Architecture and Applications
of a Geovisual Analytics Framework

Quan Van Ho

Norrköping 2013
Architecture and Applications of a Geovisual Analytics Framework

Copyright © 2013 Quan Van Ho unless otherwise noted
ho.van.quan@liu.se

Department of Science and Technology, Linköping University
SE-601 74 Norrköping, Sweden

ISSN 0345-7524

This thesis is available online through Linköping University Electronic Press:
www.ep.liu.se

Printed by LiU-Tryck, Linköping, Sweden
Abstract

The large and ever-increasing amounts of multi-dimensional, multivariate, multi-source, spatio-temporal data represent a major challenge for the future. The need to analyse and make decisions based on these data streams, often in time-critical situations, demands integrated, automatic and sophisticated interactive tools that aid the user to manage, process, visualize and interact with large data spaces. The rise of ‘Web 2.0’, which is undisputedly linked with developments such as blogs, wikis and social networking, and the internet usage explosion in the last decade represent another challenge for adapting these tools to the Internet to reach a broader user community. In this context, the research presented in this thesis introduces an effective web-enabled geovisual analytics framework implemented, applied and verified in Adobe Flash ActionScript and HTML5/JavaScript. It has been developed based on the principles behind Visual Analytics and designed to significantly reduce the time and effort needed to develop customized web-enabled applications for geovisual analytics tasks and to bring the benefits of visual analytics to the public.

The framework has been developed based on a component architecture and includes a wide range of visualization techniques enhanced with various interaction techniques and interactive features to support better data exploration and analysis. The importance of multiple coordinated and linked views is emphasized and a number of effective techniques for linking views are introduced.

Research has so far focused more on tools that explore and present data while tools that support capturing and sharing gained insight have not received the same attention. Therefore, this is one of the focuses of the research presented in this thesis. A snapshot technique is introduced, which supports capturing discoveries made during the exploratory data analysis process and can be used for sharing gained knowledge.

The thesis also presents a number of applications developed to verify the usability and the overall performance of the framework for the visualization, exploration and analysis of data in different domains. Four application scenarios are presented introducing (1) the synergies among information visualization methods, geovisualization methods and volume data visualization methods for the exploration and correlation of spatio-temporal ocean data, (2) effective techniques for the visualization, exploration and analysis of self-organizing network data, (3) effective flow visualization techniques applied to the analysis of time-varying spatial interaction data such as migration data, commuting data and trade flow data, and (4) effective techniques for the visualization, exploration and analysis of flood data.
Acknowledgements

My first thanks goes to my supervisors. To Mikael Jern for introducing me to the field of geovisual analytics and for his support and guidance during these five years. To Johan Moe at the Ericsson Research department of Ericsson company for his support during the time I worked with his group in VoSON project.

Second, I would like to thank my colleagues at NCVA and NComVA, especially to Patrik Lundblad and Tobias Aström for their collaboration during the five years, to Markus Johansson for his contribution to the development of the GAV Flash version of the framework and to Miralem Drek for his contribution to the development of the GAV HTML5/JavaScript version of the framework.

Many thanks also go to Matt Cooper for proof-reading some of my papers, to Karljohan Lundin Palmerius, Jimmy Johansson, Anja Johansson and Katerina Vrotsou for their help when I started writing this thesis and providing the LaTeX template on which this thesis is based and to Eva Skrblová for her help during the five years with many conference bookings and travel arrangements.

I would like to also thank my colleagues at MIT and my friends who have supported me during these years.

Finally, my special thanks go to my family, especially my wife Cam Tu and my mother, for their support, encouragement and sharing with me through all the good times and hard times.
Contents

A  Context of the work 1
   1  Introduction ................................................. 3
      1.1  Visual analytics ........................................... 4
      1.2  Geovisual analytics ....................................... 7
      1.3  Challenges ................................................ 8
      1.4  Overview of the included papers .......................... 9

2  Background and Related Work .................................. 11
   2.1  Interaction techniques ..................................... 11
      2.1.1  Fisheye lens technique ................................. 12
      2.1.2  Multiple linked views ................................. 12
   2.2  Toolkits for spatial-temporal multivariate data ............ 14
   2.3  Tools for capturing EDA processes .......................... 15
   2.4  Visual analysis of communication network data .............. 16
   2.5  Visual analysis of spatial interaction data .................. 17

B  Contributions 19
   3  A Web-enabled Geovisual Analytics Framework ................. 21
      3.1  Framework architecture .................................... 21
      3.1.1  Atomic and functional component architecture ........... 23
      3.1.2  Map component ........................................... 24
      3.2  Visualization methods ...................................... 25
      3.3  Interaction techniques ..................................... 26
      3.4  Perceptual methods ........................................ 29
      3.5  Linking components for multiple linked views ............. 29
      3.6  Dataset model ............................................. 31
      3.7  Architecture of visual analytics applications ............. 31
      3.8  Data providers ............................................. 32
      3.9  Snapshot for storytelling and collaboration ............... 32
      3.10  Extendibility ............................................... 34
      3.11  Choices of programming platforms .......................... 35
      3.12  Implementation challenges ................................ 36
      3.13  Conclusions and contributions ............................ 37
4 Geovisual Analytics for Multivariate Spatio-Temporal Data 39
  4.1 Geovisual analytics for time-varying multivariate volume data 40
    4.1.1 Objective 40
    4.1.2 Approaches and results 40
    4.1.3 Conclusions and contributions 44
  4.2 Geovisual analytics for self-organizing network data 47
    4.2.1 Objective 47
    4.2.2 Approaches and results 48
    4.2.3 Conclusions and contributions 52
  4.3 Geovisual analytics for spatial interaction time series data 55
    4.3.1 Objective 55
    4.3.2 Approaches and results 56
    4.3.3 Conclusions and contributions 60
  4.4 Geovisual analytics for flood data 62
    4.4.1 Objective 62
    4.4.2 Approaches and results 62
    4.4.3 Conclusions and contributions 65
5 Conclusions and Future Research 67
  5.1 Conclusions and contributions 67
  5.2 Future research 69
Bibliography 71

C Publications included in the thesis 79
 I A web-enabled visualization toolkit for geovisual analytics 81
 II Geovisual analytics framework integrated with storytelling applied to HTML5 101
 III Exploratory 3D geovisual analytics 109
 IV Geovisual analytics for self-organizing network data 119
 V Implementation of a flow map demonstrator for analyzing commuting and migration flow statistics data 129
 VI Geovisual analytics and storytelling applied to a flood scenario 141
Complete list of publications


Part A

Context of the work
Chapter 1

Introduction

There is no doubt that we are living in the age of data explosion in which large amounts of data are generated and collected every day in various areas from sensing data, simulation data to statistic data. Therefore, there is an ever increasing need for analysing data, gaining insight into data, extracting meaningful information and communicating the analytical results effectively for action. Since our human brain is not good at processing and analysing raw data, especially when facing large amounts of data, we need the means that can support us to do these jobs. Visualization is one of such means. It is a tool or method which constructs mental visual images of data to support gaining insight into the data through creating visual representations of the data. It is based on the fact that our human brain is built around pattern recognition, with more than 50 % of its neurons involved with vision.

