize the spatial arrangement of the petroglyphs in the scenes, DARK includes a data view exploiting fuzzy theory to manage the uncertainty of spatial relationships. The tool also includes common visual interaction techniques, as zooming and filtering, in order to assist users in the investigation process.

3.2.1 Structural and Semantic Abstraction through Bubble View

The aim of the structural and semantic abstractions is to extract relevant information from the scene repository in order to simplify the detection of recurrent patterns. In the following we formalize the abstraction process and present the data view implemented in DARK that visually summarize the abstracted information.

The Abstraction Process

Let $S = \{s_1, \dots, s_n\}$ be the set of symbol classes defined by archaeologists to classify petroglyphs and $O = \{o_1, \dots, o_m\}$ be the set of concepts contained in the ontology. The set of scenes P stored in the repository can be formally defined as $P = \{p_1, \dots, p_t\}$ where $p_i = \{r_{i,1}, \dots, r_{i,q}\}$ contains the petroglyph reliefs of the i -th scene. Each relief $r_{i,j} \in p_i$ is associated to a symbol in S by using the function *Shape*, i.e., $Shape(r_{i,j}) = s_k \in S$, which applies an approximate image matching algorithm. Moreover, the function *Sem* provides the set of ontology concepts associated to a scene, i.e., $Sem(p_i) \subseteq O$.

Given an ontology concept o_i , DARK constructs a graph where the nodes correspond to the symbol classes obtained by applying the *Shape* function to the reliefs of p_j such that $o_i \in Sem(p_j)$ of petroglyph symbol classes from P by including all nodes whose classes are related to o_i . The goal is to visualize the most frequent symbol classes co-occurring with a concept. In order to achieve this goal, abstraction hides the symbol classes non relevant for the selected concept and clusters the petroglyph reliefs having similar shape. More formally, an abstraction with respect to an ontology concept o_i is defined as a graph $G_{o_i} = (V, E)$ with:

- $V = \{s_x \in S \mid \exists r_{j,k} \in p_j \text{ with } Shape(r_{j,k}) = s_x \text{ and } o_i \in Sem(p_j)\}$ and
- $E = \{(s_x, s_y) \mid s_x, s_y \in V, \exists r_{j,k}, r_{l,m} \in p_j \text{ with } Shape(r_{j,k}) = s_x \text{ and } Shape(r_{l,m}) = s_y, \text{ and there exists a spatial relationship between } r_{j,k} \text{ and } r_{l,m}\}$.

Thus, the graph constructed for an ontology concept o_i represents the petroglyph symbol classes co-occurring with o_i and the spatial relationships between them. In DARK, the *Shape* function has been implemented by using the IDM algorithm used in [31], while the approach for computing the spatial relationships between the petroglyphs in the scenes is described in Subsection 3.2.2.

The Bubble View

DARK visualizes the graphs obtained from the abstraction process by using the Bubble view.

In particular, the view shows a colored bubble B_{o_i} for each graph G_{o_i} . The color of each B_{o_i} depends on the position of the concept o_i in the ontology hierarchy, namely we assign a different hue to each subtree of the nodes at level two, then we decrease the saturation on the basis of the level. As an example, Figure 3.3 depicts the Bubble view constructed for the ontology concepts at level three¹. *Fertilize* and *storm*, being children of the *Agriculture* level-two node are pink-colored, while *birth* and *sacrifice*, being children of *Earth* and *Religion*, are green and cyan-colored, respectively. It is worth to notice that the bubbles corresponding to semantically similar concepts are depicted with similar colors so as the *fertilize* and *storm* bubbles.

The size of a bubble indicates the frequency of the concept o_i in the scenes (small = low frequency, big = high frequency). In the example, we notice that the most frequent concept is *storm*.

The bubble B_{o_i} also visualizes both the most frequent symbols co-occurring with o_i and their relationship degree. In particular, the size of symbols indicates how much they frequently appear in the scenes, while the size of the links provides a suggestion about their relationship degree.

In this way, the Bubble view allows archaeologists to discover candidate patterns by highlighting the prevalent key symbols and common relationships. As an example, the graph inside the *sacrifice* bubble shows a potential pattern. The *bull* and *halberd* symbols are very frequent in the scenes associated to the *sacrifice* concept, thus, their size appears bigger with respect to the other symbols. Moreover, the bold line indicates a high spatial relationship degree between them in the scene set.

3.2.2 Scene Spatial Analysis using Ring View

The information visualized in a Bubble view provides an overview of the correlations between the symbol relationships and the ontological concepts extracted from their inter-

¹Archaeologists can select the ontology concepts either by interacting with a zooming bar, where it is possible to choose the granularity level of the concepts, or by navigating the ontology through the taxonomy relation (IS-A).

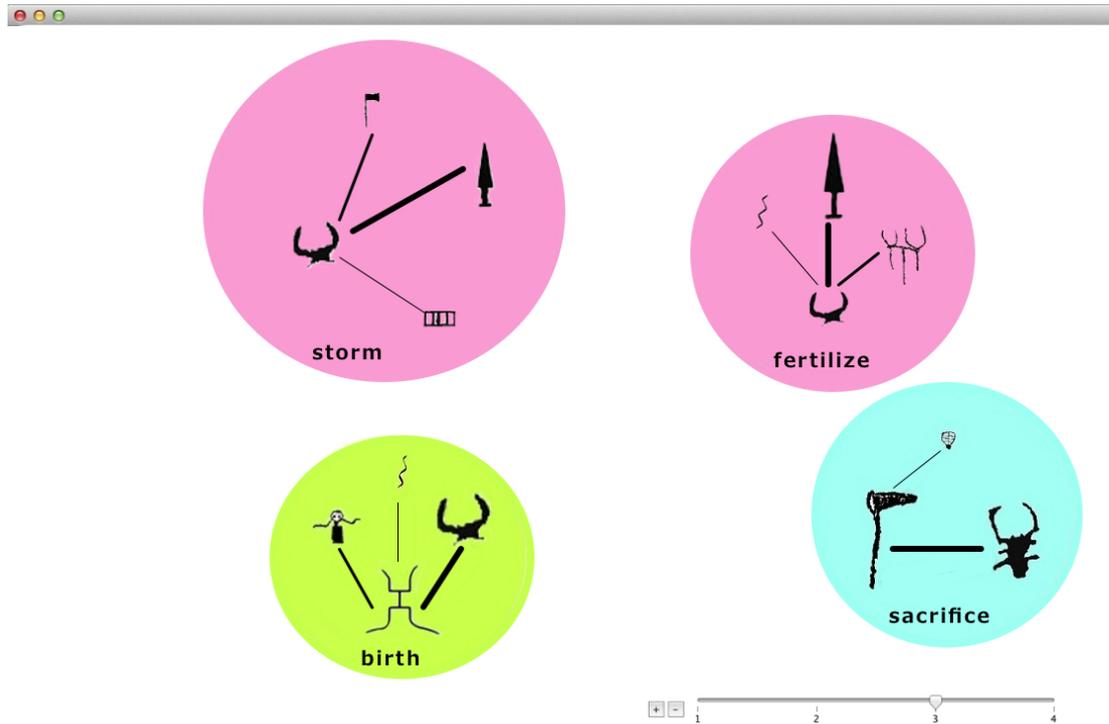


Figure 3.3: A Bubble view constructed for the concepts: *fertilize*, *storm*, *birth*, and *sacrifice*.

pretations. When the archaeologist discovers a candidate pattern, s/he needs fine-grained information on the symbols relationships involved in the pattern in order to validate it. In particular, s/he has to analyze the spatial relationships existing between the petroglyph symbols and their correlation with the selected concept.

Fuzzy Spatial Analysis of Scenes

Since the petroglyph symbols were carved in an inaccurate way, the spatial relationships between petroglyphs cannot be computed using crisp relations. We use the fuzzy logic to model the uncertainty, as this logic allows to express the concept of belonging where can not be easily defined [111]. For example, in spatial reasoning, two entities cannot fully belong to only one relation space so the fuzzy logic helps us to model the concept of uncertainty. In particular, in order to obtain the topological structure of a scene we use the region connection calculus (RCC) [79], while the directional properties between objects have been computed using the Cardinal Direction Calculus (CDC).

18 Chapter 3. A Visual Analytics System for Rock Art Knowledge Discovery

The RCC is a topological approach to the qualitative spatial representation in which the regions are regular subsets of a topological space. This method gives information about the topological structure of an image by associating it one of the topological relationships in a two-dimensional space. The RCC-8 calculus considers eight basic topological relations, which are: *externally connected* (EC), *disconnected*(DC), *tangential proper part* (TPP), *tangential proper part inverse* (TPPi), *non-tangential proper part* (NTPP), *non-tangential proper part inverse* (NTPPi), *partially overlapping* (PO), *equal* (EQ). However, it is possible to reduce the number of spatial relationships required by decreasing granularity. As an example, it could be possible to use either RCC-5 or RCC-3. This granularity reduction process has been extensively discussed in [52]. In general, the scene interpretative process does not require a fine relationship granularity, thus we use a low level RCC so as RCC-3 which includes the relationship types, *disjoint*, *overlap*, and *meet*.

As said above, cardinal directional relationships can be computed through CDC. It specifies the direction which one object is cardinally related to one another. It can be computed to obtain different levels of granularity, as the CDC-8 which correspond to determine the *north*, *south*, *west*, *east*, *south-east*, *south-west*, *north-east*, *north-west* directions, or the CDC-4 which corresponds to the *north*, *south*, *west*, *east*, or even the CDC-1 where the information represents only a relationship in any cardinal direction. In our case it would be useful keep the fine-grained information, thus we use the CDC-8.

In order to use this information we start applying the method proposed in [84], that combines fuzzy Allen's relationships, specifically designed to manage topological relationships, to directional relationships in the two-dimensional space. The Allen's relationships, explained in [3], are based on the algebra of the time intervals and are labeled as follow: $A = \{<, m, o, s, f, d, eq, di, fi, si, oi, mi, >\}$. These relationships assume the meaning of: *before*, *meet*, *overlap*, *start*, *finish*, *during*, *equal*, *during by*, *finish by*, *start by*, *overlap by*, *meet by*, and *after*. Successively, the method applies a fuzzification process to obtain the fuzzy value of the spatial information. It creates a matrix 8×8 , where the rows hold the CDC-8 topological relationships and the columns have the qualitative directional aspects of the 2D scene information. The relationships are expressed using numeric values representing the percentage area of two objects under a specific topological relationship in the qualitative direction. More precisely, the cell value for a topology relationship of a given direction is obtained using the fuzzy contributes of some Allen's relationships calculated

on specific angles depending on the direction. For instance, in case of *north* direction, it is possible to calculate the fuzzy value of the *disjoint* topological relationship by means of the *after* Allen’s relationship over the angles between $1/4\pi$ and $3/4\pi$ according to the following formula:

$$Disjoint_North = \sum_{\Theta=\pi/4}^{3/4\pi} after(\Theta) * \cos^2(2 * \Theta) \quad (3.1)$$

For the sake of clarity, let us shown two examples describing how this model works. In the first example, shown in Figure 3.4(a), we have a scene where the *god* petroglyph is located above (*north*) the *goddess* petroglyph. The petroglyphs are disjoint to each other as well. The matrix associated with this scene contains higher values in correspondence of the cells indicated by the rows 1-*disjoint* and the columns 2-*north-east*, 3-*north*, 4-*north-west* (see Figure 3.5(a)).

In the latter example, shown in Figure 3.4(b), *bull* and *goddess* petroglyphs overlap to each other. At the same time, *goddess* is located at *north* side of *bull*. As shown in Figure 3.5(b), the matrix representation of this relationship involves several rows, 1-*disjoint*, 2-*meet*, 3-*partial overlap*, 4-*tangent proper part*, 5-*nontangent proper part*, and columns, 2-*north-east*, 3-*north*, 4-*north-west*. In this case, it is possible to notice that the higher value is obtained in correspondence of the cell [3-*partially overlap*, 3-*north*].

The Ring View

For assessing a topological-directional fuzzy relationship, DARK analyzes its frequency (how many times the relationship is satisfied) and its strength (how high is the fuzzy value) and visualizes this information in the Ring view. In order to visually represent this information the Ring view shows three concentric circles, which corresponds to the topological relationships of RCC-3 obtained following the reduction process from RCC-8 described in [52]. The inner circle refers to the *partially overlap* relationship, the middle one to the *meet* relationship, while the *disjoint* relationship is associated with the outer sector. Each circle is divided into eight sectors, which correspond to the directional relationships.

The fuzzy values obtained for a topological-directional relationship between two petroglyphs are mapped into the corresponding cells of the Ring view. In particular, we have proposed two different visualizations of the fuzzy relationships. Let $S = \{S_1, S_2, \dots, S_n\}$

be a set of scenes, A and B be two petroglyphs related by means of the $R(A,B)$ relationships, and $V(R)$ be the corresponding cell in the Ring view. The first visualization colors $V(R)$ using the average of the fuzzy values obtained evaluating the fuzzy relationship R for each instance of $R(A,B)$ in S . The red color is used for the *disjoint* sectors in the outer circle, purple for *meet*, and green for *partially overlap* as shown in Figure 3.6(a). Notice that more intense colors imply higher values of the relationship while the black color indicates the total absence. In the second visualization, the fuzzy values of $R(A,B)$ in S are mapped using the color gradient, which highlights at the same time the frequency and the strength of the relationship. Let us suppose we divide a sector into n parts $P = \{P_1, P_2, \dots, P_n\}$ each associated with a strength range $F = \{F_1, F_2, \dots, F_n\}$, the size of the sector part P_i is proportional to the frequency of the $R(A,B)$ fuzzy values falling into the range F_i , while the color intensity of P_i is given by calculating the average of such fuzzy values. The gray color indicates the absence of relationship, while the purple indicates the presence. For example, in the *Disjoint-North-East* sector of 3.6(b), the number of times the relationship $R(A,B)$ has high fuzzy values is medium as well as the number of intermediate values, low values are a few. On the other hand, if we consider the *Disjoint-North* sector, we have medium quantity of high fuzzy values and high intermediate values.

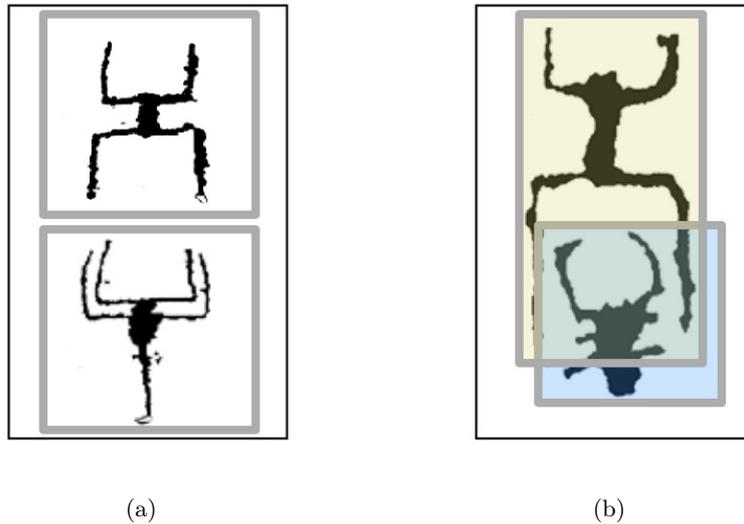


Figure 3.4: A petroglyph scene containing the *god* and the *goddess* (a), and an example of interpretation of the *overlap-north* relationship between the *mother goddess* and the *bull* (b) [32].

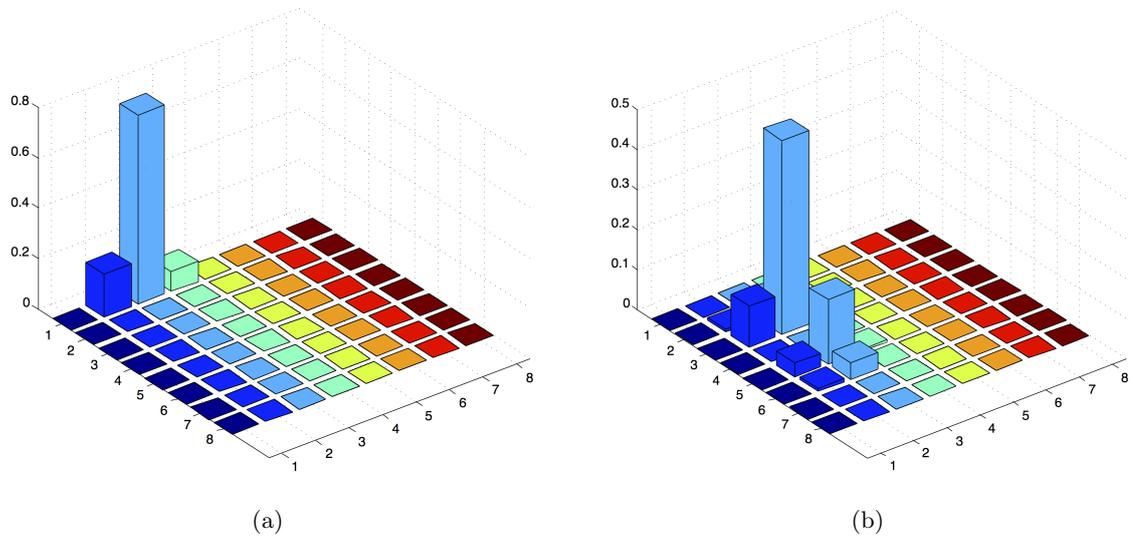


Figure 3.5: The matrices representing the fuzzy relationships between the symbols in the scenes of Figure 3.4. The correspondence between numbers and relationships is the following: on rows, 1-disjoint, 2-meet, 3-partially overlap, 4-tangent proper part, 5-nontangent proper part, 6-tangent proper part inverse, 7-nontangent proper part inverse, 8-equal, and on columns, 1-east, 2-north-east, 3-north, 4-north-west, 5-west, 6-south-west, 7-south, and 8-south-east.

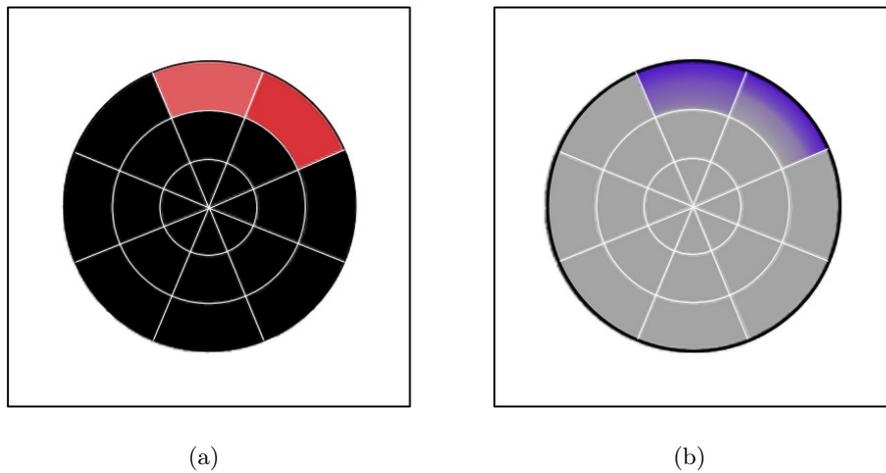


Figure 3.6: Two different visualizations of the Ring view: average fuzzy values (a) and gradients (b).