Visualization is usually divided into two main fields: scientific visualization and information visualization. The classification is based on the type of data they are concerned with, even though they overlap to a great extent. While scientific visualization is normally associated with physically based data such as physical or medical data and covers accurate visualizations of the real world, information visualization is associated with abstract and non-physically based data and covers visualizations of concepts that are often abstract in nature. In addition to this classification, visualization can also be classified based on the field to which it is applied. For example, when studied and used in cartography, visualization forms a sub-field, geovisualization, which focuses on visualization techniques and tools supporting geospatial data analysis and, to some extent, overlaps with the fields of scientific visualization and information visualization. Geovisualization and information visualization techniques are among the subjects of this thesis.

Although visualization has developed dramatically in recent decades and many visualization techniques have been invented and proven to be useful tools for supporting data analysis, they still do not meet the growing needs for gaining profound insights and answering complex questions. The events of September 11, 2001 shocked America and the whole world and showed that employing the best practices and the best technologies of those days in many areas including visualization could not protect America from the terrorist threats. The events have constituted both needs and challenges for combining technologies from various areas to support the analyst in processing and analysing data coming from various sources and being in various formats, gaining profound insights and communicating analytical results for collaboration and dissemination. Visual analytics (VA) has emerged as a
Introduction

research area to response to these needs and challenges. In 2005, the book Illuminating the Path: The Research and Development Agenda for Visual Analytics [78] was published to address these challenges as well as to define a research and development agenda for visual analytics.

The research presented in this thesis is inspired by the book and responds to the calls for further research on the development of frameworks that “support seamless integration of tools so that data requests, visual analysis, note-taking, presentation composition and dissemination all take place within a cohesive environment” as well as the calls for moving research into practice. It focuses on the architecture of a geovisual analytics framework which supports the integration of exploratory, analytical reasoning and communicative tools. It addresses challenges to develop these tools and to adapt them to the Internet to reach a broader user community. It also presents a number of real world interactive web-based visualization applications developed based on the framework. Some of them are being used daily for analysing data and presenting analytical results to the public through interactive attractive visualizations to capture their attention as well as to help them understand better our society and our world through statistic data.

The thesis is structured as follows. The remainder of this chapter presents the concepts underlying the thesis, the research challenges and an overview of the publications included in the thesis. Chapter 2 describes the background and research related to the thesis. Chapters 3 and 4 summarize a number of selected publications of the author used for this thesis, mainly including the results and contributions of these publications. These publications are included in the printed version of the thesis but not in the online version of the thesis due to copyright restrictions. Chapter 5 provides general conclusions, summarizes the main contributions of this research and discusses potential directions for future research.

1.1 Visual analytics

Visual analytics is an interdisciplinary field which emerged in the last decade. The term ‘visual analytics’ was introduced for the first time by Wong and Thomas [86] in 2004 and then defined and discussed in further detail in the book Illuminating the Path: The Research and Development Agenda for Visual Analytics [78] of Thomas and Cook which was published in 2005 after the shocking events of September 11, 2001 to address challenges for gaining profound insights from large amounts of data which are normally disparate, conflicting and dynamic and come from many sources.

Visual analytics, as defined by Thomas and Cook [78], is “the science of analytical reasoning facilitated by interactive visual interfaces”. According to the authors of the book, analytical reasoning is the key factor in the analysis process, and the science of analytical reasoning aims at providing a reasoning framework upon which analysts can determine which technologies are needed for conducting analysis. These technologies can be data processing technologies, data representation technologies, data mining technologies, data visualization technologies, and so on. Interactive visual interfaces are a part of these technologies, which mainly include visualization techniques and interaction techniques (al-
1.1 Visual analytics

though interaction techniques in many cases are a part of visualization techniques).

Visual analytics, in the context of the book, emphasizes the importance of the analyst who applies reasoning techniques and human judgments to gain insight into data and reach conclusions from a combination of evidence and assumptions. Different from the definition mentioned above, Keim et al. [43] gave another definition of visual analytics which emphasizes the role of automated analysis techniques: “Visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex data sets.”

The goal of visual analytics, as stated by Thomas and Cook, is to facilitate analytical reasoning processes through the creation of tools which maximize human capacity to perceive, understand and reason about complex and dynamic data and situations. Keim et al. [43] stated the goal of visual analytics in a more specific fashion.

“The goal of visual analytics is the creation of tools and techniques to enable people to:

- Synthesize information and derive insight from massive, dynamic, ambiguous, and often conflicting data.

- Detect the expected and discover the unexpected.

- Provide timely, defensible, and understandable assessments.

- Communicate assessment effectively for action.”

A visual analysis process normally includes the following four phases: (1) processing and representing data, (2) visualising data, (3) analysing data and (4) presenting and communicating analytical results; therefore, visual analytics, as suggested by Thomas and Cook, includes the following four focus areas:

- data representations and transformations that convert data in ways that support visualization and analysis;

- visual representations and interaction techniques which allow the user to see, explore and understand large amounts of data at once;

- analytical reasoning techniques that enable users to gain deep insights that directly support assessment, planning and decision making;

- techniques to support production, presentation and dissemination of the results of the analysis process to communicate information in the appropriate context to various types of audiences.

Data representations include techniques representing data that can be in many forms such as text, images, audios and videos. Since data representations are normally the inputs of visualizations, the quality of data representations affects the quality of visualizations. A typical data representation used in this thesis for statistical data, which is multivariate spatio-temporal, is a three-dimensional array structure, as illustrated in Figure 3.10. Data transformations are computational procedures which are used to transform data for various purposes. For example, they can be used
Introduction

- to transform data into representations that are appropriate to the analytical tasks;
- to merge data from various sources into a single source to facilitate analysis;
- to augment data by deriving or computing additional data; and
- to convert data into new, semantically meaningful forms.

Visual representations and interaction techniques are normally considered as two components of interactive visualizations; however, in a general context, they can also include graphical user interfaces (GUIs). Visualization techniques can be any techniques from the fields of information visualization, scientific visualization and geovisualization if they can support the analyst in performing analysis. Visual representations and interaction techniques are important components of a visual analytics system. The analyst normally looks at data through visual representations of data and interacts with data through interaction techniques. Therefore, they play a key role in supporting the analyst in understanding the data and making effective decisions.

Analytical reasoning techniques are based on scientific foundations such as mathematics and statistics. These techniques are normally applied to finding patterns and trends or detecting outliers and abnormal phenomena. A number of analytical reasoning techniques can be implemented as automated algorithms, such as algorithms in the fields of data mining and information retrieval for finding trends and extracting meaningful information from large amounts of data.

An analytical reasoning process is a process of applying analytical reasoning techniques to gain insight into data and make decisions. Therefore, together with visual representations and interaction techniques, analytical reasoning techniques play a key role in the data analysis process.

Analysis in many cases requires collaborative effort, especially in emergency response or complex situations. Techniques for supporting collaboration among analysts are therefore necessary for increasing the performance of the analysis process.

Production, presentation and dissemination of the analysis results are one of most important phases of the analysis process because it brings and communicates the results to the analysis consumers such as planners, policy makers and the public. This phase is also one of most time-consuming phases. The goal of visual analytics in this focus area is therefore to reduce the time for capturing and sharing the analysis results as well as to improve the effectiveness of the communication of the results. Capturing and sharing the analysis results can also be used to encourage and enhance the collaboration among analysts or to get feedback from other users to improve the analysis results. Therefore, they need to be taken into account in the development of visual analytics systems.