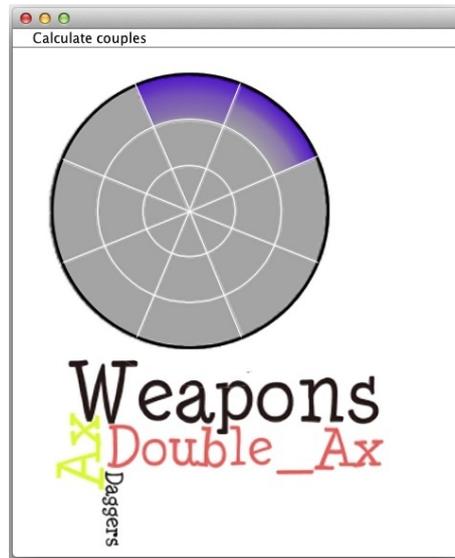


Figure 3.7: A Ring view with the Metadata view generated by selecting the North-Disjoint sector.

Archaeologists can also interact with the Ring sectors for analyzing the ontology concepts associated to the set of scenes. To this aim, we defined the Metadata view. It shows the most frequent ontology concepts of the scenes by using a tag cloud visualization. As an example, Figure 3.7 depicts the Metadata view generated by selecting the *North-Disjoint* sector of the Ring. The most frequent terms in the scenes containing such an arrangement are *Weapons*, *Double_Ax*, and *Ax*. By using this view, archaeologists verify the correctness of the discovered pattern.

Chapter 4

Classification and Recognition of Petroglyphs

Petroglyphs are engravings obtained by hitting or scraping stone surfaces with sharp tools made of stone or metal. These symbols were used by prehistoric people to communicate with their divinity or with other men. Basically, we find these carvings as a set of symbols called scenes, which depicts daily life situations. Therefore, the study of the petroglyphs is very important to gain new knowledge and awareness about the periods in which the petroglyphs were created.

Petroglyphs are prehistoric stone engravings unrevealing stories of ancient life and describing a conception of the world transmitted till today. Although they may seem as durable as the rock they reside on, petroglyphs are inevitably deteriorated by natural causes as well as vandalism. As an example, some of them have been destroyed by tourists, while others are disappearing due to acid rain and sunshine. Hence, it is very important to preserve the petroglyphs trying to identify and archive them for future generations.

Research challenges have to be addressed for digital preservation of petroglyphs, including the integration of data coming from multiple sources and the correct interpretation of drawings. The IndianaMAS project is aiming to integrate heterogeneous unstructured data related to rock carvings into a single repository, organizing classified data into a Digital Library, interpreting data by finding relationships among them, and enriching them with semantic information [33, 74]. However, the large number of sites where petroglyphs have been discovered, and the different methodologies they were made, makes their study a complex task. Archaeologists have to deal with a hard and tedious job because they have to report each symbol shape on a notebook paper and search any related information by consulting different books. To address these issues, we present PetroSketch [30], a mobile

application for supporting archaeologists in the classification and recognition of petroglyph symbols.

PetroSketch is a virtual notebook enabling users to draw a petroglyph symbol on a white page, or by following the contour of a symbol captured with the camera, and to obtain its classification and the list of symbols more similar to it. The latter is performed by a flexible image matching algorithm that measures the similarity between petroglyphs by using a distance, derived from the Image Deformation Model (IDM), which is computationally efficient and robust to local distortions. The classification system has been applied to an image database containing 17 classes of petroglyph symbols from Mount Bego rock art site achieving a classification rate of 68%.

4.1 Rock Art

Petroglyphs are a form of prehistoric art found in many cultures around the world and at many times. There are many theories to explain their purpose, depending on their location, age, and the type of image. Some petroglyphs are thought to be astronomical markers, maps, or other forms of symbolic communication, including a form of "prewriting". The form of petroglyphs is described by a variety of terms in the archaeological literature. One of the most common forms of rock art around the world is the anthropomorphic depiction. They are pictures that resemble humans, but sometimes can represent something else, such as the personification of a spirit or other nonliving thing. Other common images are animals, weapons, and tools.

We experimented our approach on the reliefs collected and cataloged from Mont Bego, in the extreme south-east of France, which due to the richness of the place in both qualitative and quantitative terms it is ideal for analysis. Archaeologists consider this place as an incredibly valuable source of knowledge, due to the up to 40,000 figurative petroglyphs and 60,000 non-figurative petroglyphs [27]. The figurative petroglyphs represent corniculates, harnesses, daggers, halberds, axes, reticulates, rectangular or oval shaded zones, and anthropomorphous figures. Between 1898 and 1910, Clarence Bicknell realized up to 13,000 drawings and reliefs, part of which were published in [12]. Bicknell identified seven types of figures taking a natural history approach: horned figures (mainly oxen), ploughs, weapons and tools, men, huts and properties, skins and geometrical forms [18]. From 1967 Henry

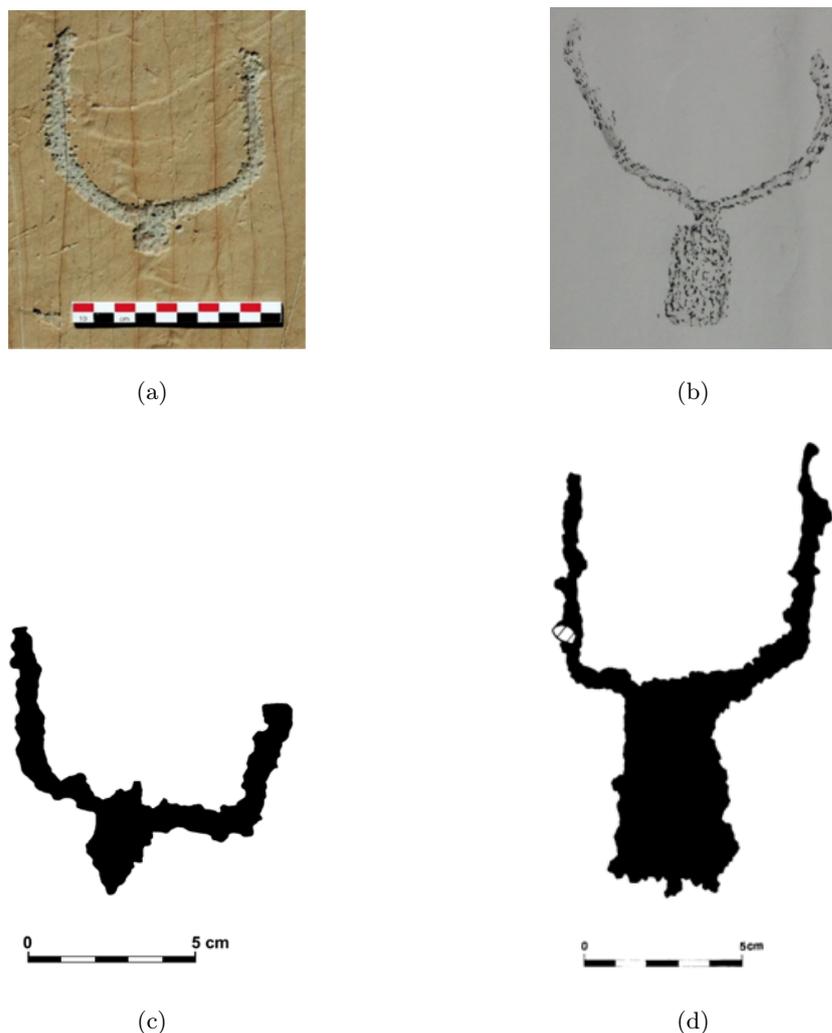


Figure 4.1: A picture of a bovine engraving of the Mont Bego (a), a picture of a bovine relief made by Bicknell on botanic sheet (b), two digitalized reliefs made by de Lumley's team (c) and (d).

de Lumley is in charge of performing research on the site. Figure 4.1 shows a picture of a bovine engraving, a Bicknell's relief, and two digitalized reliefs made by de Lumley's team.

4.2 An Approach for Petroglyph Classification

One of the most promising approach to achieve low error rates in the classification of images with high variability is the application of flexible matching algorithms [8]. Among them,

the deformation models are especially suited to compensate small local changes as they often occur in the presence of image object variability [59]. These models were originally developed for optical character recognition by Keyzers *et al.* [59] but it was already observed that it could be applied in other areas such as recognition of medical radiographies [60] and video analysis [35]. The IDM yields a distance measure tolerant with respect to local distortions since in the case two images have different values only for a few pixels, due to noise or artifacts irrelevant to classification, the distance between them is compensated by specifying a region in the matching image for each picture element in which it is allowed to detect a best matching pixel.

These properties motivate its use for petroglyph classification. In the following we describe the steps of our petroglyph classification system.

4.2.1 Shape Normalization

To recognize a petroglyph symbol regardless of its size and position, the input image is normalized to a standard size by translating its center of mass to the origin. The resulting image $f(x, y)$ is the grid image of the symbol. Then to increase tolerance to local shifts and distortions we smooth and downsample the feature images. In particular, first, to ensure that small spatial variations in the symbol correspond to gradual changes in the feature values, we apply a Gaussian lowpass filter

$$G(x, y) = e^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma^2}\right)} \quad (4.1)$$

to obtain the smoothed image $g(x, y)$ according to the following equations

$$g(x, y) = f(x, y) \times G(x, y) \quad (4.2)$$

We then downsample the images by performing symbol removing and resizing (see Figure 4.2). This further reduces sensitivity to small shifts and improves runtime performance.

4.2.2 Feature Set

To achieve better performances, instead of directly comparing image pixels, we use derivatives. In particular, for each pixel we consider the horizontal and vertical gradients as

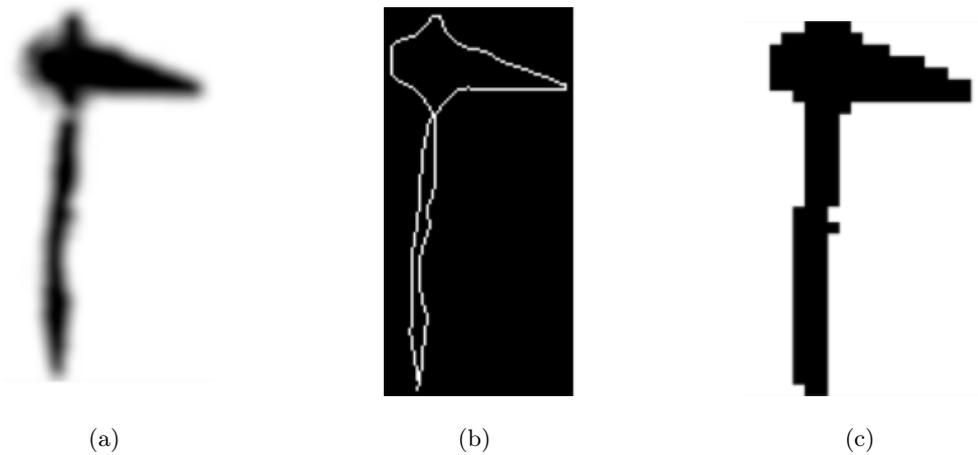


Figure 4.2: An example of normalization of an ax petroglyph: The image smoothed with the Gaussian filter (a), the point removed image (b), and the image resized at 32×32 pixels (c).

features for image matching. Each pixel of a gradient image measures the change in intensity of that same point in the original image, in a given direction. Thus, the horizontal and vertical gradients allow to get the full range of directions.

4.2.3 Classification

For the classification of petroglyphs we use a deformation model that is robust to distortions and local shifts. In particular, the image deformation model (IDM) performs a pixel-by-pixel value comparison of the query and reference images determining, for each pixel in the query image, the best matching pixel within a region around the corresponding position in the reference image.

The IDM has two parameters: warp range (w) and context window size (c). Figure 4.3 illustrates how the IDM works and the contribution of both parameters, where the warp range w constrains the set of possible mappings and the $c \times c$ context window computes the difference between the horizontal and vertical gradient for each mapping. It should be noted that these parameters need to be tuned.

The algorithm requires each pixel in the test image to be mapped to a pixel within the reference image not more than w pixels from the place it would take in a linear matching. Over all these possible mappings, the best matching pixel is determined using the $c \times c$

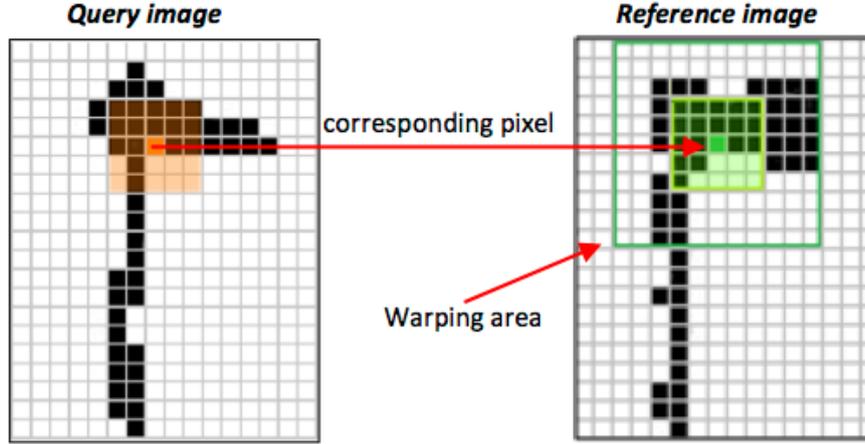


Figure 4.3: Example of areas affected by the comparison of pixels with IDM, where $w = 3$ and $c = 2$. The query pixel context (indicated by the orange area in the query image) is compared with each equal-sized rectangle within the warping area (dark-green rectangle of the reference image). The warping area is calculated by building a $m \times m$, with $m = (w + c) * 2 + 1$, square around the corresponding reference pixel (dark-green pixel).

local gradient context window by minimizing the difference with the test image pixel. In particular, the IDM distance D between two symbols S_1 (the query input) and S_2 (the template) is defined as:

$$D^2 = \sum_{x,y} \min_{d_x, d_y} \| S_1(x + d_x, y + d_y) - S_2(x, y) \|^2 \quad (4.3)$$

where d_x and d_y represent pixel shifts and $S_i(x, y)$ represents the feature values in S_i from the patch centered at x, y .

4.2.4 Performance Optimization

One of the limitations of IDM algorithm is the high computational complexity, which is even further increased when the warp range and local context are enlarged. Thus, since applying IDM to all the reference images is too slow, we introduce two optimization strategies to speed up the IDM algorithm.

The first optimization is to prune the set of candidates before applying IDM. We use the simple Euclidean L^2 distance as the pruning metric, and the first N nearest neighbors

found are given in input to the IDM. In particular, we use the distance:

$$D^2 = \sum_{k=1}^K (v_1(k) - v_2(k))^2 \quad (4.4)$$

where $v_i(k)$ corresponds to the horizontal and vertical gradients of the i -th image.

The second optimization is the early termination strategy proposed in [92], which relies on the consideration that in k NN classifiers only the k nearest neighbors are used in the classification step. Therefore the exact distance of any image with rank greater than k is not used by the classifier. This means that we can abort the computation of the distance between two reliefs as soon as it exceeds the exact distance of the image with rank r . Since the latter can only be known after all images in the collection have been compared to the query, we approximate it with the distance of the k nearest neighbor identified so far.

4.3 PetroSketch

PetroSketch is a mobile application developed with the Titanium¹ framework that allows generic users (which in most cases are tourists, trekkers or just archeology interested people) to support archeologists in collecting and interpreting petroglyph scenes.

Figure 4.4 shows the architecture of PetroSketch. Since the IDM recognition algorithm has been implemented in Matlab environment, we developed a proxy to manage the communication between the server part of the web services and the MCR (MATLAB Compiler Runtime) by using the MATLAB Builder JA.

Figure 4.5 describes the user interface of the application. It appears as a mobile application where the main part of the screen is occupied by the camera view. We did this choice to focus the user's attention in providing better sketches of the petroglyphs. The remaining part of the screen is employed for managing the menus and the corresponding functionalities. It also provides users with drawing, storing, and retrieving capabilities. In particular, its main functionalities are:

- *Camera*: This item allows users to start the picture acquisition process. It works very similarly to other capture software so we do not need to enter in details. From

¹<http://www.appcelerator.com/>

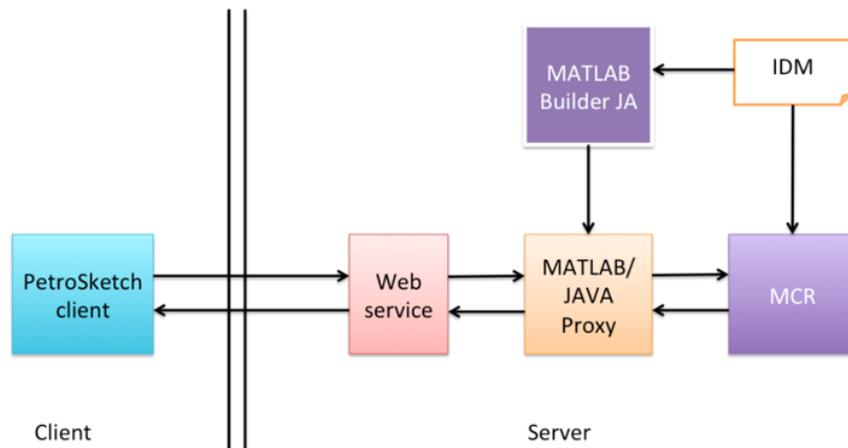


Figure 4.4: The architecture of PetroSketch.

this phase, it is possible to automatically enter in the drawing mode to trace the borders of the petroglyphs;

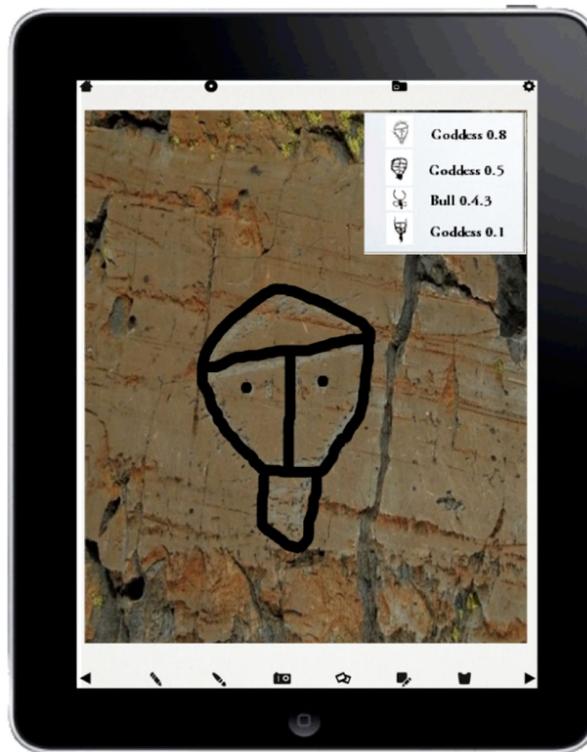


Figure 4.5: The PetroSketch user interface.

- *Pencil Tool*: Here, the user traces the petroglyph borders. S/he can enable this mode for drawing the contours of each petroglyph appearing in the picture or simply choosing the colors or the size of the pencil. Generally, in order to distinguish different single petroglyphs, it is necessary to use different colors for each of them;
- *Album*: The repository of the captured pictures is the Album. Here, photos are stored and previewed. The user can browse the content and choose the one s/he wants to manage;
- *Recognition*: This item allows to begin the recognition phase. By using the previously described approach, the system enables the search and comparison of similar petroglyphs contained within the main remote repository. The image shown in Figure 4.5 presents an example where the borders depicted in the center of the picture is compared to the remote repository and the final results are presented in the top-right corner of the screen highlighting the most similar together with the similarity values.

4.4 The Experiment

To demonstrate the validity of the proposed approach an experiment focused on measuring the effectiveness of the IDM algorithm on a real dataset has been performed. The considered dataset was extracted from the image reliefs presented in [27] and it is a representative sample of the Mount Bego petroglyphs. Basically, it contains a number of petroglyph reliefs falling into 10 main classes (anthropomorphic, ax, bull, bullgod, dagger, goddess, oxcart, personage, reticulate, stream). These classes were successively refined, with the help of several archaeologists participating to the project, into 17 classes based on the shape of the petroglyphs and additional information, such as the estimated date of the engravings.

The 3-fold cross validation test has been successively applied to analyze the performance of the IDM algorithm on the test set. The difficulty to collect data makes particularly suitable this choice. Generally, in k -fold cross-validation, the original sample is randomly partitioned into k subsamples. Of the k subsamples, a single subsample is retained as the validation data for testing the model, and the remaining $k - 1$ subsamples are used as dataset. The cross-validation process is then repeated k times (the folds), with each of the k subsamples used exactly once as the validation data. The k results from the folds

then are averaged to produce a single estimation. In this way all petroglyph reliefs are considered for both dataset and validation, and each relief is used for validation exactly once.

4.4.1 Settings

The best performance of the algorithm was achieved using the following configuration settings:

1. **Image size.** this value was set to 16 pixels. It indicates the size of the reduced image fed to the IDM. We tried also 24 and 32 pixels but no substantial differences have been found. Thus, we preferred to use the minor size for improving time performance.
2. **Warp range.** Together with the local context is one of the parameters of the IDM algorithm. The algorithm obtained the best performance using the value of 3 pix-els.
3. **Local context.** This value was set to 2 pixels.

4.4.2 Results

To evaluate the achieved results we used the k-nearest-neighbor (k NN) classification method [24], which consists in the examination of the first k results produced by the algorithm.

The overall results of the experiment are listed in Table 4.1. Data are presented in a 20x9 table where the first row represents the classification method, namely 1-NN for the best matching class, 3-NN and 5-NN for the nearest neighbors methods with 3 and 5 considered hits, respectively. The first column lists the 17 classes considered in the experiment. Basically, each data cell indicates the percentage of times a petroglyph relief is correctly classified by considering the first i hits of the IDM algorithm result and by applying the classification method (1-NN, 3-NN or 5-NN). As an example, let us consider the Ax_B row. As for the 1-NN method, during the experiment the Ax_B images were correctly associated to the class, namely another Ax_B image appeared as first hit of the IDM algorithm, only in the 33%. In the 67%, it appears either as first or second hit, and finally, it appears in 100% either as first, second, or third hit. If we consider the 3-NN method, the images of the same classes falling into the first three hits are aggregated using the inverse distance weighting. In this way, the class having the highest distance in the

Table 4.1: Classification rates of the proposed IDM algorithm.

Symbol	1-NN				3-NN		5-NN	
	1 st	2 nd	3 rd	4 th	1 st	2 nd	1 st	2 nd
<i>Antropomorphe</i>	67	100	100	100	67	100	33	67
<i>Ax_A</i>	33	67	67	100	33	67	0	100
<i>Ax_B</i>	33	67	100	100	33	100	33	100
<i>Bull_A</i>	100	100	100	100	67	100	100	100
<i>Bull_B</i>	100	100	100	100	100	100	100	100
<i>Bullgod</i>	67	100	100	100	67	100	67	100
<i>Dagger_A</i>	100	100	100	100	100	100	100	100
<i>Dagger_B</i>	33	67	67	100	67	67	33	67
<i>Goddess_A</i>	67	67	67	67	33	67	33	67
<i>Goddess_B</i>	67	67	67	67	67	67	67	67
<i>Goddess_C</i>	67	67	100	100	67	67	67	67
<i>Goddess_D</i>	67	67	67	67	33	67	0	67
<i>Oxcart</i>	100	100	100	100	67	100	67	100
<i>Personage</i>	33	67	67	67	33	67	0	67
<i>Reticulate</i>	67	100	100	100	67	100	67	100
<i>Stream_A</i>	100	100	100	100	100	100	100	100
<i>Stream_B</i>	67	67	100	100	33	100	67	67
total	68	82	88	92	61	86	54	84

new ordered list is suggested to be the class that the query drawing image belongs to. In case of *Ax_B*, its images are correctly classified in the 33% considering only the first hit of the result list and in the 100% considering the two best hits. The same analysis has been performed for the 5-NN classification method.

The last row in Table 4.1 indicates the average values of the columns. Basically, they erase the differences among classes in terms of correct response percentages and report the average behavior of the IDM associated with the different classification methods.

4.4.3 Discussion

By analyzing the results shown in Table 4.1 it is possible to notice that the IDM associated to the 1-NN classification method has a precision of 68%, slightly worst in case of 3-NN

(61%), and even more worst for 5-NN (54%). Probably, this is due because even though the most similar relief falls into the same class of the query, allowing the 1-NN classifier to correctly recognize, the other hits are not so different from the query but belong to different classes (two symbols for class are in the dataset). In this way, aggregation of the 3-NN and 5-NN makes the choice of the class more difficult and address it towards wrong classes rather than the class the query belongs to.

Another consideration concerns with the ability of the approach to suggest a number of possible solutions among which to choose the correct class. It is possible to notice that, even if the best classification approach is 1-NN, this is not always true when the algorithm try to suggest a range of possible solutions. Indeed, when considering the most scored two, 3-NN and 5-NN work slightly better than 1-NN (86% and 84% versus 82%). Unfortunately, due to the low number of symbols for class, we cannot extend this consideration for the three or four most scored hits. Anyway, for these cases, the 1-NN allows to correctly classify in the 88% and 92% of cases.

These results highlight the quality of the IDM algorithm in the handling of image deformation but also the natural complexity of the problem faced in this thesis. Indeed, in most cases, different classes contain similar reliefs and only reasoned-contextual information may help to correctly classify them.

Chapter 5

Visualizing Information in Data Warehouses Reports

Several techniques and tools have been proposed in the literature to relate and display information extracted from a data warehouse (DW), but to the best of our knowledge, no one allows to graphically represent relationships between data in different reports and consider visualizations that involve more than one type of graph. In particular, these approaches usually just provide standard graphical tools and do not allow to compose, aggregate and change the different visualizations. Moreover, these tools allow to represent data with a dashboard visualization that shows graphs separated without displaying their connections. On the contrary, our tool represents both graphs and their connections in a single visualization.

Ma *et al.* [69,70] describe the design and the implementation of a meteorological DW by using Microsoft SQL Server. In particular, the proposed system creates meteorological data warehousing processes based on SQL Server Analysis Services and uses SQL Server Reporting Services to design, execute, and manage multidimensional data reports. Moreover, the generated data reports are represented as folding tabular form related to two-dimensional maps available through the browser, but users cannot choice the type of visualization.

Hsu *et al.* [48] apply a clustering analysis on OLAP reports to automatically discover the grouping knowledge between different OLAP reports. In particular, the proposed approach highlights this knowledge by using a dendrogram/icicle plot representation and histograms. The data interrelationship is computed by using a data mining approach in

order to find the hidden rules inside the data. Unlike our approach, different reports cannot be composed and changed by the users.

Su *et al.* [93] describe a technical framework to produce a report system based on a three-layer calculating architecture which implements a meta-data mapping, an ETL module and a DW. In particular, the proposed technique uses several different data sources and performs statistical calculation to integrate them. Moreover, it allows to define and extend different report models that are in tabular form or graphically represented. As in [48], multiple reports cannot be composed.

Wojciechowski *et al.* [108] present a web based DW application designed and developed for data entry, data management and reporting purposes. The application was built using Oracle Application Express technology. The proposed application implements two kind of reports in tabular form and chart format. They do not consider relations between the reports.

Anfurrutia *et al.* [4] present a product-line approach to database reporting based on predictability and similarity among reports. In particular, this approach exploits a feature model that provides an abstract and concise syntax to express commonality and variability in a database reporting. Data reports are provided textually. Data are extracted and manipulated as XML files, while the page layout and the style format of data are defined by an XSL formatter.

Another technique is based on reports represented as animated transitions. In particular, Heer *et al.* [47] investigate the effectiveness of animated transitions between common statistical data graphics such as Bar-charts, Pie, and Scatter-plots. Unlike our approach they do not generate a single visualization and do not show relationships between data; data used to visualize the starting graph are the same used for the ending one.

Tableau Software provides VizQL [45], a formal language exploited to describe tables, charts, graphs, maps, and time series visualizations. VizQL is a declarative language that describes the desired image; low-level operations needed to retrieve data, to perform analytical calculations and to map the results into a visual representation are automatically generated by a Query Analyzer. Within a single framework one can view the data with different types of visual representations, making it easier to switch from a display to another, but maintaining the visualizations separated and not connected to each other. Image expressions and visualizations allowed by VizQL can be easily customized and controlled.

Decomposition trees are becoming very popular as a tool for the visualization of OLAP operations, and various approaches have been proposed to explore multidimensional data cubes with hierarchical visualization techniques. For instance, Mansmann *et al.* introduce a class of visual structures called Enhanced Decomposition Tree [71] in which each level of the tree structure is created in a disaggregation step along a chosen dimension, the nodes contain the corresponding sub-aggregates arranged in a graph and the arcs are labeled with their respective values. Various layouts are offered to support various activities of analysis. Data cubes are queried using a visual browser based upon a scheme which has dimensions hierarchies of their granularity levels, thus providing an efficient way to create hierarchical views. Multiple data cubes can be explored in parallel according to shared dimensions. The main area of the visual browser is exploited to show the results of user interaction in a selected visual format. Unlike our approach, the only representation of relationships between graphs is limited to the hierarchical links resulting by the aggregation or disaggregation steps.

5.1 Evaluation of Graphical Representations

In the literature, several research efforts have been devoted to the evaluation of graph notations by measuring the effects of different design aspects on user performances and preferences; several approaches can be followed based on diagrams employed, their complexity and the manner in which comprehension can be measured.

A first study [64] investigated people preferences for graphical displays in which 2-dimensional data sets are rendered as dots, lines, or bars with or without 3D depth cues. A questionnaire was used to probe participants' attitudes towards a set of graph formats. It was based on the description of presentation scenarios and the user had to choose the graphical representation which, on his opinion, better describes the information to present in a specific scenario.

Schonlau and Peters [85] assessed that adding the third dimension has no effect on the accuracy of the answers for the bar chart and has a small but significant negative effect for pie-charts. The evaluation was conducted proposing a series of graphs on the web. Participants displayed the graph on the PC screen and, successively, answered several questions for assessing the graph recall. Accuracy of the answers was measured in terms

of the percentage of correct answers.

Inbar *et al.* compared the minimalist and non-minimalist versions of bar and pie-charts also considering the appeal of the notations [53]. The evaluation was based on a questionnaire aiming at collecting the preferences expressed by the participants. Results revealed a clear preference of non-minimalist bar-graphs, suggesting low acceptance of minimalist design principles.

Bateman *et al.* evaluated whether visual embellishments and chart junks cause interpretation problems and should be removed from information charts [7]. To this aim they compared interpretation accuracy in two phases: (i) when examining the representation and (ii) when estimating long-term recall for plain and Holmes-style charts (minimalist and embellished, respectively). Results revealed that people's accuracy in describing the embellished charts was no worse than for plain charts, and that their recall after a two-to-three week gap was significantly better. Accuracy was assessed in both short and long term recall by an experimenter which scored it in a scale from 1 ('All incorrect' or 'I do not know') to 3 ('All correct').

Quispela and Maes [78] investigated the user perceptions of newspaper graphics depending on construction type (standard or non-standard) and mode of expression (pictorial or abstract). They considered two different kinds of users: graphic design professionals and laypeople. Participants rated the attractiveness and clarity of data visualizations. Also the comprehension of graphics based on two-alternative forced-choice tasks was assessed, together with the response time. Comprehension is computed in terms of the correct answers provided.

Albo *et al.* [2] evaluated three radial visualization solutions (including the radar chart) for composite indicators with time-oriented data. Also in this case the information provided is the same with different representations. A questionnaire was proposed to assess the understandability. Performances were measured in terms of accomplishment time and error rate. Also the perceptions of participants were collected.

A cyber security dashboard helping network analysts in the identification and summarization of patterns within the data has been proposed in [75]. Three different type of diagrams have been introduced: (i) location map based on a Dorling cartogram, (ii) temporal chart and heatmap, (iii) attribute bullet bar charts. The usability of the tool and the user perception on the information provided have been assessed.

Correll and M. Gleicher [22] evaluated four graphical representation for comprehension and error. Each prioritizes a different aspect such as cumulative density function or probability density function. Also a controlled experiment which examines pairs of digram types has been conducted.

Correll and M. Gleicher [22] evaluated four graphical representations for comprehension and error. Each prioritizes a different aspect such as cumulative density function or probability density function. Also a controlled experiment which examines pairs of diagram types has been conducted.

In the Software Engineering field controlled experiments have also been adopted to assess models effectiveness. As an example, in [26] a controlled experiment was conducted to evaluate the comprehension level in understanding UML class diagrams with respect to E-R diagrams. The comparison between E-R and class diagrams assesses the comprehension of the same information with different notations, because both the models make use of relationships. The questionnaire consisted in 5 multiple choice questions where each question had one or more correct answers. Comprehension and Effort have been evaluated. In particular, Comprehension is computed by using information retrieval measures (precision, recall and F-measure). To better assess the effect of Method (E-R or Class Diagram) other factors (called co-factors) which may impact the results achieved by the subjects have been considered, such as the effects of the participants' ability and the order of the sessions.

Fish *et al.* evaluated how the addition of constraints and links, respectively, on the information provided reflects on the comprehension of the phenomenon to be examined. In particular, they evaluated whether the adoption of syntactic constraints called Well Formedness Conditions (WFCs) in Euler Diagrams reduces comprehension errors [40]. The participants had to fill in a questionnaire showing several diagrams together with a number of related assertions. The comprehension was assessed considering the number of correct answers provided. Constraints restrict the way diagrams are drawn.

The considered approaches can be classified based on the following criteria, selected by adopting consolidated practices in the area of empirical software engineering [107]:

- *Graph type.* Types of graphs examined during the evaluation, e.g., pie-charts, histograms, scatterplots.

- *Kind of evaluation.* The characteristics examined/compared in the evaluation, e.g., 2D versus 3D representations.
- *Evaluation type.* Classified as: *notation*, when the considered visualization approaches propose the same information with different representations (e.g., pie-charts and histograms); *aesthetic*, the adding/subtracting of visual cues on the same representation type; *information*, when the examined approach differs in the information provided; *relationship*, when the representations add logical relationship between data.
- *Evaluation task.* The task typology adopted for conducting the evaluation, e.g., comprehension task.
- *Evaluation approach.* The approach adopted to perform the evaluation, e.g., controlled experiment, user perception.
- *Evaluation criteria.* The measures used in the assessment:
 - (a) user perception (a qualitative evaluation of the user attitude towards the given representation),
 - (b) accuracy,
 - (c) effort (time needed to accomplish a task),
 - (d) comprehension,
 - (e) correctness.
- *Participants.* The kind of users involved in the evaluation tasks, e.g., students, professionals, etc.

The analyzed methods have been classified by using the criteria above reported (see Table 5.1).

Table 5.1: Classification of existing techniques for the evaluation of graphical representations.

Paper	Kind of comparison	Evaluation type	Evaluation task	Evaluation approach	Evaluation criteria					Participants
					a	b	c	d	e	
[64]	2D vs. 3D depth cues	aesthetic	perception questionnaire	qualitative evaluation	•					undergraduates
[85]	table vs. 2D vs. 3D depth cues	notation-aesthetic	comprehension questionnaire	quantitative evaluation		•				respondents of a web survey panel
[53]	minimalist vs. non-minimalist versions	aesthetic	questionnaire	user perception evaluation	•					students
[7]	visual embellishments vs. chart junk	aesthetic	experimenter-coded response scores, eye-gaze data, and a preference questionnaire	quantitative and perception assessment	•					graduated and students
[78]	standard vs. non-standard charts, pictorial vs. abstract expression mode	notation-aesthetic	perception and comprehension questionnaire	controlled experiment	•		•	•		professionals and laypeople
[26]	E-R vs. class diagram	notation	comprehension questionnaire	controlled experiment			•	•		students
[22]	bar chart with error bars, box, gradient and violin plots	information and notation	comprehension questionnaire	controlled experiment			•	•		North American Turkey population
[2]	three different radial solutions for composite indicator	notation	media survey comprehension questionnaire	controlled experiment	•		•			students
[75]	location vs. temporal view	information	perception questionnaire, think-aloud session	usability study	•					network analysts and managers
[40]	well-formed vs. non-well-formed Euler diagrams	notation-information	comprehension questionnaire	controlled experiment				•	•	undergraduates

(a) user perception, (b) accuracy, (c) effort, (d) comprehension, (e) correctness.

Chapter 6

Complex Graphs generated by the CoDe Methodology

Graphic representation of scientific data, or of information obtained by scientific observations and processing, has been widely investigated [9, 14, 98]. Often the graphic representation of information has been confined to specialist areas each one developing its own professional communication techniques. Less attention has been devoted to visualizations of complex and heterogeneous information variously interconnected with each other. Typical examples are visualizations of statistical data [47, 99].

As a result of the increased interest in human-computer interaction to enhance the usability and friendliness of computing systems, visualization techniques of scientific information are recently getting growing attention. Moreover, modern decision-making requires a greater skill to quickly acquire variously interrelated data rather than single and isolated information. Cognitive maps summarizing relationships without loss of details are useful supports [51, 77].

6.1 Modeling and Generation of CoDe Graphs

The CoDe language allows one to design visualizations that compose and aggregate several charts. The user represents the architectural structure of the visualization at a metalevel by means of graphic terms and relations, which form the CoDe model. A CoDe model is designed by specifying the relationships among the information items to be displayed. The choice of a bar chart, a pie chart or other dashboard types does not influence the semantic relationships among the data themselves. The Code Graph Generator tool executes the

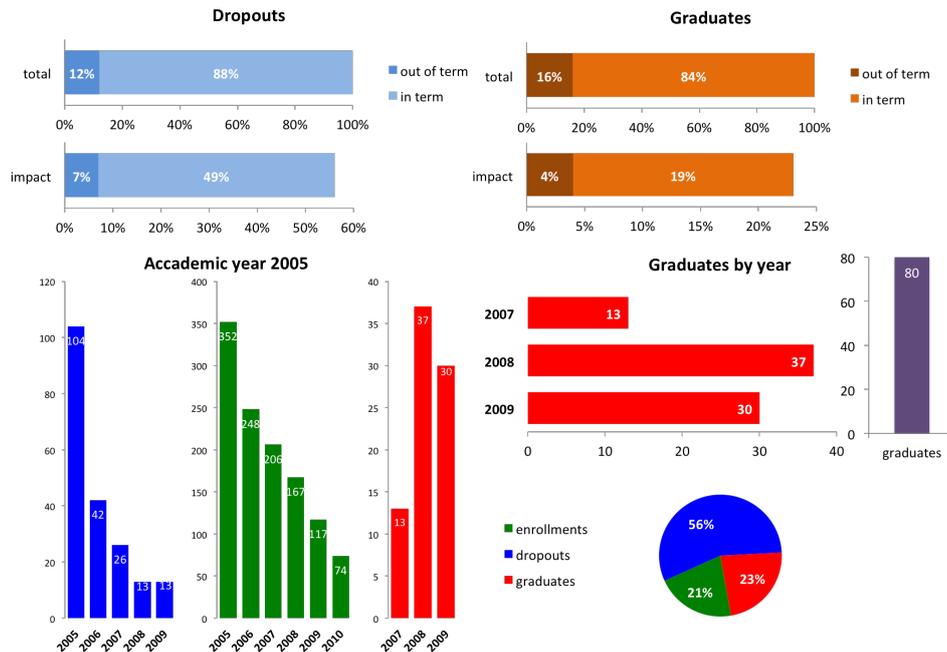


Figure 6.1: Examples of dashboard diagrams.

CoDe model and produces a CoDe Graph coherent with the type of charts selected by the user. For a detailed description of the CoDe methodology see [83].

In the following we recall the concept of CoDe model and of the related CoDe Graph by means of some examples (Section 6.1.1); the CoDe Graph Generator tool is presented in Section 6.1.3.