Visual analytics versus information visualization

Although visual analytics and information visualization overlap to a great extent, the main difference between them is that while information visualization focuses more on developing visualization techniques, visual analytics focuses more on developing data analysis
1.2 Geovisual analytics

Geovisual analytics is an emerging interdisciplinary field which is based on the principles of visual analytics. It can be considered as a sub-field of visual analytics but focuses more
on geographic aspects such as geospatial data, map techniques and geographic phenomena. For example, in 2007, Andrienko et al. [6] suggested one research focus area for geovisual analytics which focuses on spatial decision support and is named ‘Geovisual Analytics for Spatial Decision Support’. This focus area deals with spatial decision problems which normally (1) have complex nature of geographic and temporal spaces, (2) involve multiple actors with different roles and (3) involve tacit criteria and knowledge. It aims at supporting users to make effective decisions through the use of automated algorithms, interactive visualizations and human judgements. Another key topic of geovisual analytics is ‘Space and Time’ [5] which focuses on both space and time aspects and addresses the temporal nature of geographic data, the characteristics of time as well as the structure of time.

Similar to visual analytics, geovisual analytics is built on the basis of research in many different areas such as information visualization, geovisualization, geographic information science and data mining. It also takes into account the concepts from a number of other fields such as cognition and perception.

Geovisual analytics can be used for many activities ranging from counter-terrorism, disaster management to strategic business decision making.

Figure 1.1 illustrates fields and techniques integrated into geovisual analytics in this thesis.

1.3 Challenges

Although many advances have been made in the research of visual analytics and geovisual analytics, many challenges still remain. The first challenge is that the analyst wants to have more interaction techniques and interactive features in visualizations. For example, when working with scatter plots, the analyst normally asks for more interaction techniques and interactive features such as zoom, animation, colour changing, attribute switching, bubble scaling, bubble tracing, bubble labelling, and so on. In general, it is still a challenge to combine a rich set of interaction techniques and interactive features into visualizations in a good way to support better analysis.

The second challenge is that during the exploratory data analysis (EDA) process the analyst normally wants to capture discoveries made and save them so that he or she can share them with his or her colleagues or present them to other users such as analysis consumers or the public. This feature requires effective methods for capturing and saving interactive visualizations so that other users can see what the analyst has discovered and even interact with saved interactive visualizations to confirm the result or give feedback. In current practice, most of visual analytics systems miss this feature.

The third challenge is to apply visual analytics to solving problems in different domains. There is still a big gap between the increasing needs of the user for analysing data in many different domains and what visual analytics has contributed so far. The challenge comes from the fact that data in different domains normally have different characteristics and, in many cases, are very complex. For each specific type of data, it may need a specific solution with a specific set of visual analytics techniques. The needs of the user may also
1.4 Overview of the included papers

This section presents a short overview of the publications included in the thesis. A more thorough description of their contributions are presented in chapters 3 and 4. The author of this thesis is the first author of all the papers and the main contributor to papers III, IV, V and VI and one of the main contributors to papers I and II. The main focus of the included publications is as follows:

Paper I presents the architecture, modules and features of the GAV Flash version of the framework which is named GAV Flash framework. It also presents the challenges for the implementation of the framework in Adobe Flash Action Script. The author of this thesis (1) is responsible for the development of the whole framework, (2) together with the

vary a lot from domain to domain. This challenge relates also to the fact that the research so far has not focused on the development of frameworks and systems which can reduce the time and effort for developing visual analytics applications. While visual analytics tools and techniques are many, it is still a challenge to integrate them into frameworks which can facilitate the development of visual analytics applications for different problems in different domains.

Another challenge is to adapt visual analytics tools to the internet environment and mobile platforms so that they can reach a broader user community and the public can benefit from them. Together with the rise of ‘Web 2.0’ and the dramatic development of social networks, blogs, wikis and mobile platforms, the need for using visual analytics tools to disseminate data and analytical results in the form of interactive visualizations to get more attention from the public and the media on various issues is also increasing continuously. However, the current programming platforms for developing web-based visualizations that can run on different web platforms and mobile platforms are still limited in many aspects and still difficult to use.

In summary, the major challenges addressed in this thesis include:

• enhancing visualizations with a rich set of interaction techniques and interactive features to enhance their power in exploring and analysing data;

• effective methods for capturing and saving discoveries made during the EDA process;

• applying visual analytics techniques and principles to solving problems in different domains;

• the development of frameworks and systems which integrate various visual analytics tools and techniques to facilitate the development of visual analytics applications; and

• the adaptation of visual analytics tools and systems to the internet environment to reach a broader user community;
third author are two main contributors to the design, architecture, modules, features and
the implementation of the framework except the storytelling technique and write the sec-
tion GAV Flash framework, (3) together with the second author are two main contributors
to the snapshot technique and write the subsection Snapshot of the section Storytelling
and (4) is the main contributor to the section Related work and the subsection ANROSS
of the section Case studies.

Paper II introduces the GAV HTML5/JavaScript version of the framework, an adap-
tation and extension of the GAV Flash version to HTML5/JavaScript to support tablets
and smartphones, and named GAV HTML5/JavaScript framework. It also presents the
challenges for the implementation of the framework in JavaScript. The author of this thesis
(1) is responsible for the development of the GAV HTML5/JavaScript framework, (2) is
one of two main contributors to the adaptation and extension of the GAV Flash frame-
work to HTML5/JavaScript, (3) is one of two main contributors to the implementation
work, (4) is the main contributor to sections 2 and 3, and (5) partly contributes to section 4.

Paper III presents an approach to and a prototype for the exploration of time-varying
multivariate volume data based on the synergy among a number of techniques from fields
of scientific visualization, information visualization and geovisualization. The author of
this thesis is the main contributor to the paper and the implementation work.

Paper IV presents an approach to and a prototype for the visual analysis of self-organizing
network data to demonstrate how an automated algorithm configures a cellular radio net-
work. The author of this thesis is the main contributor to the paper and the implementa-
tion work.

Paper V presents an approach to and a demonstrator for the visual analysis of spa-
tial interaction data. The author of this thesis is the main contributor to the paper and
together with the second author are two main contributors to the implementation work.

Paper VI presents an approach to and a demonstrator for the visual analysis of flood
data in a flood scenario. The author of this thesis is the main contributor to the paper
and the implementation work.
Chapter 2

Background and Related Work

This chapter will give the background of the research work presented in this thesis and an overview of research works which relate to this research work and are published before the papers included in this thesis are published. It mainly addresses the following four subjects: (1) interaction techniques, (2) geovisual analytics frameworks and toolkits, (3) tools for capturing gained insight and (4) geovisual analytics approaches applied to a number of specific domains.

2.1 Interaction techniques

Insight is normally gained through interaction. Without interaction, visualizations become static images and therefore less useful. Although static images can still be used for analysing data, they do not allow the user to interact with data and therefore limit the ability to support the user in performing data analysis.

Although interaction (including interaction techniques and interactive features) is an important component of visualizations, many research papers have focused more on visual representations than interaction aspects. This may originate from the fact that the implementation of interaction techniques and interactive features normally takes much more time than the implementation of visual representations.