6.1.1 Code Models and CoDe Graphs

Let us start by showing an example of use of the CoDe methodology for describing a CoDe Graph obtained from the DW of a university, composed by three data-marts concerning staff, students and accounting. In particular, we considered the information related to the students' careers (i.e., information on enrollments, grouped by academic year, faculty, course of study, student origins, gender, secondary school type and grade, dropouts, exams and credits obtained per academic year, grades and duration of studies).

Figure 6.1 shows a sample visualization of data available in the selected DW by means of the traditional dashboard diagrams, and Figure 6.2 depicts the CoDe Graph on the same data. Both these figures show the trend of the student cohort dataset analyzed in [83] (i.e.,

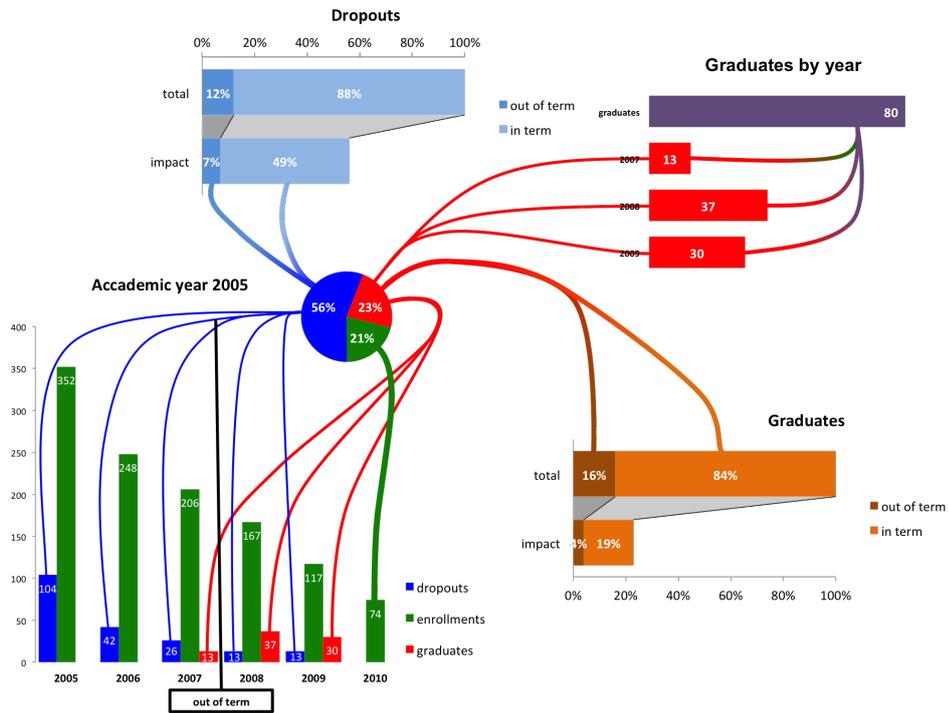


Figure 6.2: Example of a CoDe Graph.

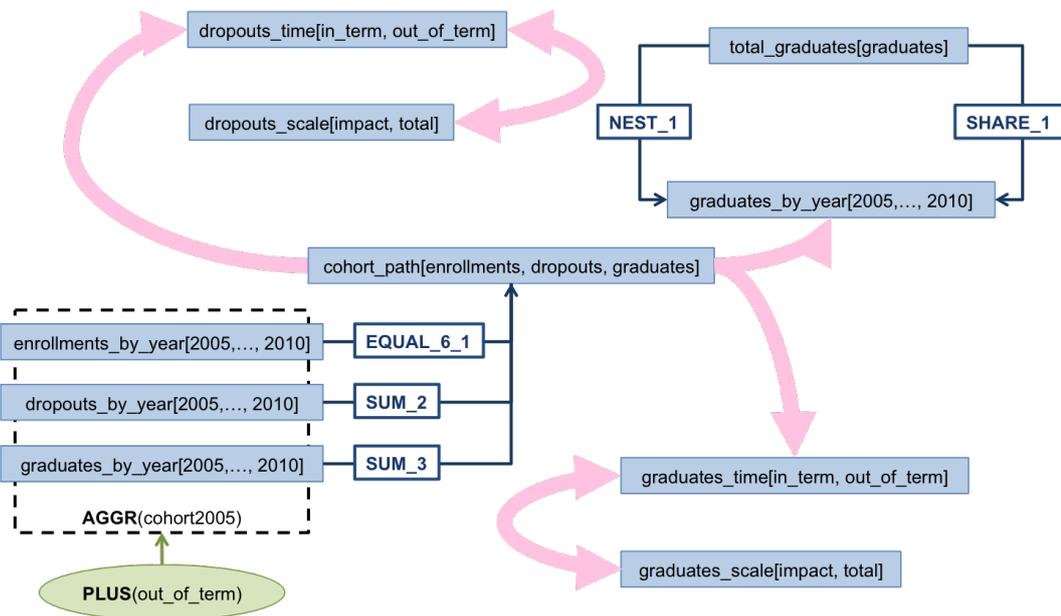


Figure 6.3: The CoDe model of the CoDe Graph in Figure 6.2.

Table 6.1: Data values of terms on graduates.

<i>graduates_by_year</i>						<i>total_graduates</i>
2005	2006	2007	2008	2009	2010	<i>graduates</i>
null	null	13	37	30	null	80

all students enrolled in 2005) subdivided into three categories (i.e., enrollments, graduates and dropouts) evaluated in the six years following the enrollment (the first three years concern the course degree period, and the remaining three years concern the out-of-term period). Since the dataset values are dated March 31, 2011, the 2010 data series contains only information about enrollments.

The CoDe Graph in Figure 6.2 is generated by the CoDe model depicted in Figure 6.3. It is important to note that a CoDe model expresses semantic relationships among data, independently of specific chart types (e.g., pie chart or histogram): starting from a CoDe model different CoDe graphs can be generated.

The rectangles in Figure 6.3 are named *Terms*; a term has n components, each one representing a data series.

In order to better explain the main ideas underlying the CoDe methodology, let us focus our attention on Figure 6.4, where the top right of Figure 6.3 and its corresponding CoDe Graph (the top right of Figure 6.2) are depicted (Figure 6.4(b) and Figure 6.4(a), respectively). The term *graduates_by_year* has six components $\{2005, \dots, 2010\}$. The components 2005, 2006 have no data, since the degree program lasts three years and students may get the degree only from 2007 on. Table 6.1 shows the data tables associated to the considered terms.

Assume that the user chooses the horizontal bar chart for both the CoDe terms when executing the CoDe terms depicted in Figure 6.4(b). The effect of the function *NEST_1* is that of distributing the component 1 (i.e., *graduates*) of the term *total_graduates* (from which the left-side arrow departs) on all components of the *graduate_by_year* term. Moreover, the application of the function *SHARE_1* adds the component *graduates* of the *total_graduates* term on the first position of the horizontal bar chart generated for the *graduate_by_year* component. The generated CoDe Graph is shown in Figure 6.4(a).

Figure 6.5(a) shows the bottom part of Figure 6.3, while the generated CoDe Graph is shown in Figure 6.5(b). The function *AGGR(cohort2005)* is applied to the terms

dropouts_by_year, *enrollments_by_year*, and *graduates_by_year* so generating the fusion of the three vertical bar charts (denoted by the colors blue, green and red, respectively) corresponding to those terms, as depicted in Figure 6.5(b). This function groups its input terms into a single term preserving the distinct identities of the related data series.

The *cohort_part* term is represented by the pie chart in Figure 6.5(b). The function *EQUAL_6_1* lets the 6-th component (2010) of the *enrollment_by_year* term be equal to the first component of the *cohort_part* (the green pie slice), while *SUM_2*, respectively *SUM_3*, lets the second, respectively the third, component of *cohort_part* be the sum of *graduates_by_years* and *enrollments_by_year*, respectively. To this aim, the area of the *dropouts* (respectively *enrollments*) slice of the pie chart is equal to the sum of the areas of all the bars representing dropouts (respectively enrollments). In addition, Figure 6.5(b) shows the text "out of term" included into a rectangle with an associated vertical line. It has been created by the *PLUS* operator in Figure 6.5(a). This operator allows to add vertical or horizontal lines with labels in order to improve the aesthetic impact of the final visualization and to add auxiliary information not given in the represented raw-data.

The existence of a logical relationship between terms is specified by the *LINK* relation, represented by a thick pink arrow connecting two terms. Two *LINK* relations are shown in Figure 6.6(a). In particular, the two-way *LINK* relation on the top-right of Figure 6.6(a) defines the graphical relationship between the *dropout_time* term, representing the ratio between out of term and in term student dropouts, and *dropout_scale* term, representing the same ratio computed with respect to the entire cohort of students (i.e., 56%). This information corresponds to the bars on the top of Figure 6.6(b). The one-way *LINK*

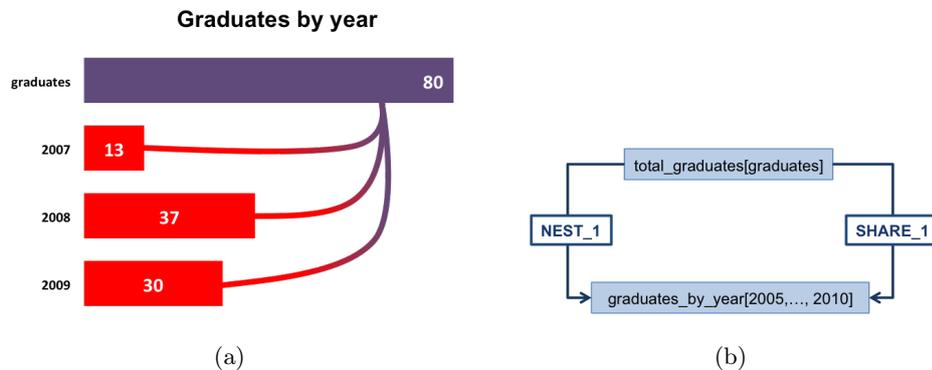
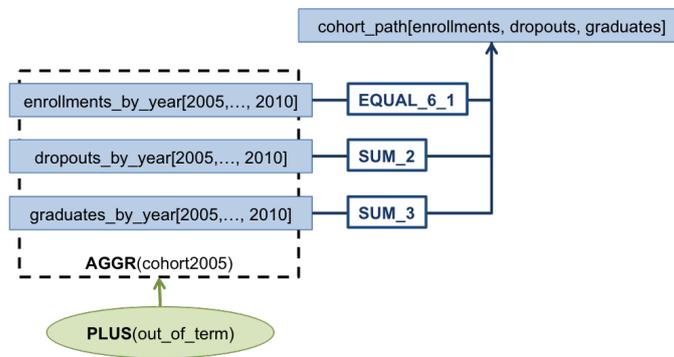
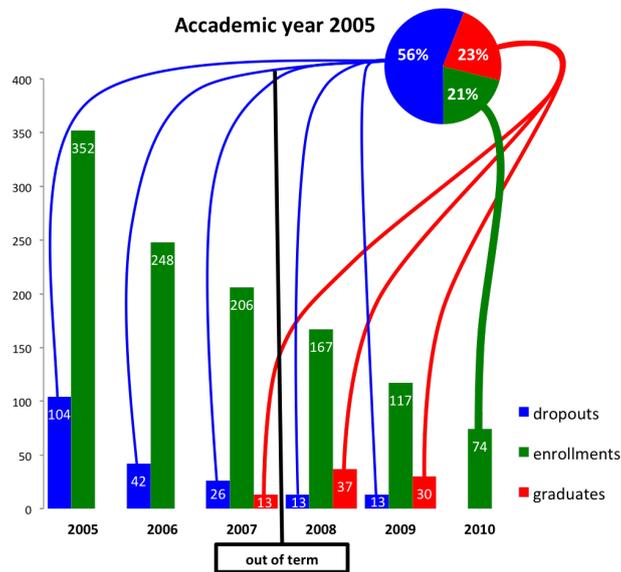


Figure 6.4: The CoDe Graph (a) generated from the CoDe model (b).

relation on the bottom of Figure 6.6(a) defines the relationship between *cohort_path* and *dropout_time* terms, which corresponds to the graphical links from the pie to the bars on the bottom of Figure 6.6(b). This LINK relation is applied to all pairs of components in the terms. In particular, for the pair (*dropouts*, *in_term*) and (*dropouts*, *out_of_term*) there exists a relation represented by a link in 6.6(b). The other pairs, such as (*enrollments*, *in_term*), are not related and, consequently, not connected in the CoDe Graph.

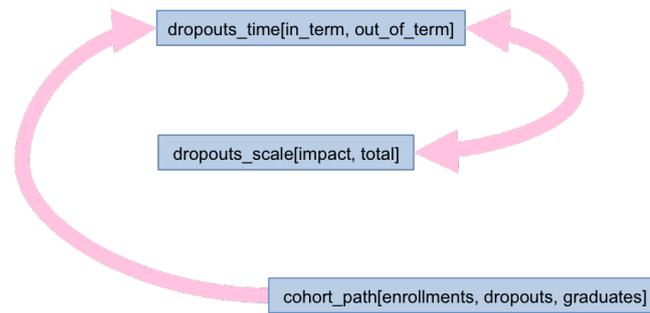


(a)

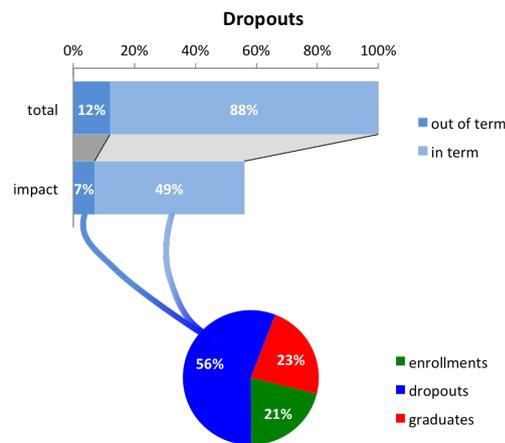


(b)

Figure 6.5: The *AGGR* function (a), and the generated CoDe Graph (b).



(a)



(b)

Figure 6.6: The *LINK* relation (a), and the generated CoDe Graph (b).

6.1.2 Conceptual Organization of Report Visualization: the CoDe Paradigm

A *report* is a *double-entry table* (see Table 6.2), where *title_name* denotes a single information item, each C_i denotes a *category* and $value_i$ its corresponding value. The tuple $[value_1, \dots, value_n]$ is referred to as *data series*.

Following [9], a *graph* is the visualization of the information carried out by a report.

Table 6.2: A double-entry table

<i>title_name</i>		
C_1	...	C_n
$value_1$...	$value_n$

Thus, we call *information item* a graph and its associated report.

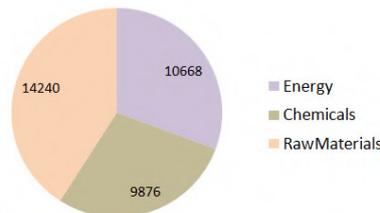
Figure 6.7 shows an information item describing the resources consumption (in terms of energy, chemicals and raw-materials) used by a company. In particular, Figure 6.7(a) depicts the *Resources* report whereas Figure 6.7(b) shows its corresponding graph in terms of a Pie standard graph. According to the FOL paradigm, CoDe represents the information item *Resources* as the logical term shown in Figure 6.7(c).

The CoDe language allows to organize visualizations involving more than one type of graphs that have to be composed and aggregated. Through the CoDe model the user can visually represent information items and their interrelationships at different levels of abstraction keeping consistency between items and ground data. The CoDe model can thus be considered a conceptual map of the complex information underlying the visualization of the ground data.

The process of organizing information visualization consists mainly of four phases (see Figure 6.8): (i) modeling of the cognitive map by describing information items and their relationships through the CoDe model (i.e., *CoDe Modeling*), (ii) definition of information to be extracted in terms of OLAP operations (i.e., *OLAP Operation Pattern Definition*), (iii) extraction of information in tabular form (i.e., *OLAP Operation*), and (iv) visualization of the final reports as a *single image* [9](i.e., *Report Visualization*).

<i>Resources (Companies)</i>		
Energy	Chemicals	RawMaterials
10668	9876	14240

(a) *Resources* report.



(b) *Resources* visualization.

Companies[Energy, Chemicals, RawMaterials]

(c) CoDe term representing *Resources* item.

Figure 6.7: *Resources* information item and its representation in CoDe.

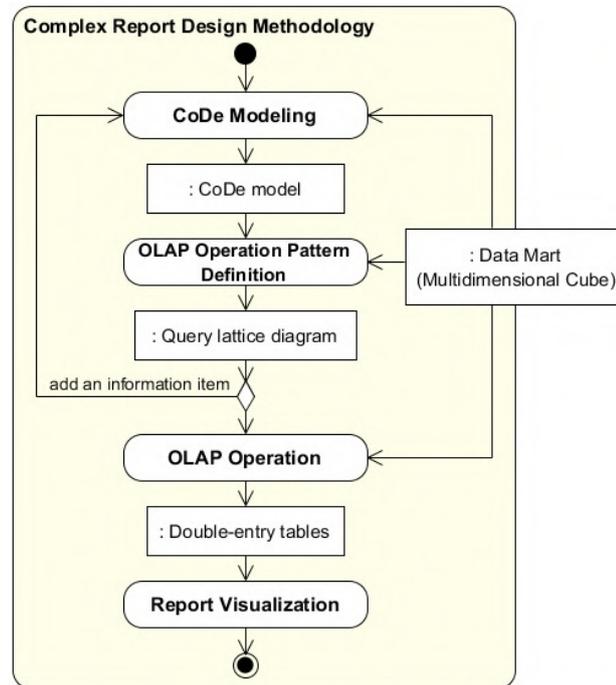


Figure 6.8: The visualization design methodology.

In Figure 6.8 rounded rectangles represent process phases, whilst rectangles represent intermediate artifacts produced at the end of each phase.

The **CoDe Modeling** phase allows to edit the CoDe model representing the cognitive map of information to be visualized.

The **OLAP Operation Pattern Definition** and **OLAP Operation** phases dynamically generate reports corresponding to information items of the CoDe model. These reports are extracted from a data mart represented as a multidimensional cube.

The construction of a report from a data mart maps the cube dimensions on a structure composed by one horizontal axis (corresponding to the components in the report) and one or more vertical axes (corresponding to the data series values). The resulting report is extracted by applying a combination of selection and/or aggregation *slice/dice/pivot/rollup/drilldown* dimensional operators (i.e., the OLAP operations that allow multidimensional data analysis) [17]. We define *operation pattern* the combination of OLAP operations to be performed. Operation patterns are expressed considering only meta-data of the data mart. The actual execution of operation patterns to extract data is performed during the OLAP Operation phase at the end of the design process.

A general example of operation pattern to extract a report from a multidimensional cube is shown in Figure 6.9, where a label in the brackets represents a single OLAP operation, whilst the label h (resp. v) in the parentheses denotes the horizontal (resp. vertical) axis on which the operation is performed (the multiplicity is expressed by the asterisk symbol).

Five steps labeled from (a) to (e) are used to build the report. In particular: (a) $pivot(h \cup v)$ is used to rotate the cube on any dimension, (b) $rollup(h)$ or $drilldown(h)$ are performed on the horizontal axis in order to decrease or increase the details of the report categories, respectively, (c) $dice(h)$ is performed on the horizontal axis to select a subset of dimensional attributes and to exclude the others, (d) $rollup(v)$ or $drilldown(v)$ are similar to (b) but they are performed on the vertical axis, and (e) $slice(v)$ is executed on the vertical axis to reduce the number of selected dimensions.

Figure 6.10(a) shows the multidimensional cube (with four dimensions: *Companies*, *Resources*, *Locations*, and *Pollution*) providing information about the production by companies located in Italy with respect to resources employed and pollution produced,

Information item	OLAP operation pattern
CoDe term	$pivot(h \cup v) * [rollup(h) drilldown(h)] * dice(h) [rollup(v) drilldown(v)] * slice(v)$

Figure 6.9: A general operation pattern describing OLAP operations to extract a report.

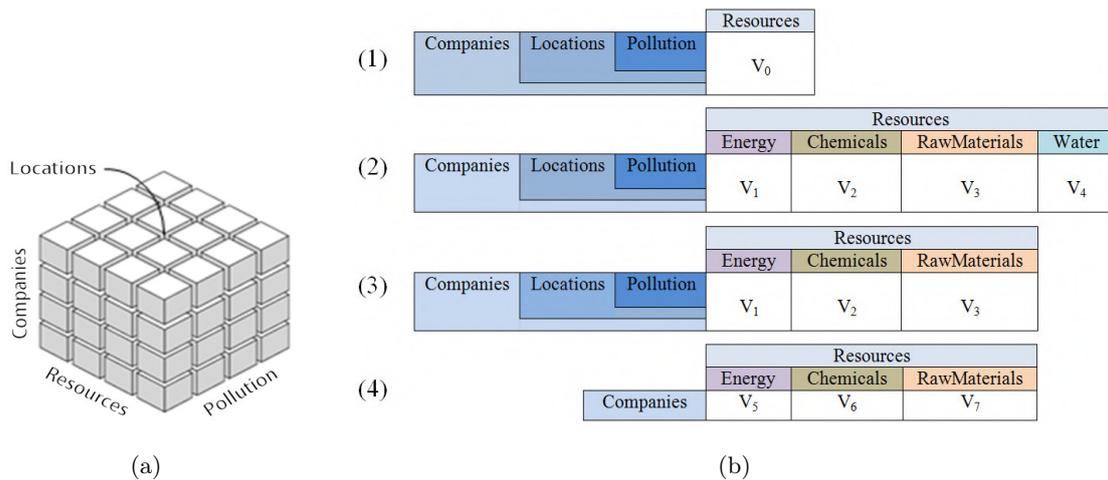


Figure 6.10: A multidimensional cube (a) and OLAP operations (b) performed to extract the report in Figure 6.7(a).

whereas Figure 6.10(b) shows the application of a specific instance of the operation pattern given in Figure 6.9 to extract the report displayed in Figure 6.7(a).

The first OLAP operation is *pivot*, which rotates the cube to place the *Resources* dimension on the horizontal axis. Remaining dimensions of the cube are considered on the vertical axis (see subfigure (1) of Figure 6.10(b)). A *drilldown* operation is performed on the horizontal axis (i.e., *Resources*) in order to increase the details (see subfigure (2)). A *dice* operation on the horizontal axis is then performed to select a subset of *Resources* attributes by excluding the *Water* attribute (see subfigure (3)). Since *Companies* represents the final dimension attribute to be computed, *rollup* or *drilldown* operations are not needed on the vertical axes (i.e., *Companies*, *Locations* and *Pollution*) to aggregate data or to increase the details, respectively. Finally, a *slice* operation on the vertical axis reduces the dimensions to *Companies* (see subfigure (4)) and produces the report.

In order to specify the members on which the OLAP operation is performed, each OLAP operation can specify a parameter m that lists the set of members, separated by commas, involved in that operation. The symbol ; is used to separate the list of members from the axes specification. Thus, $pivot(h \cup v; m)$ is used to rotate the dimensional members m on any single dimension, $rollup(h; m)/rollup(v; m)$ or $drilldown(h; m)/drilldown(v; m)$ are performed on the horizontal/vertical axis in order to decrease or increase the details of the set m , respectively, while $dice(h; m)$ is performed on the horizontal axis to select a subset of dimensional members m and to exclude the others (if present), and $slice(v; m)$ is executed on the vertical axis to reduce the number of selected dimensional members to

Table 6.3: Summary of the mapping among Code syntax and OLAP operation patterns.

CoDe syntax	OLAP operation pattern
<i>CoDe term</i>	$pivot(h \cup v)^* [rollup(h) drilldown(h)]^* dice(h) [rollup(v) drilldown(v)]^* slice(v)$
<i>SUM_i</i>	$rollup(h)^* pivot(h_i \cup v) dice(h_1 \dots h_i \dots h_n)$
<i>NEST_i</i>	$slice(h_1 \dots h_{i-1} h_{i+1} \dots h_n) drilldown(h_i)^*$
<i>EQUAL_{i_j}</i>	$slice(h_1 \dots h_{i-1} h_{i+1} \dots h_n) pivot(h_j \cup v) dice(h_1 \dots h_j \dots h_m)$
<i>SHARE_i</i>	$\forall j = 1, \dots, n \quad slice(h_1 \dots h_{j-1} h_{j+1} \dots h_n) pivot(h_i \cup v) dice(h_1 \dots h_i \dots h_m)$
<i>AGGR/UNION</i>	$pivot(h \cup v)^* [rollup(h) drilldown(h)]^* dice(h) [rollup(v) drilldown(v)]^* slice(v)$
<i>LINK</i>	$\forall j = 1, \dots, n \quad slice(h_1 \dots h_{j-1} h_{j+1} \dots h_n) drilldown(h_i)^*$

To ease the readability the OLAP operation patterns, we have omitted the set of members on which the OLAP operations is performed.

the ones in m .

The summary of the mapping among Code syntax and OLAP operation patterns is shown in Table 6.3, where each label represents a single OLAP operation, the symbol h (resp. v) in the parentheses denotes the horizontal (resp. vertical) axis on which the operation is performed, and the multiplicity is expressed by the $*$ symbol.

A careful reader could observe that the order of OLAP operations performed to obtain the same report may not be unique. The OLAP operation patterns allow to organize the visualization at a suitable abstract level without taking into account the ground data. In other words, the effective queries are a consequence of the visualization design phase, and not vice versa, since they are applied when the CoDe model design activity ends.

The OLAP Operation Pattern Definition phase produces a set of operation patterns able to generate reports corresponding to CoDe terms defined in the CoDe model. Since OLAP operations can share data, we can find a partial order between these operation patterns. We thus organize the set of operation patterns as a *query lattice* structure [46,66] to determine in what order OLAP operations have to be executed.

More precisely, considering a multidimensional cube C , with d dimensions, let us denote with op_A and op_B two OLAP operation patterns applied on C that respectively provide two dimensional attribute tuples $A = (a_1, a_2, \dots, a_d)$ and $B = (b_1, b_2, \dots, b_d)$, where each a_i and b_i are attributes in the hierarchy for the i -th dimension. Then, a partial ordering \preceq on the set Q of the OLAP operation patterns can be defined by setting $op_A \preceq op_B$

if and only if for any i , $1 \leq i \leq d$, the dimensional attribute a_i in A can be computed by OLAP operations given in op_A applied on the attributes in B .

The partial ordering \preceq allows to define a *lattice* on the set Q since any couple of OLAP operation patterns has a least upper bound and a greatest lower bound. Moreover the empty operation pattern, which corresponds to the overall DW is the top element with respect to the partial ordering relation \preceq [46].

As an example, let $op_{Resources}$ and op_{Energy} be the two OLAP operation patterns respectively providing report *Resources* in Figure 6.7(a) and report *Resources.Energy* in Figure 6.11. The latter specifies resources employed by companies in terms of energy consumption. The related attribute tuples are in the following hierarchical relation:

$$(Companies, \mathbf{Energy}, none, none) \preceq_d (Companies, \mathbf{Resources}, none, none) \quad (6.1)$$

<i>Resources.Energy (Companies)</i>			
Diesel	Electricity	Fuel	Methane
762	5715	423	3768

Figure 6.11: The *Resources.Energy* report.

Since *Pollution* and *Locations* dimensions are not considered in the reports we specify *none* attributes in the relation (6.1).

The operation pattern $op_{Resources}$ is described in Figure 6.10(b). In order to define the operation pattern op_{Energy} we can start from the attribute tuple provided by $op_{Resources}$. A *dicing* operation allows to select *Energy* from *Resources* attributes, then a *drilling* operation increases details providing the actual data of the *Resources.Energy* report.

We thus can assert that $op_{Energy} \preceq op_{Resources}$ and that the computation of op_{Energy} made from the results of $op_{Resources}$ reduces the number of OLAP operations performed on the multidimensional cube.

It is worth noting that CoDe modeling does not negatively affect the OLAP query implementation, thus it does not degrade the performances. Moreover, the CoDe modeling of graph composition exploiting the query lattice structure allows to optimize the OLAP operations.

The CoDe model is defined iteratively, thus the approach allows an incremental construction of the visualizations. In fact, the first two steps of the design process can be repeated whenever the designer adds an information item or a relationship to the CoDe model. This feature enhances the design process of the visualization.

The OLAP Operation phase extracts data from the multidimensional cube through the OLAP operation patterns organized in the *lattice* structure. OLAP processing could be slow so the use of the *lattice* structure improves the performance reducing the total number of OLAP operations to be performed. In fact, by exploiting the partial ordering relation between OLAP operation patterns we compute the results of some OLAP operation patterns starting from the results of another one. Thus we find the minimal set of OLAP operations generating the required OLAP operation patterns. This set allows to reduce the access to raw data in order to provide the required reports [46].

The **Report Visualization** phase displays the visualization of extracted reports and their relationships according to the CoDe model. In particular, the visualization is im-

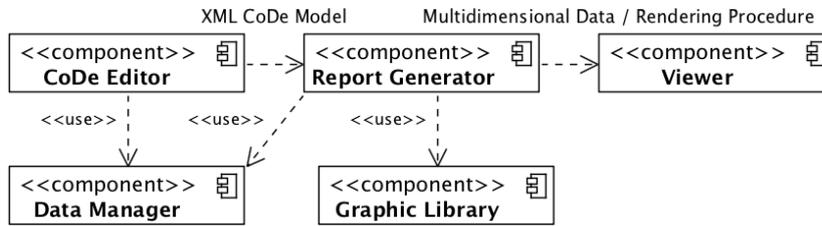


Figure 6.12: The Complex Graph Generator tool architecture.

plemented taking into account the units of related data series in order to preserve the ratio between the quantitative data extracted from the multidimensional cube. During this phase the designer selects the type of standard graph to draw the reports and places them in specific locations of the drawing area. Moreover, additional information labels or visual symbols can improve visualization details.

6.1.3 Generating CoDe Graphs

The CoDe Graph Generator tool has been developed as an Eclipse plug-in. It generates CoDe Graphs (e.g., like the one shown in Figure 6.2), given a CoDe model (e.g., like the one shown in Figure 6.3). In the following, we outline how the tool automatically generates and visualizes them. In particular, we detail how the layout of a CoDe Graph is structured, how its components are dimensioned taking into account the component relevance in terms of its quantitative measure and the numerical proportion among data; also overlapping avoidance is considered. The architecture of the CoDe Graph Generator tool is depicted in Figure 6.12 and consists of the following components:

- The *CoDe Editor* is the graphical environment that enables the manager to design CoDe models and to select the standard graph types of each term and other visualization properties, e.g., colors, texts and numeric data formats.
- The *Data Manager* accesses the DW schema and provides the CoDe Editor with the terms needed to design a CoDe model. The CoDe Editor outputs the XML description of the CoDe model, which provides information about the logical coordinates (i.e., the standard graph type associated to each term in the CoDe model, the xy -position of the graph computed considering the position of the related term in the

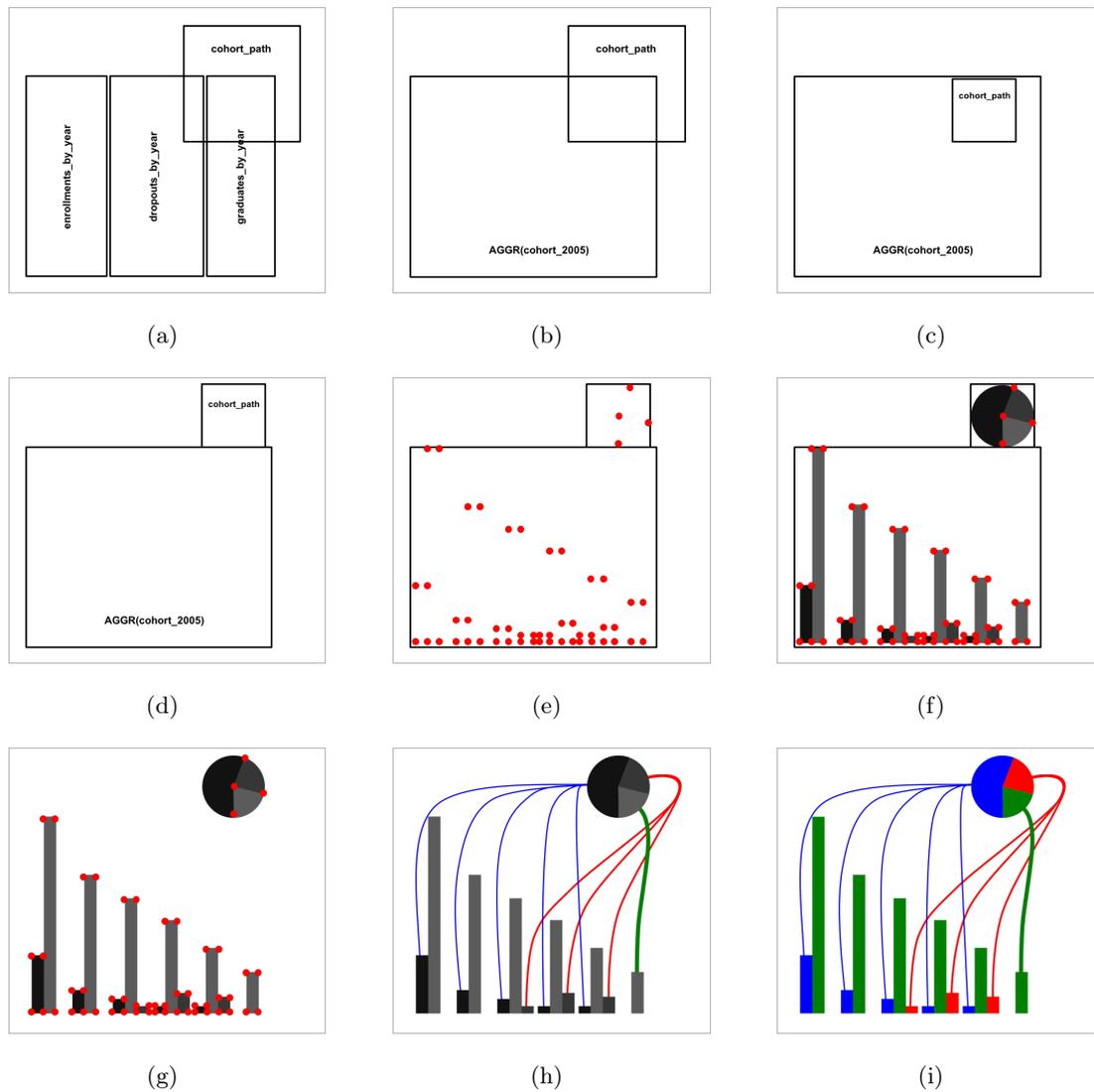


Figure 6.13: Rendering of the CoDe Graph in Figure 6.5(b).

editing area, the scaling factor of the graph and the OLAP query to extract the data from DW).

- The *Report Generator* gets the XML description, uses the OLAP queries to obtain the multidimensional data from the Data Manager and visualizes the CoDe Graph. The Report Generator converts these data into a JSON format and saves them into a data file. Moreover, it creates another file containing the rendering procedure (written in Javascript) for drawing the CoDe Graph by exploiting the features (i.e.,

graphical primitives and charts) of the CoDe *Graphic Library*.

- The *Viewer* (e.g., the web browser) runs the rendering procedure to visualize the CoDe Graph starting from the data file.

Figure 6.13 shows behavior of the rendering procedure when generating the CoDe Graph from the CoDe model. The entire process is automatic, except for some aesthetic final refinements performed with the user intervention. The rendering procedure is composed of 6 steps.

1. For each term in the CoDe model, compute the size of the corresponding graph by exploiting the data extracted by the OLAP query and the scaling factor associated to it. To this aim, the key-points of a chart are computed (e.g., the vertices of a bar in an histogram) in order to represent its sub-parts. These points are used to determine the bounding box of the whole graph. Figure 6.13(a) shows the four bounding boxes of the terms in Figure 6.5(a). Those boxes are positioned taking into account the logical coordinates of the terms in the CoDe model.
 - Each AGGR function in the CoDe model is considered as a single term. As an example, in Figure 6.13(b) the three terms of *AGGR(cohort2005)* are grouped into a single box.
 - If a SHARE function is applied to two terms, the shared sub-part is added to the chart of the second term. The size and the bounding box are updated accordingly.
2. For each function EQUAL, SUM or NEST between two terms in the CoDe model, the dimensions of the charts representing the terms are computed as follows: a scaling factor between the charts is computed by considering the areas of the sub-parts involved in the function. In particular, the area of the chart in the right side of the function becomes equivalent to the area of the chart in the left side of the function. Figure 6.13(c) shows how the pie is resized: the EQUAL function is selected and the scaling factor is computed in such a way that the pie slice of *cohort_part[enrollments]* has the same size of *enrollments_by_year[2010]*.

If the same chart is involved in more than one function, different scaling factors are computed and it is chosen the factor relative to the EQUAL function; otherwise, the

- average of the factors is considered. The bounding boxes and the key-points are resized by considering the computed scales.
3. If there exists any overlapping among charts, the center of the CoDe Graph is computed taking into account all the bounding boxes. All the charts are moved along the radial direction with respect to the center until they do not overlap anymore (see Figure 6.13(d)).
 4. The charts are drawn (see Figures 6.13(e), 6.13(f) and 6.13(g)).
 5. The functions EQUAL, NEST, SUM, and LINK are drawn by adopting B-spline functions [5] on the key-points of the charts (see Figure 6.13(h)). A path detection algorithm avoids (or minimizes) overlapping among charts and links.
 6. Final graphical refinements are added:
 - The graphical objects are colored taking into account the related items (see Figure 6.13(i)).
 - Graphical symbols, icons (specified through the PLUS operator), and labels are added to the CoDe Graph taking into account the terms to which they are related. Their position is user defined; for example, the user has to specify "*center-x, bottom-y*" to place the box generated for the PLUS(out_of_term) as in Figure 6.5(b).
 - Chart elements like legend, chart title and axes are added by the user.

Chapter 7

User Comprehension of CoDe Graph Reports

A wide set of tools is already available for visual representation of tabular reports by means of standard graphs as histograms, pies, etc. However, these tools usually only allow representation through separated graphs and cannot visualize relationships between information contained in different reports [82]. Starting from the main concepts of the CoDe methodology [83] and presenting the generation process implemented in the CoDe Graph Generator tool, there exists a need to evaluate the comprehensibility of CoDe Graphs with respect to dashboard reports.

The research activity tries to answer the following research question: *in the context of decision support, to what extent a report generated by the CoDe Graph Generator tool does impact user comprehension?* To answer this question, we need to validate how the graphical reports (CoDe Graphs) generated by Code Graph Generator tool on data extracted from data warehouses improves the interpretation of phenomena underlying the data.

7.1 Definition and Planning of the Empirical Evaluation

In order to evaluate the advantages of using CoDe Graphs, we conducted a controlled experiment involving participants to assess the comprehension of CoDe Graphs with respect to standard dashboard diagrams.

The design and planning of the experiment has been structured according to the guidelines of Juristo and Moreno [57], Kitchenham *et al.* [61], and Wohlin *et al.* [107]. By Applying the Goal Question Metric (GQM) template [6]

- *Analyze* {the name of activity or attribute}

- *for the purpose of* {overall goal}
- *with respect to* {the aspect to be considered}
- *from the viewpoint of* {interested people}
- *in the context of* {environment}

our goal can be defined as:

- *Analyze* dashboard and CoDe Graphs
- *for the purpose of* evaluating their use
- *with respect to* the comprehension of data correlation and the effort
- *from the viewpoint of* a decision maker
- *in the context of* a university by involving students with management and data-warehouse knowledge.