Research on interaction can be divided into two parts. In one part, it focuses on categorizing interaction techniques [22, 44, 68, 88]. For example, Shneiderman [68] classifies interaction techniques into: (1) overview, (2) zoom, (3) filter, (4) details-on-demand, (5) relate, (6) history and (7) extract. Dix and Ellis [22] divide them into (1) highlighting and focus, (2) accessing extra information – drill down and hyperlinks, (3) overview and context zooming and fisheye views, (4) same representation, changing parameters, (5) same data, changing representation and (6) linking representation – temporal fusion. In another part, it focuses on finding new interaction techniques or applying interaction techniques to visualizations [50, 60, 65, 71]. This is also one of the focuses of this research work. The following subsections summarize a number of interaction techniques and highlight the outstanding features of the interaction techniques designed in this research work in comparison with other research works.
2.1.1 Fisheye lens technique

The concept of fisheye was introduced for the first time by Robert W. Wood in 1906 and then applied widely to many areas including visualization areas. For example, Sarkar and Brown [65] applied it to large graphs and maps. Nevertheless, they did not illustrate how the user can use this technique interactively. In a similar manner, Tominski et al. [81] applied it to tree views and graph visualizations. The distortion caused by the fisheye lens technique, in many cases, makes the focus view difficult for normal users to perceive. To make it easier for normal users, this research work does not use the same principle of the fisheye technique; instead, it modifies this technique by removing the distortion effect (as briefly described below) and applies it to bar charts and time graphs, as illustrated in Figure 3.7 and Figure 2.1 (the bottom view). For the bar chart, the bars in the focus view have the same width and this width is bigger than the width of the bars in the context view and can be controlled by the user. This means the user can increase or decrease the width of the bars in the focus view. Similarly, for the time graph, the distances between two consecutive time points in the focus view are the same and these distances are ten times larger than the distances between two consecutive time points in the context view. The focus view in these visualizations can be changed interactively through a dual slider in the bottom of the view. The user can move the slider to move the focus view or move two control points of the slider to extend or shorten the focus view (i.e. to increase or decrease the number of items displayed in the focus view).

2.1.2 Multiple linked views

The multiple linked views subject has been addressed in many research papers [7, 44, 62]. Baldonado et al. [7] introduced guidelines for using multiple views in information visualization. Keim [44] and Roberts [62, 63] emphasized the advantages of multiple linked views. Roberts [62, 63] also provided an excellent overview of using multiple linked views (or coordinated and multiple views). The common mechanisms for linking views include (1) linking based on selection or brushing, (2) linking based on sharing the same data and (3) linking based on filtering. For example, Carr et al. [15] and Becker and Cleveland [9] introduced linking based on selection and brushing in their scatter plots. When items are brushed in one scatter plot, they will be highlighted in other scatter plots. Dykes [23] introduced CDV in which a scatter plot and a choropleth map are linked to each other through a selection mechanism. Shneiderman [67] and Livny et al. [47] introduced another way of linking views: linking based on sharing the same data among views. This data is queried from a database through dynamic queries which can be updated through sliders controlling the parameters of the queries. The work of Jern and his group [36, 37, 38] is an excellent example of how a number of different linking mechanisms can be integrated into one system. The linking mechanisms supported in their framework include:

- linking based on selection; when one or many items are selected in one views, they are highlighted in the other linked views;
2.1 Interaction techniques

Figure 2.1: An illustration of two focus-and-context techniques applied to the time graph: (1) the top time graph uses two sub-graphs, one for the focus view and another for the context view, and (2) the bottom time graph uses a technique which is similar to the fisheye lens technique.

- linking based on sharing the same data model; when the data input of views is updated through dynamic queries, all the views are updated to reflect this change;
- linking based on filtering; this mechanism is done through axes of a parallel coordinate plot which act as sliders; items (or records) are filtered based on their attribute values;
- linking based on sharing the same colour scheme; this mechanism is done through a colour map which contains a number of settings and controls to allow the user to change the colour map; when the colour map or the colour scheme is changed, all linked views are updated based on this change;
- linking based on animation (i.e. by sharing the same point in time); when the time period is changed, all linked views are updated to reflect this change;

The research work presented in this thesis inherits, refines and extends the work of Jern and his group. It develops a model to facilitate linking visualization components to each other. Each linking mechanism is implemented through a linking component (or a linking manager). Views sharing the same linking component are linked to each other. In addition, this research work also introduces one more linking mechanism: linking based on colour filtering, as illustrated in Figure 3.9. This filtering mechanism is similar to the filtering mechanism which is based on attribute values as mentioned above. The difference
Background and Related Work

is that it is done through a colour map containing controls which allow the user to filter items based on their colours. It also extends the linking mechanism based on selection to support the following feature: when one or more items are selected in one view, in addition to being highlighted, these items (or the last selected items if the positions of the items are far to each other in views) are brought to the centre of linked views. This allows the user to quickly see the selected items (or at least the last selected items) in all linked views.

2.2 Toolkits for spatial-temporal multivariate data

Visual exploration of spatial-temporal and multivariate data has been the subject of many research papers. Andrienko and Andrienko illustrated several motivating approaches in their earlier papers [3, 4] for the visual exploration of spatial temporal data. Muller and Schumann [52] provided a good overview of visualization methods for time dependent data. Many research papers and efforts [12, 13, 26, 27, 31, 36, 40, 57, 73, 74, 75, 76, 80, 83] focused on the development of visualization toolkits and systems for different types of data and different purposes. This section provides an overview of frameworks, toolkits and systems which highly relate to the research work presented in this thesis.

Jazz [12] is an extensible zoomable user interface (ZUI) graphics toolkit developed in Java. It is developed based on ‘minilithic’ design philosophy and scene graph techniques. It is designed to particularly support the development of ZUI applications. It supports fisheye views and multiple views; however, it is unclear whether or not it supports linked views. It is also not designed to provide highly functional visualizations (e.g. choropleth map, scatter plot, PCP) to allow developers to quickly build applications. Instead, it provides a set of nodes based on which developers build visualizations by themselves.

Polaris [73] is a system developed for query, analysis and visualization of multi-dimensional relational databases. It is designed to visualize tabular data structures queried from a relational database. It develops a grammar of graphics, which is similar to the work of Wilkinson [84], to build up visualizations. A number of visualizations introduced in the system include a graphic table, Gantt charts, a map, a line chart, a bar chart and a scatter plot. It supports multiple linked views through the mechanisms based on sharing the same data, filtering or brushing.

GeoVISTA studio [74, 76], VIS-STAMP [31], Infovis toolkit [26] and GAV C# framework [36, 40] are the toolkits and frameworks which are close to the framework developed in this research work. The GeoVISTA studio is developed based on Java and JavaBeans, the Infovis Toolkit is based on Java, and the GAV C# framework is based on C#/.NET. All of them have component architecture and support multiple linked views. Linking mechanisms supported in these systems include linking based on selection/brushing, filtering and dynamic queries. The GAV C# framework also supports linking views through colouring and animation mechanisms. The GeoVISTA studio implements a visual programming interface to facilitate building applications. The Infovis toolkit implements a fisheye lens technique in all visualizations it supports. To the author’s best knowledge, except the GAV C# framework which mentions a mechanism (named snapshot) for capturing EDA
2.3 Tools for capturing EDA processes

Processes, none of toolkits or systems mentioned above support capturing EDA processes, which is one of necessary and important features for visual analytics systems and also one of the subjects of this research work.

Although there are many tools for spatial-temporal multivariate data, it has still been a challenge to adapt these tools to the Internet and reach a broader user community, which is one of the goals of this research work. Web-enabled tools are therefore needed for applications which are explicitly designed with the purpose of visualizing, exploring and communicating spatial-temporal and multivariate data through web environments. Such tools should also employ data transformers and data providers, layout mechanisms, interaction, time animation and storytelling suited to geovisual analytics tasks. Protovis [13, 57] and Flare [27] are examples of web-enabled tools. They include a collection of interactive visualization components. Nevertheless, they do not seem to support multiple linked views, time-animation and abilities to capture EDA processes.

Tableau [75] is another example of a popular web-enabled tool applied to business analytics. In addition to a collection of interactive visualizations, it also supports multiple linked views.

To the author’s best knowledge, most of web-enabled visualization toolkits (in the context before the GAV Flash framework developed in this research work was published) either do not support a map view or support a map view with very limited interactive features. They do not seem (1) to support working with online maps such as Google map, Bing map and OpenStreetMap, and (2) to allow putting visualizations, for example glyph layers, on top of the online maps. These features are among the objectives of this research work.

2.3 Tools for capturing EDA processes

Capturing EDA processes is important for sharing, communicating and evaluating analytical results, and collaboration. Many research papers [35, 69, 78] addressed this need. One common and simple approach is to capture the screen or views as static images. Another approach is to capture the state of views. Jern described this idea in his early paper [35] in which discoveries made during an EDA process can be captured into snapshots which copy the state of components at the time each snapshot was taken. These discoveries can be published and presented to the user through interactive visualizations which allow the user to interact with. Robinson [64] presented a method called “Re-Visualization” and a related tool, named ReVise, that allows users to save and re-use what they have discovered. Shrinivasan and Wijk [69] introduced a framework which can automatically capture the state of visualizations during navigation processes and allows the user to revisit the visualization states sequentially in the order of creation using back and forward arrow keys. Keel [42] presented a visual analytics system of computational agents which supports the exchange of task-relevant information and incremental discoveries of relationships and knowledge among team members commonly referred to as sense-making. However, it is not clear what method or approach is used to capture discoveries.
The research work presented in this thesis uses the same approach as the one used by Jern [35, 39] and the others [64, 69]. In addition to capturing the state of visualizations (i.e. property values of visualizations), it also captures the state of the application, such as the application layout that includes (1) the relative size of views (or the ratios among views) in the application and (2) which views are being active if views are tabbed. This allows the reader to have the same view of the application as the author who shares the discoveries made during an EDA process.

2.4 Visual analysis of communication network data

As communication networks become more complex, it becomes more important for the network operator to understand the behaviour of communication networks. Communication network data visualization has therefore been the topic of many research papers. Withall et al. [85] provided an excellent overview of the research in this domain. Many research efforts have focused on the visualization of the topology of networks, which is not the goal of this research work. Many others have focused on the visualization of the geographical structure of networks. For example, Becker et al. [8, 10, 11] presented SeeNet, a system that uses three graphical tools, ‘Link map’, ‘Node map’ and ‘Matrix display’ to visualize network data, analyse network traffic and study overloads. ‘Link map’ uses line segments and arcs to represent pairs of connected nodes in two dimensions. ‘Node map’ uses simple glyphs such as circles and rectangles to represent node locations and node data. Melissarigos and Pu [51] presented a variation of the ‘Link map’ in which line segments connecting between two nodes are extended to represent load-per-link. Cox et al. [17, 18] extended the work of Becker et al. with three-dimensional arcs and bars to present Internet traffic information on a globe representation of the earth. They also used three-dimensional arcs with a two-dimensional map of the earth to partly reduce the problem of clutter.

Koutsofios et al. [45, 46] introduced SWIFT-3D that includes multiple linked views of interactive 3D maps, statistical displays, network topology diagrams and pixel-oriented displays for interactive exploration of large-scale network data. The research work presented in this thesis shares the same approach of using multiple linked views but uses different visualization techniques and analysis approaches to suit well to its goal and its data input which are different from the goal and the data input of other research works. More specifically, in addition to node data and node-node relation data, the self-organizing network data used in this research work also contains a specific type of data: network events that describe how an automated algorithm operates the network. Therefore, one of the goals of this research work is to visualize network events to demonstrate the way an automated algorithm works.
2.5 Visual analysis of spatial interaction data

Spatial interaction data such as migration and commuting data has been the subject of many research papers and efforts [30, 49, 55, 59, 79, 87]. Many of them have focused more on the visualization aspect than the analysis aspect. A common approach to the visualization of spatial interactions (or flows) is to use straight lines, arcs or curves to represent flows and the line thicknesses and/or line colours for the volume of the flows. This approach is similar to one of the approaches mentioned above for network data, since both types of data (network data and spatial interaction data) share the same feature: connections or relations among nodes (or among geographical locations in the case of spatial interaction data). When applied to large geographical datasets, flows tend to be cluttered. To address this problem, there are a number of different approaches. One approach is to group lines into bundles [19, 33, 34, 55]. This approach can effectively reduce the cluttering in a flow map; however, it does not seem to easily support tasks such as comparison of flows back and forth between two locations or to support flow filtering; supporting such tasks are among the objectives of this research work. Every time the user wants to focus on a subset of flows, e.g. the top ten biggest flows or only the flows originating from an origin, and filter out all other flows, then the bundles should be redrawn or flow lines should be clustered again. In the latter case, the curves representing the flows can be changed. This can make it more difficult for the user to observe the flows.

The second approach is to use a grid to compute the line density of each cell based on the total number of migrants moving along flow lines through each cell [59]. This approach is suitable for giving an overall pattern of interactions between places; however, as this approach results in an image, it does not seem to support operations such as filtering, selection, brushing, zooming and panning. Supporting such operations is also among the objectives of this research work.

The third approach is to divide a map into a regular grid in which each cell is used as a small replica of the whole map [87]. This means each cell has a smaller regular grid (each small cell in a smaller grid is called a square) and a smaller map. Each flow is mapped to a square of the cell from which the flow originates. The coordinates (i.e. row and column positions) of this square in the small grid are the same coordinates of the destination cell of the flow in the large grid. Flows mapped to the same square are aggregated. This approach is very promising as it avoids the overlapping among flows while still preserves the spatial layout of all origin and destination locations. However, the overlapping among squares and small maps can make it difficult for the user to determine the destination of flows. Normal users may also find it difficult to understand this technique. It is also hard to compare flows back and forth between two locations.

The fourth approach is to use interactive queries to select and show only a small subset of data at a time [59]. This approach is more suitable for the objectives of this research work. Therefore, it is adopted in this research work.

Although there have been a lot of research efforts in this area, most of them do not seem to meet the objectives of this research work which focuses more on the aspects of usability, user interactions, animation, analysis and knowledge communication instead of
Background and Related Work

giving an overview of spatial interactions.

First, most of them, except the work of Guo [30], do not support user interaction operations such as filtering, selection, brushing, zooming, panning and linking which are very important to facilitate the EDA process.