7.1.1 Context Selection

The context of the experiment is the comprehension of graphical representations to enable instantaneous and informed decisions. The study was executed at the University of Salerno, Italy. The participants were 47 Computer Science master students which passed the exams of Management, Information Systems and Database Systems. All of them were from the same class with a comparable background, but different abilities. Before conducting the experiment, the students were asked to fill in a pre-questionnaire aiming at collecting their demographic information. A quantitative assessment of the ability level of the students was obtained by considering the average grades they got in the three exams cited above. In particular, students with average grades below a fixed threshold, i.e., 24/30¹ were classified as *Low Ability*, while the other students were classified as *High Ability*; then, 11 students were classified as Low Ability, whilst 36 students were classified as High Ability.

¹In Italy, the exam grades are expressed as integers and assume values between 18 and 30. The lowest grade is 18, while the highest is 30.

Participants were informed that their performance in that experiment would not have any effect on their grades in any courses; moreover, they were also informed that data collected during the experiment would only be used for research purposes, confidentially treated, and shared in an anonymous and aggregated form.

During the experiment participants had to analyze data representations (or experimental objects) from the following data-warehouses:

*DW*₁. the University data-warehouse, described in Section 6.1,

*DW*₂. the Foodmart data-warehouse, providing information about sales of food products in the United States, Mexico and Canada from 1997-1998. Also customer information are included.

7.1.2 Variable Selection

We considered the following independent variable:

Method: the treatment of this experimentation. It can assume as value CG or DG (i.e., *CoDe Graph* or *Dashboard Graph*). DG is used for comparison (control group).

Several different variables can be analyzed when comparing two different graphical approaches, as discussed in Chapter 5. In particular, we are mainly interested in verifying the support offered in terms of Comprehension, which is accurately evaluated in [26] and [40] through a controlled experiment. In [26] Comprehension is assessed in terms of effectiveness (*F-Measure*) and effort (*time*). We follow that direction by also considering Efficiency. Thus, to analyze the participants' performances, we selected three dependent variables: *Effort*, *Effectiveness* and *Efficiency*.

Effort and task completion time

To assess the *Effort* variable we used task completion time. Considering the time as an approximation for effort is customary in the literature and it is compliant with the ISO/IEC 9126 standard [55] definition: effort is the productive time associated with a specific project task. Other aspects that may be related to effort, for example, the cognitive effort of participants, were not measured. Task completion time indicates the number of minutes to fill in the Comprehension Questionnaire and then it assumes only positive integer values.

Low values for task completion time mean that participants spent less time (or less effort) to complete a comprehension task.

Effectiveness and task effectiveness

Effectiveness is evaluated by considering the task effectiveness in terms of the answers to the Comprehension Questionnaire provided by the participants. In particular, it is computed in terms of *F-Measure* [81] as follows. Given

- n , the number of questions composing a given section of the questionnaire,
- m_s , the number of answers provided by the participant s ,
- k_s , the number of correct answers provided by the participant s ,

two information retrieval metrics are computed: $precision_s = \frac{k_s}{m_s}$ and $recall_s = \frac{k_s}{n}$. Precision and recall measure two different concerns, namely correctness and completeness of the answers, respectively. To balance them, we adopted F-Measure, their harmonic mean (i.e., $\frac{2 \cdot precision_s \cdot recall_s}{precision_s + recall_s}$). This mean has been used to measure the *task effectiveness*. The measures above assume values in the range $[0, 1]$. For this measure, a value equals to 1 means that a participant correctly and completely answered all the task questions.

Efficiency and task efficiency

We also considered the efficiency with which a task was accomplished. The efficiency construct is measured by means of the variable *task efficiency*. It is a measure computed as the ratio between task effectiveness and task completion time and estimates the efficiency of a participant during a comprehension task. In particular, this measure estimates the ability of a participant to effectively comprehend the data report without wasting time:

$$task\ efficiency = \frac{task\ effectiveness}{task\ completion\ time} \quad (7.1)$$

The task efficiency values have been normalized in the range $[0, 1]$. The larger the task efficiency value, the better it is. Efficiency measures task effectiveness achieved in time unit. For example, if a participant $p1$ obtains 1 for task effectiveness and 25 minutes for task completion time, then his/her task efficiency is equal to 0.04. On the other hand, if a

participant $p2$ obtains 0.8 for task effectiveness and 30 minutes for task completion time, then his/her task efficiency is equal to 0.0266. Comparing the task efficiency values for $p1$ and $p2$, we understand that $p1$ has a better efficiency because he/she has got a better trade-off between effectiveness and effort to comprehend the report chart and answer the questionnaire.

Co-factors

Other factors may influence directly dependent variables or interact with the main factor. For this reason, we took into account the following co-factors [26]:

- *Ability*. The participants having a different ability could produce different results.
- *Session*. The participants had to fill-in a two sections questionnaire. Although we designed the experiment to limit the learning effect, it is still important to investigate whether participants perform differently in the two experimental sessions.

7.1.3 Hypothesis Formulation

The objective of our study is to investigate the effectiveness of CoDe Graphs on the user comprehension. To achieve our goal, we propose several null hypotheses, as follows:

- H_{n1} : when performing a comprehension task, the method (CG or DG) **does not significantly improve** the *Effectiveness* (Task Effectiveness) achieved by decision makers.
- H_{n2} : when performing a comprehension task, the method (CG or DG) **does not significantly reduce** the *Effort* (Task Effort) achieved by decision makers.
- H_{n3} : when performing a comprehension task, the method (CG or DG) **does not significantly improve** the *Efficiency* (Task Efficiency) achieved by decision makers.

Alternative hypotheses for H_{n1} (i.e., $\neg H_{n1}$), for H_{n2} (i.e., $\neg H_{n2}$), and for H_{n3} (i.e., $\neg H_{n3}$) admit a positive effect of CG or DG. All these hypotheses are two sided because we could not postulate any effect of CG or DG on the dependent variables.

Table 7.1: Experimental Design.

	Group 1	Group 2	Group 3	Group 4
Session 1	T_1 -DG	T_1 -CG	T_2 -DG	T_2 -CG
Session 2	T_2 -CG	T_2 -DG	T_1 -CG	T_1 -DG

7.1.4 Experiment Design, Material, and Procedure

Participants were divided into four groups, making sure that High and Low Ability participants were equally distributed across groups. We assigned an equal number of participants to each group. The experiment design is of the type "*one factor (graph type) with two treatments*", referred to two data representation modalities:

- *DG*. Participants try to comprehend the meaning of data when represented by using traditional Dashboard Graphs;
- *CG*. Participants try to comprehend the meaning of data when represented by CoDe Graphs.

Each group performed the following two tasks, one in each experimental session (Session 1 and Session 2):

- T_1 . Comprehension of graphs related to DW_1 . It represents a comprehension activity for understanding which is the extent of dropouts.
- T_2 . Comprehension of graphs related to DW_2 . It represents a comprehension activity for understanding which are the outcomes in the various categories of shops (supermarket, grocery stores, etc.) in the various countries.

The experiment design is summarized in Table 7.1. We used participants' average grades to distribute participants among groups in this table. In other words, these groups were similar to each other in terms of number of participants with high and low average grade (ability). The design ensured that each participant worked on a different task in a single session, receiving a different treatment for each task. Also, the design allowed us to consider different combinations of task and treatment in different order across the two sessions. We opted for this kind of design because it mitigates possible carry-over effects² as well as the effect of experimental objects in each experiment [107].

²Carryover is the persistence of the effect of one treatment when another treatment is applied later [102].

Before the experiments, the participants have been trained on CoDe Graphs and Dashboard Graphs different from those used in the experiment. Participants spent 10 minutes on the training task.

During the experiment, we assigned the two comprehension tasks T_1 and T_2 to each participant. The tasks were designed in such a way to reflect the typical analyses that a manager has to perform. In particular, participants had to fill in the Comprehension Questionnaire composed of two sections, one for each task. In these two tasks, the participants were presented a graphical description of real data, extracted from the selected data-warehouses by using OLAP operations. The coherence and completeness of the proposed graphs were validated by a data analysis expert of the University of Salerno. Each task proposed a number of related multiple-choice questions. Participants had to indicate the choices that they believed to be correct. When filling-in their questionnaire, the participants annotated the starting and ending time for each session.

The questionnaire for the task T_1 is reported in Table 7.2. It is related to both the graphs shown in Figure 6.1 and Figure 6.2 for the method DG and CG, respectively. It was composed of 6 questions expecting closed answers in the range between 3 and 6.

When the participants ended a session, they had to pass to the successive one. At the end of the experiment, the participants filled in the PostExperiment Questionnaire,

Table 7.2: The Comprehension Questionnaire for DW1.

ID	Question
Q1	How many students enrolled in 2005? <input type="checkbox"/> 352 <input type="checkbox"/> 248 <input type="checkbox"/> 206 <input type="checkbox"/> 167 <input type="checkbox"/> 117 <input type="checkbox"/> 74
Q2	Which is the percentage of off-course graduates on the total amount of graduates? <input type="checkbox"/> 7% <input type="checkbox"/> 16% <input type="checkbox"/> 19% <input type="checkbox"/> 84%
Q3	How many students enrolled in 2010? <input type="checkbox"/> 352 <input type="checkbox"/> 248 <input type="checkbox"/> 206 <input type="checkbox"/> 167 <input type="checkbox"/> 117 <input type="checkbox"/> 74
Q4	How many students had dropped-out from 2005 to 2010? <input type="checkbox"/> 7% <input type="checkbox"/> 12% <input type="checkbox"/> 56% <input type="checkbox"/> 88%
Q5	How many students regularly (in three years) took the degree? <input type="checkbox"/> 13 <input type="checkbox"/> 37 <input type="checkbox"/> 30
Q6	Which is the percentage of regular graduates (in three years) on the total amount of graduates? <input type="checkbox"/> 7% <input type="checkbox"/> 16% <input type="checkbox"/> 19% <input type="checkbox"/> 84%

Table 7.3: PostExperiment Questionnaire.

ID	Question
P1	The questions of the evaluation questionnaire were clear to me.
P2	The task objectives of the were clear to me.
P3	I found useful the relationships of the CG representation for comprehending data.
P4	Understanding of the meaning of the data was problematic when they are represented using CG.
P5	The proportion among sub-graphs of CG is useful for comprehending data.
P6	The arrangement of sub-graphs of CG helps to comprehend data.

reported in Table 7.3. It was composed of 6 questions expecting closed answers scored using the 5-point Likert scale [76] specified in Table 7.4.

The purpose of the PostExperiment Questionnaire was to assess whether the questions and the experiment objectives were clear, and participants' opinion on the CoDe Graphs.

7.1.5 Analysis Procedure

To test the defined null hypotheses, we planned to use parametric statistical tests. When apply parametric tests the normality of data is verified by means of the Shapiro-Wilk test [89]. It tests the null hypothesis that the sample is drawn from a normally distributed population. A p-value smaller than the threshold α allows us to reject the null hypothesis and conclude that the distribution is not normal. We opted for this normality test because of its good power properties as compared to a wide range of alternative tests. In case the p-value returned by the Shapiro-Wilk test is larger than the threshold α , we used an unpaired t-test. We exploited unpaired tests due to the experiment design (i.e., participants experimented CG and DG on two different experiment objects DW_1 and DW_2). In our statistical tests we decided to accept a probability of 5 percent of committing Type I errors, i.e., of rejecting the null hypothesis when it is actually true. Thus, the null hypothesis is rejected when the appropriate statistical tests provide a p-value less than the standard α -level of 5 percent.

Table 7.4: The adopted Likert scale.

1.Strongly disagree	2.Weakly disagree	3.Neutral	4.Weakly agree	5.Strongly agree
---------------------	-------------------	-----------	----------------	------------------

In addition to the tests for the hypotheses formulated in Section 7.1.3, we also evaluated the magnitude of performance difference achieved with the same user group with different methods. To this aim, we evaluated the effect size in several ways. In case of parametric analyses, we used *Cohen's d* [42] to measure the difference between two groups. The effect size can be considered negligible for $d < 0.2$, small for $0.2 \leq d < 0.5$, medium for $0.5 \leq d < 0.8$, and large for $d \geq 0.8$ [58]. For non-parametric analysis we used the *Cliff's Delta* effect size (or d) [42]. The effect size is small for $d < 0.33$ (including negative values), medium for $0.33 \leq d < 0.474$ and large for $d \geq 0.474$.

For parametric tests we also analyzed the statistical power (i.e., post-hoc or retrospective power analysis). To compute the power, we also took into consideration the type of hypothesis tested (one-tailed vs. two-tailed). Whatever is the kind of test, statistical power represents the probability that a test will reject a null hypothesis when it is actually false. Then, it is worthy to be analyzed only in case a null hypothesis is rejected. The highest value is 1, while 0 is the lowest. A high statistical power denotes a high probability to reject a null hypothesis when it is actually false. The value 0.80 is considered as a standard for adequacy [39].

We used interaction plots to study the presence of possible interactions between the main factor and the co-factors (Ability and Sequence). They are line graphs in which the means of the dependent variables for each level of one factor are plotted over all the levels of the second factor. If the lines are almost parallel no interaction is present. On the contrary, there exists an interaction. Cross lines indicate a clear evidence of an interaction between factors.

We adopted histograms to show the distribution of the results of the post-experiment survey questionnaire. In this way, we provided a quick visual representation.

7.1.6 Threats to Validity

In this section, threats that could affect results and their generalization are presented and discussed to better comprehend strengths and limitations of our experiment- Despite our effort to mitigate as many threats as possible, some threats are unavoidable. We discuss the threats to validity following the guidelines proposed by Wohlin *et al.* [107].

Internal Validity

Threats concern factors that may affect our dependent variables and are relevant in studies that try to establish a causal relationship.

- *Maturation.* The adopted experimental design (a kind of cross-over design — Table 7.1) should mitigate the presence of carry-over effects [107]. We also analyzed the effect of the laboratory trials (o sessions) on the response variables. If this effect is present, we properly mention it in our data analysis.
- *Diffusion or imitation of treatments.* This threat concerns information exchanged among participants within each experiment and among experiments.

Experiment supervisors monitored participants to prevent that they communicated with one another. As an additional measure to prevent diffusion of material, we asked participants to return material at the end of each trial.

- *Selection.* The effect of letting volunteers take part in an experiment may influence results. In fact, volunteers could be more motivated.
- *Fatigue effect.* We formulated each task in such a way to require little effort by limiting the number of quantitative and qualitative questions.

External Validity

External validity threats concerns generalizability of results.

- *Interaction of selection and treatment.* The choice of students as participants may affect the generalizability of results [15, 20, 44]. To mitigate this threat, we selected as participants masters students which had passed specific exams and that were accustomed to interpreting Dashboard Graphs. The participants did not know the hypotheses of the experiments and were not evaluated on their results. Working with students implies various advantages, such as the fact the students' prior knowledge is rather homogeneous and there is the possible availability of a large number of participants [103].

- *Interaction of setting and treatment.* In our study, the kind of experimental tasks may affect result validity. For example, differences observed on the kind of experimental objects could not be related to the choice of the method (CG or DG), but to external factors (e.g., the domain of the data or how they are represented with the different methods). It may happen that participants were more familiar with data from the University data-warehouse than those of the Foodmart one. The adopted experimental design should mitigate this aspect.

Construct Validity

This threats concern the link between concrete experimental elements and abstract concepts of experimental goals. Threats to construct validity are also related to experiment design and social factors.

- *Low statistical power.* The power of a statistical test is the ability of the test to reveal a true pattern in the data. If the power is low, there is a high risk that an erroneous conclusion is drawn.
- *Interaction of different treatments.* To mitigate this kind of threat, we adopted a kind of cross-over design and analyzed the possible effect of co-factors.
- *Confounding constructs and level of construct.* The procedure used to divide participants into the groups in Table 7.1 could affect construct validity.
- *Evaluation apprehension.* We mitigated this threat because participants were not evaluated on the results they achieved in the experiments.
- *Hypothesis guessing.* When people take part in an experiment they might try to figure out what the purpose and intended result of the experiment is. We did not mention to participants the objectives and the hypotheses of our experiment.
- *Experimenters' expectancies.* Experimenters can bias results both consciously and unconsciously based on what they expect from an experiment. To deal with this kind of threat we asked a data analysis expert to validate the charts proposed during the experiment tasks, in such a way to avoid to favor none between CG or DG. The comprehension was assessed answering questions based on data related to real databases.

- *Violated assumptions of statistical tests.* Certain tests make assumptions on, for example, normally distributed samples. Violating the assumptions may lead to wrong conclusions. In our procedure we selected the appropriate statistical tests according to the sample normality. Similarly, we applied statistical power when assumptions were not violated.
- *Appropriateness of measures.* Comprehension was measured in terms of an information retrieval based approach. The comprehension was assessed answering questions based on data related to real databases. In addition, the DG and CG graphs proposed in the experiment were validated by a data analysis expert for verifying whether the graphs were appropriate for answering the questionnaire questions. Regarding the variable time, we asked the participants to write down the start and the end times when they filled in a questionnaire section. This information was validated by supervisors. The post-experiment survey questionnaire was designed using standard methods and scales [76].

Conclusion validity

It concerns issues that may affect the ability of drawing correct conclusions.

- *Random heterogeneity of participants.* There is always heterogeneity in a study group. If the group is heterogeneous, there is the risk that the variation due to individual differences may affect the results [107]. We tried to mitigate this threat by distributing the participants in groups by taking into account the average grade (ability).
- *Reliability of measures.* This threat is related to how response variables were measured. The method used to assess effectiveness and efficiency constructs could have affected results. Regarding task completion time, we asked participants to write start and stop times [49]. In our experiments, the selection of the population might affect this experimental validity. To reject the null hypotheses, we used statistical tests and power analyses. The conclusion validity could be also affected by the sample size, consisting in 47 participants. To this aim, the statistical power of the tests was evaluated. Finally, the used Post-Experiment survey questionnaire was designed using standard ways and scales [76].

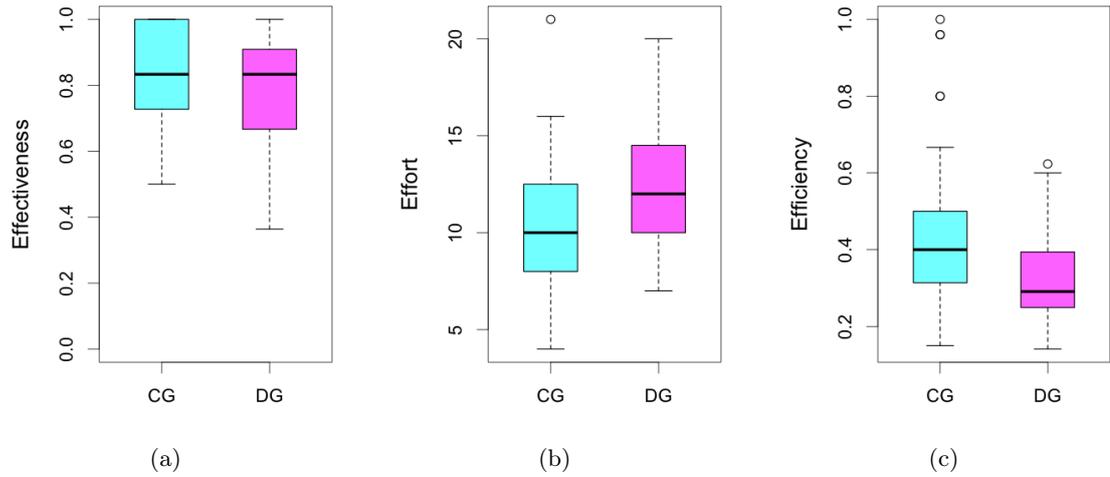


Figure 7.1: BoxPlots of the empirical analysis.