Second, none of them use an additional dataset containing location attributes (e.g. income level, employment rate, purchase power) in combination with the flow dataset to support the analysis of the flow dataset, for example, answering questions such as ‘Why do people move from one location to another?’

Third, they do not seem to support the visualization of flow data time series, which is an important aspect in most migration and commuting scenarios.

Fourth, they do not support web-enabled visualization which allows public users to access their work easily through internet. Public users, therefore, hardly benefit from their work.

Fifth, they do not support capturing analytical results. Therefore, the user cannot easily communicate or share their discoveries with the others, which is one of important and needed features for geovisual analytics systems.
Part B

Contributions
Chapter 3

A Web-enabled Geovisual Analytics Framework

The research presented in this thesis is based on a recommendation from the visual analytics (VA) research program, “support seamless integration of tools so that data requests, visual analysis, note-taking, presentation composition and dissemination all take place within a cohesive environment” [78, page 18], which addresses the need for integrated exploratory, analytical reasoning and communicative tools. It is concerned with the development of geovisual analytics techniques and the integration of these techniques into a framework to facilitate the development of applications for geovisual analytics tasks. The research is targeted at the web environment, which is growing dramatically, to meet the needs for the exploration and analysis of data and the presentation and dissemination of gained insight through web and internet to reach a broader user community. This chapter provides an overview of the architecture, modules and features of a web-enabled geovisual analytics (GAV) framework developed within this research. The first version of the framework is implemented in Adobe Flash ActionScript, named GAV Flash, and is introduced in paper I. The second version of the framework is implemented in HTML5 and JavaScript, named GAV HTML5/JavaScript, and is introduced in paper II. Both versions share the same design principle, architecture and most of the features.

3.1 Framework architecture

The web-enabled framework in this thesis is developed based on component thinking or the component based approach which is inherited from its predecessor (GAV C# framework [36, 37, 38]). Based on this approach, the framework is built as a collection of components which allows application developers to pick and choose from a wide range of visualizations, data providers and data transforms and combine them in various ways they want. This facilitates the development of large-scale systems. In many cases the process of building an application based on this framework becomes the process of rigging components together.

Components are designed so that they can easily communicate and be combined with other components and can also be easily extended. To achieve this goal, the interface programming mechanism is exploited. Each component is defined through an interface which details only necessary functions and properties for communicating with other components.
An example of this is the dataset, whose interface is limited to functions that supply data and metadata, but all other functions are encapsulated in its implementation. As the components are only aware of the interfaces, we can easily replace the dataset with some other structure, for example, a direct database connection, without re-implementing any visualizations or data processors.

Figure 3.1: An overview of the web-enabled GAV framework architecture

Within this research, the framework is developed to include the following five component types:

- layout components which are for creating application layouts;
- data components which are for loading data, processing data and transforming data;
- visualization components which are for visualization of data and interaction with data;
3.1 Framework architecture

- linking components which are for linking visualization components together to create multiple-linked views applications; and

- snapshot and storytelling components which are for capturing and saving the state of components and creating stories as well as publishing stories.

Figure 3.1 shows an overview of the framework architecture in which data components are in the bottom, visualization components are in the middle and linking components are in the left. On top of these components is the application level.

3.1.1 Atomic and functional component architecture

Visualization components are usually dynamic and complex components. To facilitate the development of these components, the multilayered architecture is adopted. Each component is broken down into small blocks called atomic components or layers. These atomic components are used together to form a fully functional component; however, they are not dependent on each other. Therefore, they can be combined in any way. This concept can take many forms depending on the parent component. One of the clearest examples is the map component which uses many different layers to display different levels of data as presented in subsection 3.1.2.

Figure 3.2: The multilayered architecture of a functional parallel coordinates plot component
In addition to visual layers, each functional visualization component is designed to have a setting panel which includes many controls and settings to allow the user to control the visual appearance of the component or the way to interact with the component. Figure 3.2 shows an example of a functional visualization component, a parallel coordinates plot (PCP), which is composed of a number of visual layers and a setting panel. The setting panel includes settings which allow the user to change the line opacity and show or hide a number of layers of the PCP.

The next section will describe an example of a functional component to demonstrate the concept of building a functional component from layers.

### 3.1.2 Map component

The heart of the map component is a map base class which contains no visual parts at all but controls everything needed to display something within its context. It creates transformations to deal with projections and keeps track of the user’s input as he or she zooms or pans. The visual parts are created in layers, for example, a polygon layer for coloured regions, a highlighted border layer for selected regions and a glyph layer for region data. By making the map independent of its layers, we can combine them in many different ways or create new layers as long as they adhere to the basic principles of the map. For example, to make it easier for the user to work with geographic regions, a map layer which allows the user to use online web map services such as Google map, Bing map and OpenStreetMap is developed and added into the map component as a background map layer. Based on this approach, map layers can be easily added into or removed from the map component.

In addition to visual layers, a setting panel is also designed to allow the user to show or hide layers or switch among online maps if any. Some layers such as a pie chart glyph layer are also designed to have their own setting panels which allow the user to control the appearance or the property values of the layers, such as glyph colour, glyph scale and glyph opacity.

The multilayered architecture also facilitates the implementation of interaction techniques for the map component. For example, the selection technique is implemented through a selection layer which highlights selected regions. Similarly, the hovering technique is implemented through a hovering layer which highlights the region under mouse hovering; or an area-of-interest selection technique can be implemented through a rectangle layer which highlights the area of interest.

Figure 3.3 illustrates the multilayered architecture of the map component with four typical layers: a background map layer (Google map in this case), a border layer, a coloured region layer and a glyph layer (a pie chart glyph layer in this case). Chapter 4 will describe a number of other map layers and interaction techniques which are developed and added into the map component to solve different geovisual analytics tasks.
3.2 Visualization methods

To facilitate various visual analysis tasks, the framework is designed to provide a wide range of visualization methods including information visualization methods and geovisualization methods. Main visualization methods supported in the framework include:

- a map with dynamic layers for the visualization and analysis of geographic data;
- a parallel coordinate plot, table lens, scatter plot and scatter plot matrix for the visualization and analysis of multivariate data, especially for finding trends, outliers and correlations among variables;
- a bar chart, pie chart and a radar plot for comparing multivariate data items;
- a time graph for the visualization and analysis of time-series data;
- a tree map for the visualization and analysis of hierarchical data including both spatial hierarchical data and demographical hierarchical data.

Figure 3.4 illustrates visualization methods supported in the framework.
Since insight is normally gained through interaction, visualization components in the framework are integrated with various interaction techniques and interactive features to enhance their ability of data exploration and analysis. Interaction techniques implemented in the framework include hovering, brushing, selection, zoom, pan, focusing (i.e. bringing selected items to the centre of view and in case of the map component, the map is zoomed in on the selected regions), dynamic filtering, focus-and-context, drill-down, tooltips, details-on-demand, animation, and so on. For example, the map component is integrated with a rich set of interaction techniques and interactive features: selection, tooltips, zoom, pan, hovering, focusing, focus-and-context, marker placing, distance measuring, annotation, route creating and animation. Many of these techniques and features result in layers of the component as mentioned above. All interaction techniques and their corresponding layers of the map component are configurable. This means they can be added into or removed from the map component based on the user’s need.

Figure 3.5 shows an example of the map component with a number of interaction techniques and interactive features: selection, hovering, marker placing, distance measuring, annotation and tooltips applied to the analysis of a flood scenario in Lisbon [20], as presented in paper VI.

Figure 3.6 shows another example of the map component which is integrated with a
3.3 Interaction techniques

Figure 3.5: The map component integrated with various interaction techniques and interactive features: hovering, selection, marker placing, distance measuring, annotation and tooltips applied to a flood scenario in Lisbon. Selected buildings are displayed with pink borders and a building under hovering is highlighted with a yellow border.

focus-and-context technique and applied to 8000 Italian municipality regions. The user can interactively select an area of interest in the context map in the left view and then all regions in that area or the province will be displayed in the focus map in the top right view [28].

Other components of the framework are also enhanced with a rich set of interaction techniques and interactive features. For example, in addition to common interaction techniques and interactive features such as selection, hovering, tooltips and animation, the parallel coordinates plot is also enhanced with filtering, axis switching and semantic zoom; the scatter plot is enhanced with zoom, scaling and animation; the treemap is enhanced with drill-down techniques; the bar chart and the time graph are enhanced with the focus-and-context technique, and so on. Figure 3.7 shows the bar chart component integrated with a technique which is similar to the fisheye lens technique.

Note: Semantic zoom in the context of this research work is different from ordinary graphical zoom in the aspect that it works on the value space instead of the view space. Nevertheless, values are normally mapped to positions in the view space; in this case, semantic zoom has the similar effect as ordinary graphical zoom.
Figure 3.6: An illustration of the focus-and-context technique applied to the map component to analyze a large regional dataset of 8000 Italian municipality regions. The left view shows the context map, the top right view shows the focus map and the bottom right view shows only the items in the focus map.

Figure 3.7: An illustration of the fisheye lens style technique applied to the bar chart component.
3.4 Perceptual methods

In addition to various interaction techniques, a number of perceptual methods are also implemented in visualization components to increase user perception. For example, a colour map is carefully designed to provide divergent colour scales to highlight regions with low or high values. It also allows the user to design her or his own colour scheme. A dynamic filtering technique is attached to the colour legend to allow the user to filter data based on colour to see only relevant data. Other examples are moving bubbles with trace lines for temporal data in the scatter plot (Figure 3.8) and profiles in parallel coordinates plot (Figure 3.2) to compare two or more regions for multiple indicators.

![Figure 3.8: Trace lines and moving bubbles during animation are used to enhance user perception during the process of analyzing temporal data.](image)

3.5 Linking components for multiple linked views

Multiple linked views is an important feature of visual analytics systems. Many papers [44, 62, 63] emphasize the advantages of multiple linked views for visual data exploration and analysis processes. To facilitate the development of multiple linked views applications, the framework is designed to support a number of linking mechanisms as follows:
- linking based on data, i.e. if a dataset is changed, all views linked to this dataset are updated;

- linking based on selection, i.e. if an item is selected in one view, it will be treated as a selected item in all linked views;

- linking based on filtering, i.e. if an item is filtered out in one view, it will be filtered out in all linked views; filtering can be done based on the attribute value or the colour of items; Figure 3.9 illustrates the linking mechanism based on colour filtering;

- linking based on time animation, i.e. all linked views share the same time animation controller which controls the time period during an animation process; when the time period is changed, all linked views are updated; and

- linking based on colouring, i.e. all linked views share the same colouring scheme; whenever there is a change made on the colouring scheme, all views linked to this colour scheme are updated to reflect this change.

Figure 3.9: An illustration of the linking mechanism based on colour filtering; only items with red colour are displayed, other items are filtered out

To implement these linking mechanisms, the framework implements a number of linking components or managers: a selection manager, a visibility manager, a colour map and an animation manager. The selection manager controls the list of selected items, called selection list. The visibility manager controls the list of the filtering status of items, called visibility list. The colour map controls the colouring scheme and the animation manager controls the time period. A new concept is also introduced: a program context
which contains a dataset, a selection manager, a visibility manager, a colour map and an animation manager. Components sharing the same program context will, therefore, share the same dataset, selection list, visibility list, colouring scheme and time period. Whenever a change happens in the program context, all components using this program context will be informed about the change and each of them will respond to this change by its own way. For example, a click on a region in a map component will change the selection list and this change will be informed to all components (including the map component itself) which share the same program context as the map component. All linked components will then highlight the selected data item corresponding to the selected region.

### 3.6 Dataset model

The framework uses a simple dataset model as a base for storing and communicating data. It is designed to manage data in three dimensions, represented by attribute, space and time (see Figure 3.10) and can communicate the boundaries and content of these dimensions to the components. To make it independent of actual storage structure, a simple interface is defined for data access. This allows different storage structures to be implemented to serve for different purposes. A simple storage structure for datasets implemented in the framework is a 3D array which allows for fast access and can also handle large spatio-temporal, multivariate datasets.

![Figure 3.10: A 3D dataset model of the framework](image)

### 3.7 Architecture of visual analytics applications

The framework facilitates the development of visual analytics applications that have a pipeline architecture of three modules, which is similar to the pipeline of Card et al. [14], as illustrated in Figure 3.11. In this architecture, data is first loaded from data sources into the data provider module and then passed to the data transformation module for analysis
and/or processing before being passed to the visualization module for visualization. The transformations are optional, and an application can even combine visualizations that display the data both before and after the transformations. Visualizations can link back to control the transformations. Therefore, it gives the user direct access to how the data is manipulated. This architecture allows developers to integrate their own components into the framework; for example, they can add new data transform components which implement data analysis algorithms such as the k-means algorithm.

![Figure 3.11: A typical pipeline architecture of visual analytics applications](image)

### 3.8 Data providers

The framework provides a collection of data providers which can load data from a single data source or from multiple data sources and merge them into a single dataset usable by the visualizations. Data sources can be data files or databases accessed through web services. Currently, the framework supports a number of file formats such as Excel, Unicode text, csv, xml, PC-Axis [54] and SDMX [66] and a number of data formats widely used in statistics community, as illustrated in Figure 3.12. The framework is also designed to allow developers to easily add new data providers for specific needs. For example, in the application for the exploration and analysis of flood data [20], a new data provider is developed and added into the framework. This data provider loads flood data stored in a specific format which is designed for fast data reading.

### 3.9 Snapshot for storytelling and collaboration

Storytelling is one of the most impactful ways to teach, learn and persuade. Storytelling, in the context of this research, is about telling a story on the subject of explored data and related analytics reasoning about how the gained knowledge was achieved through interactive visualizations. The framework is designed to support various steps of a visual analytics process including data exploration, data analysis, capturing discoveries, creating stories based on discoveries, sharing stories for collaboration and publishing stories, as illustrated in Figure 3.13.

The research presented in this thesis focuses on the step of capturing discoveries made during a visual analysis process. This is a basic step for creating visual stories. It is done
3.9 Snapshot for storytelling and collaboration

Through a mechanism called ‘snapshot’ that allows capturing the state of all interactive visualization views, the dataset and the layout of the application at the capturing time and saving them as a snapshot into a story file, as illustrated in Figure 3.14.