Table 7.5: Descriptive statistics for Dependent Variables.

Dependent variable	Method	Median	Mean	St. Dev.	Min	Max
<i>Effectiveness</i>	CG	0.83	0.84	0.14	0.5	1
	DG	0.83	0.77	0.17	0.36	1
<i>Effort</i>	CG	10	10.38	3.44	4	21
	DG	12	12.3	3.05	7	20
<i>Efficiency</i>	CG	0.4	0.44	0.2	0.15	1
	DG	0.29	0.32	0.12	0.14	0.62

7.2 Results of the Empirical Evaluation

The results of the experiment are discussed by considering the effect of method, analysis of co-factors, post-experiment results, and implications.

Table 7.5 reports descriptive statistics of the dependent variables Effectiveness, Effort and Efficiency grouped by treatment (DG, CG). The results achieved are also summarized as boxplots in Figure 7.1. These statistics show that the average Effectiveness was higher in case of CG (7% more on average), while users took on average 1 min. and 42 (15%) secs. less to comprehend the same problem when using a DG with respect to a CG. Users are on average 12% more efficient with CG with respect to DG.

7.2.1 Hypotheses Testing

Results for the null hypotheses H_{n1} , H_{n2} and H_{n3} are summarized in Table 7.6. For each hypothesis we report p-values, effect size and statistical power (when applicable), grouped by treatment (DG, CG).

H_{n1} : task effectiveness.

We obtained $p - value < 0.004$ by applying Shapiro-Wilk normality test to the F-measure DG and CG samples. Since all p-values are less than 0.05 the samples are not normally distributed we had to adopt a non-parametric test. By applying the Wilcoxon Rank-Sum Test we determined the p-values reported in Table 7.6. The Wilcoxon Rank-Sum Test revealed that there exists a significant difference on comprehension in terms of Effectiveness ($p - value = 0.045$) in case participants used CG and DG (i.e., H_{n1} can be rejected). The effect size is small.

H_{n2} : task effort.

Concerning the Time performance, by applying Shapiro-Wilk normality test we got $p - value = 0.317$ and $p - value = 0.22$ for DG and CG, respectively. Since all p-values were greater than 0.05, both the samples are normally distributed, so we adopted a parametric test (t-test). Table 7.6 summarizes the results of the hypotheses tests for H_{n1} and H_{n2} and reports the values of effect size and statistical power. Concerning H_{n2} , the t-test revealed that the null hypothesis, used to assess the influence of Method on Effort, could be rejected since $p - value = 0.007$. The effect size was medium and negative (i.e., -0.589) and the statistical power was 0.81.

H_{n3} : task efficiency.

By applying Shapiro-Wilk normality test to the Efficiency DG and CG samples we obtained $p - values$ less than 0.05 (the samples are not normally distributed) so we adopted a non-parametric test. The Wilcoxon Rank-Sum Test highlighted that there exists a significant difference on Efficiency ($p - value = 0.001$) in case participants used CG and DG (i.e., H_{n3} can be rejected). The effect size is medium.

7.2.2 Analysis of Co-factors

Effect of Subjects' Ability

The analysis of the interaction plots in Figure 7.2(a) highlights that participants with High Ability achieved better Effectiveness than Low ability ones in both the CG and DG treatments. The analysis also revealed that there is no interaction between Method and Ability: the gap is the same with the two methods, with a performance improvement for CG. Similar results were also obtained when we analyze the influence of the participants' Ability on Effort and on Efficiency (Figure 7.2(b) and Figure 7.2(c), respectively).

Effect of the Session

To investigate whether there is a learning effect when the participants answered successive sessions of the questionnaire, let us analyze the Interaction Plots in Figure 7.3. In particular, it is possible to observe that for Effectiveness, Effort and Efficacy there is a learning effect. Probably, this is due to the task type (a comprehension task) that is similar in the two sessions, except for the task domain and for the graph notation. However, this little learning effect has been mitigated by the balanced experiment design.

7.2.3 Post-Experiment Results

The data collected from the Post-Experiment Questionnaire (Table 7.3) are visually summarized in Figure 7.4. P1 suggests that many participants considered the questions of the comprehension tasks clear (10 expressed a very positive judgement and 29 a positive judgement). The histogram of P2 shows that the participants found the objectives of the experiment clear (11 expressed a very positive judgement and 27 a positive judgement).

Table 7.6: Results of the analyses on method.

Hypotheses	p-value	Effect size	Stat. Power
H_{n1} (Effectiveness)	Yes (0.045)	0.234 (Cliff's delta small)	-
H_{n2} (Effort)	Yes (0.007)	-0.589 (Cohen medium)	0.81
H_{n3} (Efficiency)	Yes (0.001)	0.411 (Cliff's delta medium)	-

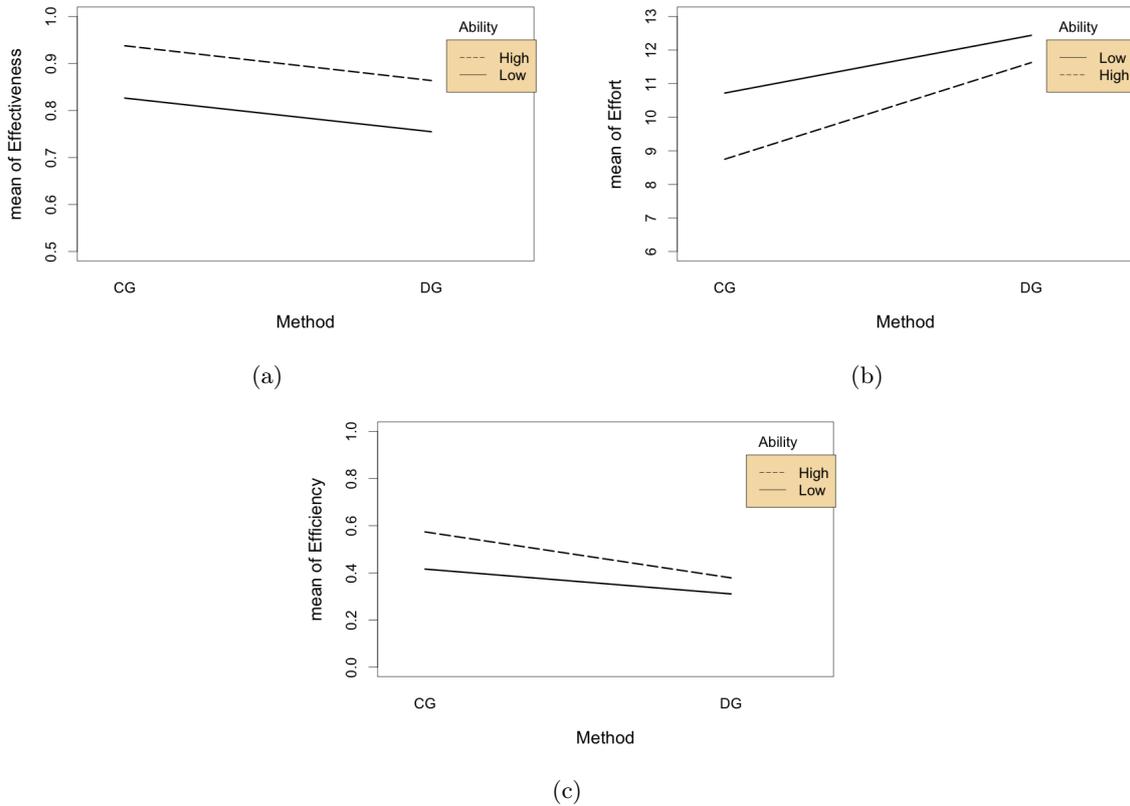


Figure 7.2: Interaction plots of Effectiveness (a), Effort (b), and Efficiency (c) for the Ability level.

The participants found useful the relationships of the CG representation for comprehending data (P3), 14 expressed a very positive judgement and 22 a positive judgement. By looking at the answers to questions P4, many participants did not considered problematic understanding the meaning of the data represented using CG. (14 scored 5 and 22 scored 4). 11 participants considered the proportion among sub-graphs of CG very useful for comprehending data, while 19 useful (P5). Concerning the help provided by the arrangement of sub-graphs of CG for comprehending data (P6), 8 (resp., 17) expressed a very positive (resp., positive) judgment.

7.2.4 Implications

In the following we discuss observed results with respect to the defined research question, and the implication of this study from a qualitative and quantitative point of view.

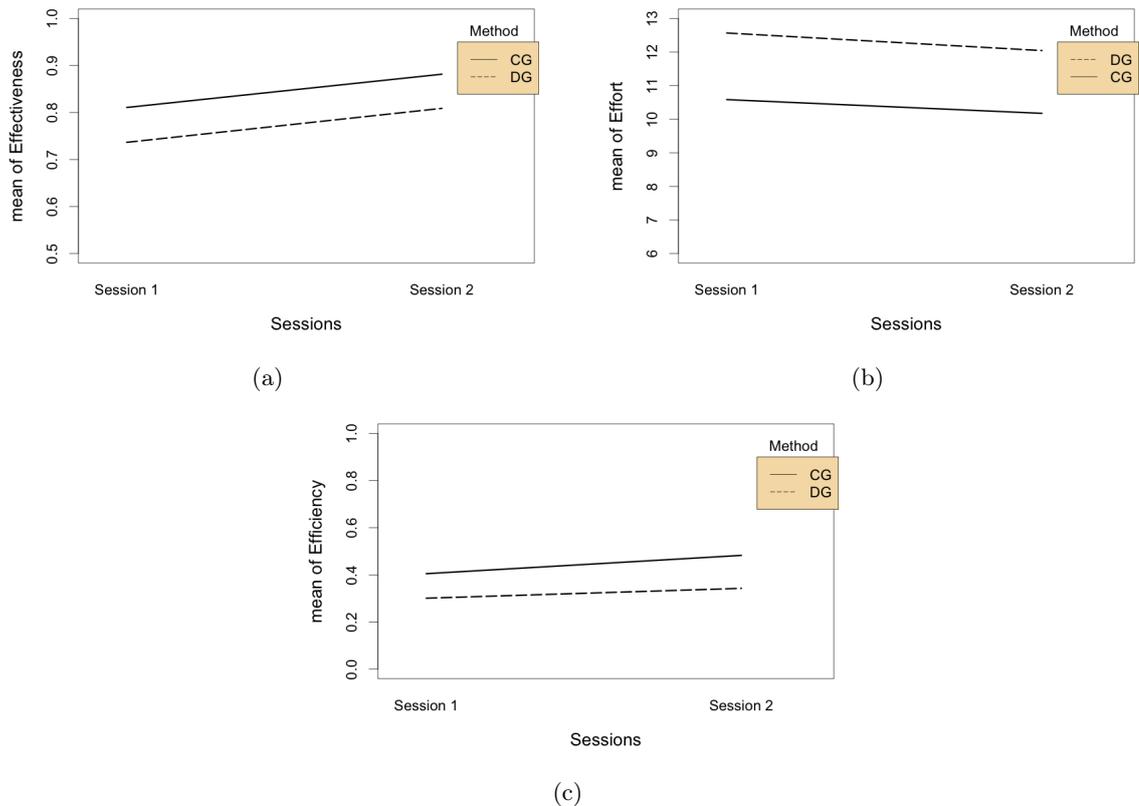


Figure 7.3: Interaction plots of learning effects for Comprehension (a), Effort (b), and Efficiency (c).

As for our research question, (i.e., *To what extent a report generated by the CoDe Graph Generator tool does impact user comprehension?*) we were able to reject H_{n1} , H_{n2} and H_{n3} . In other words, it seems that the comprehension with CG has a positive impact w.r.t. DG. The Effect size clarifies the extent of the improvement: it resulted *small* for Effectiveness and *medium* for both Effort and Efficiency. Thus, it seems that the comprehension has a little improvement in the CG case, but the ability to well understand and without waste time is further improved. The Statistical Power concerning Effort (the only computable) is 0.81 and respects the adequacy standard [39].

Both high and low ability users took advantages from the CoDe Graph representation in term of Effectiveness, Effort and Efficiency. This finding is relevant for both the decision maker and less skilled people: CoDe Graphs represent an effective visualization approach that enables a quicker and higher comprehension of data. Thus, they may improve the

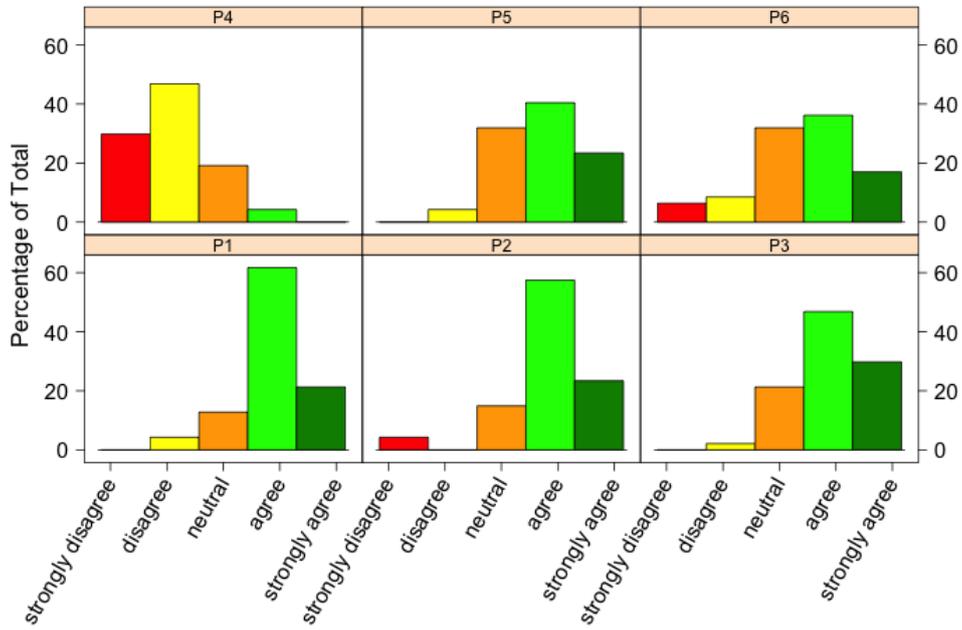


Figure 7.4: PostExperiment Questionnaire results.

report comprehension and may reduce the decision making effort.

User preferences and performances both took advantage of the adoption of CG. Thus, they may be a better mechanism for information visualization and for interpreting data.

Chapter 8

Conclusions and Future Works

In the first part of the thesis, we presented a classifier for petroglyph symbol reliefs robust to distortions and local shifts. The method is based on IDM algorithm [59] to measure the distances between the queries and the reference images, which is an effective means of compensating for small local image. The experimental results show the potential of the proposed method for petroglyph classification with a classification rate of 68%, which considerably improves a previous distance proposed for petroglyph recognition of about 33% [111]. These results are achieved on a representative dataset of Mount Bego petroglyphs, which includes all the main petroglyph classification challenges.

The main research challenges for the future will be the investigation of other optimization strategies for IDM and the validation of the results on a larger dataset. In particular, in order to improve the efficiency of the approach we are going to explore the possibility to reduce the number of IDM comparisons by means of a clustering algorithm. Moreover, we intend to investigate the use of a distance with a behavior orthogonal to IDM, i.e., able to achieve good performances for IDM misclassified images. In this way, by applying a suitable weighting scheme it would be possible to exploit the advantages of both techniques.

By considering the visual analytics, our main contributions are summarized as follows:

- we presented DARK (Discovery of Ancient Rock art Knowledge) a prototype system that supports archaeologists in their investigation activities,
- we introduced novel abstraction techniques for ontology based and interactive visual analysis of large repositories of petroglyphs, and

- we developed a set of visualization techniques which helps to analyze repositories of petroglyphs.

The tool supports archaeologists in the detection of recurrent petroglyph patterns by providing views that highlight the correlations between the spatial relationships of petroglyph symbols and the semantic information extracted from their interpretations. The latter are taken from an ontology used to organize data associated with petroglyphs. The abstraction techniques used for the construction of the views are based on the semantic and structural information of the petroglyphs, while fuzzy theory is used to evaluate the spatial relationships between the symbols in the scenes.

In the future, we plan to exploits visual grammars for modeling the language of identified patterns [23]. We will also investigate the use of query by sketch as a technique to ease user interaction and improve retrieval effectiveness in the repository by extending the agent-based framework presented in [16] with a LDCRF-based sketch recognition algorithm [34]. Finally, we plan to validate DARK with usability studies involving domain experts.

Dashboards are essential tools for monitoring business performance and empowering managers to make better decisions faster. The challenge, then, is to select the best data visualization for a given business. This research started by proposing a tool and a graphical approach for extracting data and representing them in the form of CoDe Graphs. There was the need of deeply understanding whether this approach effectively improves operational efficiency and data comprehension.

In the second part of the thesis, after recalling the main concepts on the CoDe methodology, we introduce the CoDe Graph Generation tool and describe how CoDe Graphs are generated starting from their CoDe model. Then, we assessed the data visualization based on that graphs by conducting a controlled experiment. The aim of this investigation was to determine whether this approach is effective when users analyze real data and on which extent.

Results indicate that CoDe Graphs significantly reduced the time effort of 15% on average and enhanced the comprehension of 7% on average when analyzing graphical data represented by CoDe Graphs with respect to traditional dashboard diagrams. Thus, we can conclude that this kind of graphs can be adopted because it reduces interpretation effort and enhances data comprehension. In addition, the Post-Experiment questionnaire

revealed that participants expressed a positive judgment on CoDe Graphs. The experiment objects were based on data extracted by real data-warehouses and have been validated by a data analysis expert. We considered these experiment objects realistic enough for manager decisions. However, additional investigations should be conducted to further verify that the achieved results scale to executive decisions. This point is of interest for the future work.

References

- [1] S. Afonin, A. Kozitsyn, and I. Astapov. SQLReports yet another Relational Database Reporting System. In *Proceedings of the 9th International Conference on Software Engineering and Applications (ICSOFT-EA)*, pages 529–534. IEEE, 2014.
- [2] Y. Albo, J. Lanir, P. Bak, and S. Rafaeli. Off the radar: Comparative evaluation of radial visualization solutions for composite indicators. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):569–578, 2016.
- [3] J. F. Allen. Maintaining Knowledge About Temporal Intervals. *Communication of ACM*, 26(11):832–843, 1983.
- [4] F. I. Anfurrutia, O. Diaz, and S. Trujillo. A Product-Line Approach to Database Reporting. *IEEE Latin America Transactions*, 4(2):70–76, 2006.
- [5] R. H. Bartels, J. C. Beatty, and B. A. Barsky. *An Introduction to Splines for Use in Computer Graphics and Modeling*. Morgan Kaufmann, 1987.
- [6] V. R. Basili and H. D. Rombach. The TAME Project: Towards Improvement-Oriented Software Environments. *IEEE Transactions on Software Engineering*, 14(6):758–773, 1988.
- [7] S. Bateman, R. L. Mandryk, C. Gutwin, A. Genest, D. McDine, and C. Brooks. Useful Junk?: The Effects of Visual Embellishment on Comprehension and Memorability of Charts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 2573–2582. ACM, 2010.
- [8] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

- [9] J. Bertin. *Semiology of Graphics: Diagrams, Networks, Maps*. UMI Research Press, 1983.
- [10] Jacques Bertin. *Semiology of Graphics*. University of Wisconsin Press, 1983.
- [11] N. Bianchi. Mount bego: prehistoric rock carvings. In *Adoranten*, pages 70–80. Scandinavian Society for Prehistoric Art, 2010.
- [12] C. A. Bicknell. *A guide to the prehistoric rock engravings in the Italian Maritime Alps*. Printed by G. Bessone, 1913.
- [13] L. Candela, G. Athanasopoulos, D. Castelli, K. El Raheb, P. Innocenti, Y. Ioannidis, A. Katifori, A. Nika, G. Vullo, and S. Ross. The Digital Library Reference Model. Technical Report Deliverable D3.2b, DL.org: Coordination Action on Digital Library Interoperability, Best Practices and Modelling Foundations. Funded under the 7th Framework Programme. Project Number 231551., 2011.
- [14] S. K. Card, J. D. Mackinlay, and B. Shneiderman, editors. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.
- [15] J. Carver, L. Jaccheri, S. Morasca, and F. Shull. Issues in Using Students in Empirical Studies in Software Engineering Education. In *Proceedings of the 9th International Symposium on Software Metrics (METRICS)*, pages 239–. IEEE CS Press, 2003.
- [16] G. Casella, V. Deufemia, V. Mascardi, G. Costagliola, and M. Martelli. An agent-based framework for sketched symbol interpretation. *Journal of Visual Languages and Computing*, 19(2):225–257, 2008.
- [17] S Chaudhuri and U. Dayal. An overview of data warehousing and OLAP technology. *SIGMOD Record*, 26:65–74, 1997.
- [18] C. Chippindale and C. A. Bicknell. *Archaeology and Science in the 19th Century*, volume 58. 1984.
- [19] C. Chippindale and P. S. C. Taçon. *The archaeology of rock-art*. Cambridge University Press, 1998.

- [20] M. Ciolkowski, D. Muthig, and J. Rech. Using Academic Courses for Empirical Validation of Software Development Processes. *Proceedings of EUROMICRO Conference*, pages 354–361, 2004.
- [21] D. Constantinidis. Seeing is believing: The visual analysis of archaeological data. In *Proceedings of the 2002 Advanced Information Visualization in Archaeology Workshop*, 2002.
- [22] M. Correll and M. Gleicher. Error bars considered harmful: Exploring alternate encodings for mean and error. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2142–2151, 2014.
- [23] G. Costagliola and G. Polese. Extended positional grammars. In *Proceedings of the International Conference on Visual Languages*, pages 103–110, 2000.
- [24] T. Cover and P. Hart. Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 2006.
- [25] M Danese, U. Demar, N. Masini, and M. Charlton. Investigating material decay of historic buildings using visual analytics with multi-temporal infrared thermographic data. *Archaeometry*, 52(3):482–501, 2010.
- [26] A. De Lucia, C. Gravino, R. Oliveto, and G. Tortora. An Experimental Comparison of ER and UML Class Diagrams for Data Modeling. *Empirical Software Engineering*, 15(5):455–492, 2010.
- [27] H. de Lumley and A. Echassoux. The rock carvings of the chalcolithic and ancient bronze age from the mont bego area. the cosmogonic myths of the early metallurgic settlers in the southern alps. *L Anthropologie*, 113(5):969–1004, 2009.
- [28] J. S. Dean, G. J. Gumerman, J. M. Epstein, R. L. Axtell, A. C. Swedlund, M. T. Parker, and S. McCarroll. *Dynamics in Human and Primate Societies*, chapter Understanding Anasazi Culture Change Through Agent-based Modeling, pages 179–205. Oxford University Press, Oxford, UK, 2000.
- [29] V. Deufemia, V. Indelli Pisano, L. Paolino, and P. de Roberto. A Visual Analytics System for Supporting Rock Art Knowledge Discovery. In *Proceedings of the Inter-*

- national Conference on Computational Science and Its Applications (ICCSA)*, pages 466–480. Springer International Publishing, 2014.
- [30] V. Deufemia, V. Indelli Pisano, L. Paolino, and P. de Roberto. *A Mobile Application for Supporting Archaeologists in the Classification and Recognition of Petroglyphs*. Lecture Notes in Information Systems and Organisation. Springer, 2015.
- [31] V. Deufemia and L. Paolino. Combining unsupervised clustering with a non-linear deformation model for efficient petroglyph recognition. In *Advances in Visual Computing*, volume 8034 of *LNCS*, pages 128–137. Springer, 2013.
- [32] V. Deufemia, L. Paolino, and H. de Lumley. Petroglyph recognition using self-organizing maps and fuzzy visual language parsing. In *Proc. of IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 852–859, 2012.
- [33] V. Deufemia, L. Paolino, G. Tortora, A. Traverso, V. Mascardi, M. Ancona, M. Martelli, N. Bianchi, and H. De Lumley. Investigative analysis across documents and drawings: Visual analytics for archaeologists. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI)*, pages 539–546, New York, NY, USA, 2012. ACM.
- [34] V. Deufemia, M. Risi, and G. Tortora. Sketched symbol recognition using latent-dynamic conditional random fields and distance-based clustering. *Pattern Recognition*, 47(3):1159–1171, 2014.
- [35] P. Dreuw, T. Deselaers, D. Keysers, and H. Ney. Modeling image variability in appearance-based gesture recognition. In *ECCV Workshop on Statistical Methods in Multi-Image and Video Processing*, pages 7–18, Graz, Austria, 2006.
- [36] A. Echassoux, H. de Lumley, J.-C. Pecker, and P. Rocher. Les gravures rupestres des pléiades de la montagne sacrée du bego, tende, alpes-maritimes, france. *Comptes Rendus Palevol*, 8(5):461–469, 2009.
- [37] M. Elias, M.-A. Aufaure, and A. Bezerianos. Storytelling in visual analytics tools for business intelligence. In *Proceedings of the 14th International Conference on Human-Computer Interaction (INTERACT)*, pages 280–297, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

- [38] M. Elias and A. Bezerianos. Annotating bi visualization dashboards: Needs & challenges. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1641–1650, New York, NY, USA, 2012. ACM.
- [39] P. D. Ellis. *The Essential Guide to Effect Sizes: Statistical Power, Meta-Analysis, and the Interpretation of Research Results*. Cambridge University Press, 2010.
- [40] A. Fish, B. Khazaei, and Roast C. User-comprehension of Euler diagrams. *Journal of Visual Languages & Computing*, 22(5):340–354, 2011.
- [41] M. Friendly. *A Brief History of Data Visualization*, pages 15–56. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [42] R. J. Grissom and J. J. Kim. *Effect Sizes For Research: A Broad Practical Approach*. Taylor & Francis Group, 2005.
- [43] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199–220, 1993.
- [44] J. Hannay and M. Jørgensen. The Role of Deliberate Artificial Design Elements in Software Engineering Experiments. *IEEE Transactions on Software Engineering*, 34:242–259, 2008.
- [45] P. Hanrahan. VizQL: A Language for Query, Analysis and Visualization. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 721–721, New York, NY, USA, 2006. ACM.
- [46] V. Harinarayan, A. Rajaraman, and J. D. Ullman. Implementing data cubes efficiently. In *Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 205–216, 1996.
- [47] J. Heer and G. Robertson. Animated Transitions in Statistical Data Graphics. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1240–1247, 2007.
- [48] K. C. Hsu and M. Z. Li. Applying Clustering Analysis on Grouping Similar OLAP Reports. In *Proceedings of the 2nd International Conference on Computer Engineering and Applications (ICCEA)*, volume 2, pages 417–423, 2010.

- [49] L. Huang and M. Holcombe. Empirical Investigation towards the Effectiveness of Test First Programming. *Information & Software Technology*, 51(1):182–194, 2009.
- [50] O. Huisman, I. F. Santiago, M. J. Kraak, and B. Retsios. Developing a geovisual analytics environment for investigating archaeological events: Extending the spacetime cube. *Cartography and Geographic Information Science*, 36(3):225–236, 2009.
- [51] D. K. Iakovidis and E. Papageorgiou. Intuitionistic Fuzzy Cognitive Maps for Medical Decision Making. *IEEE Transactions on Information Technology in Biomedicine*, 15(1):100–107, 2011.
- [52] Z. M. Ibrahim and A. Y. Tawfik. Spatio-temporal reasoning for vague regions. In *Advances in Artificial Intelligence*, volume 3060 of *Lecture Notes in Computer Science*, pages 308–321. Springer Berlin Heidelberg, 2004.
- [53] O. Inbar, N. Tractinsky, and J. Meyer. Minimalism in Information Visualization: Attitudes Towards Maximizing the Data-ink Ratio. In *Proceedings of the 14th European Conference on Cognitive Ergonomics: Invent! Explore! (ECCE)*, pages 185–188. ACM, 2007.
- [54] R. Inkpen, B. Duane, J. Burdett, and T. Yates. Assessing stone degradation using an integrated database and geographical information system (gis). *Environmental Geology*, 56(3-4):789–801, 2008.
- [55] International Organization for Standardization. *Information Technology–Software Product Evaluation: Quality Characteristics and Guidelines for their Use, ISO/IEC IS 9126*. ISO, Geneva, Switzerland, 1991.
- [56] N. R. Jennings, K. P. Sycara, and M. Wooldridge. A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1(1):7–38, 1998.
- [57] N. Juristo and A.M. Moreno. *Basics of Software Engineering Experimentation*. Kluwer Academic Publishers, 2001.
- [58] V. By Kampenes, T. Dybå, J. E. Hannay, and D. I. K. Sjøberg. A Systematic Review of Effect Size in Software Engineering Experiments. *Information and Software Technology*, 49(11-12):1073–1086, 2007.

- [59] D. Keysers, T. Deselaers, C. Gollan, and H. Ney. Deformation models for image recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1422–1435, 2007.
- [60] D. Keysers, C. Gollan, and H. Ney. Local context in non-linear deformation models for handwritten character recognition. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, pages 511–514, Washington, DC, USA, 2004. IEEE Computer Society.
- [61] B. Kitchenham, S. Pfleeger, L. Pickard, P. Jones, D. Hoaglin, K. El Emam, and J. Rosenberg. Preliminary Guidelines for Empirical Research in Software Engineering. *IEEE Transactions on Software Engineering*, 28(8):721–734, 2002.
- [62] T. A. Kohler, J. Kresl, C. van West, E. Carr, and R. H. Wilshusen. *Dynamics in Human and Primate Societies*, chapter Be There then: A Modeling Approach to Settlement Determinants and Spatial Efficiency Among Late Ancestral Pueblo Populations of the Mesa Verde Region, U.S. Southwest, pages 145–178. Oxford University Press, Oxford, UK, 2000.
- [63] K.C. Laudon and J.P. Laudon. *Management Information Systems Activebook*. The Prentice Hall Activebook Experience. Prentice Hall, 2002.
- [64] E. Levy, J. Zacks, B. Tversky, and D. J. Schiano. Gratuitous Graphics? Putting Preferences in Perspective. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 42–49. ACM, 1996.
- [65] S.-K. Lin. Rock art science: The scientific study of palaeoart. second edition. *Arts*, 2(1):1–2, 2013.
- [66] J. Liu, Y. Wu, and G. Yang. Optimization of Data Retrievals in Processing Data Integration Queries. In *Proceedings of the International Conference on Frontier of Computer Science and Technology (FCST)*, pages 183–189, 2009.
- [67] J. Lladós, E. Valveny, G. Sánchez, and Enric Martí. *Symbol Recognition: Current Advances and Perspectives*, pages 104–128. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.

- [68] G. Lock and Z. Stancic. *Archaeology and GIS: A European Perspective*. Taylor and Francis, 1995.
- [69] N. Ma, M. Yuan, Y. Bao, Z. Jin, and H. Zhou. The Design of Meteorological Data Warehouse and Multidimensional Data Report. In *Proceedings of the 2nd International Conference on Information Technology and Computer Science (ITCS)*, pages 280–283, Washington, DC, USA, 2010. IEEE Computer Society.
- [70] N. Ma, Y. Zhai, Y. Bao, and H. Zhou. Design of Meteorological Information Display System Based on Data Warehouse. In *Proceedings of the International Conference on Management and Service Science (ICMSS)*, pages 1–4, 2010.
- [71] S. Mansmann and M. H. Scholl. Exploring OLAP Aggregates with Hierarchical Visualization Techniques. In *Proceedings of the ACM Symposium on Applied Computing (SAC)*, pages 1067–1073, New York, NY, USA, 2007. ACM.
- [72] V. Mascardi, D. Briola, A. Locoro, D. Grignani, V. Deufemia, L. Paolino, N. Bianchi, H. de Lumley, D. Malafrente, and A. Ricciarelli. A holonic multi-agent system for sketch, image and text interpretation in the rock art domain. *International Journal of Innovative Computing, Information and Control (IJICIC)*, 10(1):81–100, 2014.
- [73] V. Mascardi, V. Deufemia, D. Malafrente, A. Ricciarelli, N. Bianchi, and H. de Lumley. Rock art interpretation within indian MAS. In *Agent and Multi-Agent Systems. Technologies and Applications*, volume 7327 of *Lecture Notes in Computer Science*, pages 271–281. Springer, 2012.
- [74] V. Mascardi, V. Deufemia, D. Malafrente, A. Ricciarelli, N. Bianchi, and H. de Lumley. Rock art interpretation within indian MAS. In *Proceedings of the 6th KES International Conference on Agent and Multi-Agent Systems: Technologies and Applications (KES-AMSTA)*, pages 271–281, 2012.
- [75] S. McKenna, D. Staheli, C. Fulcher, and M. Meyer. *BubbleNet: A Cyber Security Dashboard for Visualizing Patterns*, volume 35. 2016.
- [76] A. N. Oppenheim. *Questionnaire Design, Interviewing & Attitude Measurement*. Pinter, 1992.

- [77] E. I. Papageorgiou. Review Study on Fuzzy Cognitive Maps and their Applications during the Last Decade. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 828–835, 2011.
- [78] A. Quispel and A. Maes. Would You Prefer Pie or Cupcakes? Preferences for Data Visualization Designs of Professionals and Laypeople in Graphic Design. *Journal of Visual Languages & Computing*, 25(2):107–116, 2014.
- [79] D. A. Randell, Z. Cui, and A. Cohn. A Spatial Logic Based on Regions and Connection. In *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning*, pages 165–176. Morgan Kaufmann, San Mateo, California, 1992.
- [80] D. Ren, M. Brehmer, Bongshin Lee, T. Höllerer, and E. K. Choe. ChartAccent: Annotation for Data-Driven Storytelling. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 230–239, 2017.
- [81] F. Ricca, M. Di Penta, M. Torchiano, P. Tonella, and M. Ceccato. How Developers’ Experience and Ability Influence Web Application Comprehension Tasks Supported by UML Stereotypes: A Series of Four Experiments. *IEEE Transactions on Software Engineering*, 36(1):96–118, 2010.
- [82] M. Risi, M. I. Sessa, M. Tucci, and G. Tortora. Visualizing Information in Data Warehouses Reports. In *Symposium on Advanced Database Systems (SEBD)*, pages 247–259, 2011.
- [83] M. Risi, M. I. Sessa, M. Tucci, and G. Tortora. CoDe Modeling of Graph Composition for Data Warehouse Report Visualization. *IEEE Transactions on Knowledge and Data Engineering*, 26(3):563–576, 2014.
- [84] N. Salamat and E. Zahzah. Two-dimensional fuzzy spatial relations: A new way of computing and representation. *Advances in Fuzzy Systems*, 2012:1–15, 2012.
- [85] M. Schonlau and E. Peters. Graph Comprehension: An Experiment in Displaying Data as Bar Charts, Pie Charts and Tables With and Without the Gratuitous 3rd Dimension. In *RAND Working Paper Series No. WR- 618*, 2008.

- [86] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1139–1148, 2010.
- [87] M. Seidl and C. Breiteneder. Detection and classification of petroglyphs in gigapixel images – preliminary results. *Proceedings of the 12th International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage (VAST)*, pages 45–48, 2011.
- [88] M. Seidl and C. Breiteneder. Automated petroglyph image segmentation with interactive classifier fusion. In *Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, pages 66:1–66:8, New York, NY, USA, 2012. ACM.
- [89] S. S. Shapiro and M. B. Wilk. An Analysis of Variance Test for Normality. *Biometrika*, 3(52), 1965.
- [90] Y. A. Sher. *Petroglyphs in Central Asia*, 1980.
- [91] Spotfire. <http://www.spotfire.com>.
- [92] M. Springmann, A. Dander, and H. Schuldt. Improving Efficiency and Effectiveness of the Image Distortion Model. *Pattern Recognition Letters*, 29(15):2018–2024, 2008.
- [93] J. Su and H. Su. A Study and Practice of Report System Techniques Based on Three-layer Calculating Architecture. *International Workshop on Education Technology and Computer Science (ETCS)*, 02:654–657, 2010.
- [94] Tableau. <http://www.tableausoftware.com>.
- [95] R. Takaki, J. Toriwaki, S. Mizuno, R. Izuhara, M. Khudjanazarov, and M. Reutova. Shape analysis of petroglyphs in central asia. *Forma*, 21(3):243–258, 2006.
- [96] G. Toffoli. *Basic Notions of JasperReports*, pages 15–23. Apress, Berkeley, CA, 2007.
- [97] K. Tombre, S. Tabbone, and P. Dosch. *Musings on Symbol Recognition*, pages 23–34. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [98] E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, 1983.

- [99] A. Unger, P. Muigg, H. Doleisch, and H. Schumann. Visualizing Statistical Properties of Smoothly Brushed Data Subsets. In *Proceedings of the 12th International Conference Information Visualisation (IV)*, pages 233–239, 2008.
- [100] S. Valtolina, B. R. Barricelli, G. Bagnasco Gianni, and S. Bortolotto. Archmatrix: Knowledge management and visual analytics for archaeologists. In *Human Interface and the Management of Information. Information and Interaction for Learning, Culture, Collaboration and Business*, volume 8018 of *Lecture Notes in Computer Science*, pages 258–266. Springer Berlin Heidelberg, 2013.
- [101] P. van Leusen. *Archaeology and GIS: A European Perspective*, chapter GIS and archaeological resource management: a European agenda, pages 27–41. Taylor and Francis, 1995.
- [102] S. Vegas, C. Apa, and N. Juristo. Crossover Designs in Software Engineering Experiments: Benefits and Perils. *IEEE Transactions on Software Engineering*, 42(2):120–135, 2016.
- [103] J. Verelst. The Influence of the Level of Abstraction on the Evolvability of Conceptual Models of Information Systems. In *Proceedings of the International Symposium on Empirical Software Engineering (ISESE)*, pages 17–26. IEEE CS Press, 2004.
- [104] W. R. Warren. Efficient delivery of cross-linked reports with or without live access to a source data repository, August 27 2013. US Patent 8,521,841.
- [105] D.S. Whitley. *Handbook of rock art research*. G - Reference, Information and Interdisciplinary Subjects Series. AltaMira Press, Beverly Hills, CA, 2001.
- [106] M. Williams. Archaeology and gis: Prehistoric habitat reconstruction. In *Proceedings of ESRI User Conference*, 2004.
- [107] C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, and A. Wesslén. *Experimentation in Software Engineering*. Springer, 2012.
- [108] T. Wojciechowski, B. Sakowicz, D. Makowski, and A. Napieralski. Transaction System with Reporting Capability in a Web-based Data Warehouse Application Developed in Oracle Application Express. In *Proceedings of the 10th International*

- Conference - The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*, pages 273–276, 2009.
- [109] J. Zhang, K. Kang, D. Liu, Y. Yuan, and E. Yanli. Vis4heritage: Visual analytics approach on grotto wall painting degradations. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1982–1991, 2013.
- [110] Q. Zhu, X. Wang, E. Keogh, and S.-H. Lee. Augmenting the generalized hough transform to enable the mining of petroglyphs. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1057–1066, New York, NY, USA, 2009. ACM.
- [111] Q. Zhu, X. Wang, E. Keogh, and S.-H. Lee. An efficient and effective similarity measure to enable data mining of petroglyphs. *Data Mining and Knowledge Discovery*, 23(1):91–127, 2011.