Snapshots can be captured at any time during an EDA process and they are normally associated with text to form a story which can help the analyst to subsequently guide other analysts or users to follow the discoveries made.

A number of examples of using snapshots for storytelling and collaboration are presented in chapter 4 through applications developed based on the framework.
3.10 Extendibility

To ensure that a specific visual analytics task can be solved by the framework, the framework is designed to allow developers to add new features or new layers into an existing component or rewrite a component to improve its performance or even add new components into it. To facilitate this process, the interface mechanism is extensively employed in the design of components. More specifically, components are based only on interfaces that are necessary to make them work when being connected to other components. Nevertheless, to make the implementation of components easier, the interfaces are kept as simple as possible. This means that they only contain definitions necessary for communicating with the components. For example, in the dataset model above, datasets use a simple interface, called IDataset, which only includes definitions of basic functions such as getValue, getNumRecords, getNumAttributes, getNumSlices and some extra functions such as getMax, getMin and getMean to supply necessary data to other components as well as to avoid the recalculation of the same operation in different components.
3.11 Choices of programming platforms

The development of the framework for web platforms was started in summer 2008. During that time there were only a few programming languages for web platforms such as Microsoft Silverlight and Adobe Flash ActionScript. Among of them Adobe Flash emerged as a de facto standard for the development of web-based applications because of its wide spread use and easy access to internet users. Furthermore, it is an object-oriented programming language which facilitates large-scale web-based applications. Therefore, in 2008, Adobe Flash has been chosen as a platform for the development of the GAV Flash version. Nevertheless, since 2011, there has been an increasing trend of using mobile devices to access internet. For the first time ever in 2011, the number of sales of smartphones exceeded those of PCs including tablet computers [61]; however, not all mobile devices support Adobe Flash, especially Apple ones. To reach a broader user community, we need to find a programming platform which is supported by all modern web browsers and cross device. Since HTML5 and JavaScript meet this condition and are also promoted for future web content, they are chosen for the development of the GAV HTML5/JavaScript version. Using HTML5 and JavaScript facilitates integrating the framework into other web-based frameworks and systems and increases the accessibility of the framework. Nevertheless, choosing Adobe Flash ActionScript in 2008 and then HTML5/JavaScript in early 2012 for the development of the prototypes of the framework also face a number of challenges as discussed in the following section.
3.12 Implementation challenges

Choosing Adobe Flash ActionScript as a programming platform for the development of the framework faces two major challenges when dealing with large datasets and advanced components such as PCP and choropleth map: performance and interactivity. For example, for a PCP, the performance depends largely on the number of lines, the number of line segments of each line, line thickness, line opacity and the view size. For small datasets of less than 2000 items over about 10 attributes, the straightforward approach is to represent a line by a Flash Shape object. This approach facilitates the implementation of line filtering and line colour/opacity transforms through employing the abilities of Flash Player to show/hide objects and transform their colour/opacity without redrawing. Nevertheless, the rendering time increases quickly for larger datasets. This is due to the pixel colour blending process which consumes more time when the number of lines through a single pixel increase. To speed up this process we divide lines into groups, each of which includes a subset of lines; the number of lines in each group can be adjusted by users. Each group is then drawn into a Shape object and then cached as a bitmap. A number of tests are done and the result shows that the approach with 200 lines in a group is from 6 to 51 times faster than the non-grouping approach but uses twice as much memory as the other. It also shows that putting 100 lines in a group does not improve rendering time but takes one and a half times as much memory as putting 200 lines in a group.

The second challenge comes from the fact that Flash Player does not support multithreaded ActionScript, which makes the application unresponsive during the execution of tasks that, in many cases, can take up to tens of seconds, such as drawing a large number of items. Unresponsiveness for a long time can distract the user and make the exploration process of the user less efficient. To address this problem in a similar manner to Piringer [56], a library has been developed to simulate a multithreading environment. The approach is to assign each large task (e.g. drawing 10000 lines) to a so-called micro thread (or a pseudo thread) and to divide each large task into a number of smaller tasks (e.g. drawing 100 lines at a time) so that each small task can be executed in a reasonable time period. Small tasks are then scheduled into frames so that the total execution time of each frame is less than what the given frame rate allows, typically 30 ms per frame. After each frame the control is returned to the Flash Player and allows the user to interact with the application. Threads can be cancelled early due to user interaction, and new threads can be scheduled and executed to reflect user updates. A demonstration of this approach can be found at the link [28].

Choosing HTML5/JavaScript faces a different challenge which comes from the nature of JavaScript. JavaScript is a prototype-based programming language and does not support object-oriented programming (OOP) in the conventional sense. It does not have the concepts of class and class inheritance like conventional object-oriented programming languages, such as C++ or Java, which facilitate the development of large scale systems. To overcome this problem, a number of techniques are implemented to simulate these concepts in JavaScript. For example, a special object type (named Klass) is implemented to simulate the concept of class; inheritance is implemented through the use of prototype
3.13 Conclusions and contributions

This part of the research inherits the component thinking approach, the multilayered component architecture, the linking mechanisms based on selection and filtering, and the concept of snapshot mechanism presented in the papers [36, 37, 38, 40]) to develop two versions of a web-enabled framework.

All components are reviewed, refined and in many cases are re-designed to have better layouts, better interfaces, better implementations, new features and new interaction techniques. For example, the PCP is extended with a setting panel which can be easily accessed by one click. Its layer structure is refined; for example, instead of using one layer for both selected lines and unselected lines, it uses two layers, one for each type of lines. This avoids re-rendering the unselected lines when the colour or the opacity of the selected lines is changed; new layers are also added, such as a line legend layer and a mean line layer.

The scatter plot is re-designed with a new interface and extended with many new features such as moving bubbles with trace lines, zoom, dynamic scaling and labelling.

The filtering mechanism is extended to include more options: filtering by item and filtering by category in addition to filtering by value.

The snapshot mechanism is refined and adapted to the web platform.

The framework has been used to develop a number of geovisual analytics applications such as (1) OECD eXplorer [25] which is developed in collaboration with OECD [53] for visualization, analysis and dissemination of statistic data, (2) ANROSS/VoSON [58] which is developed in close collaboration with the Swedish telecom company Ericsson [16] for analysing the behaviour of automated algorithms used to automatically configure self-organizing networks and (3) RoadVis [48, 82] which is developed in close collaboration with Swedish Meteorological and Hydrological Institute (SMHI) [70] for analysing and communicating information about road weather conditions, particularly during the Swedish winter months. A number of other geovisual analytics applications developed based on the framework, such as a flow map application [32] and a flood application [20], are presented in chapter 4. The framework has also been used to develop applications which are used
A Web-enabled Geovisual Analytics Framework

in a research project to examine the effectiveness, efficiency and user satisfaction of using geovisual analytics tools for teaching students [72].

In summary, this part of the research is believed to contribute to the geovisual analytics field a new framework which is web-enabled, extendible and has the following features:

- including a wide range of highly interactive visualizations which are enhanced with a rich set of interaction techniques and interactive features;
- supporting different projections and online web map services;
- supporting multiple coordinated and linked views through various linking mechanisms;
- supporting a library which simulates a multithreading environment to improve interactivity;
- facilitating the development of web-enabled interactive visualization applications through the component architecture;
- supporting many data formats which are widely used in the statistics community;
- supporting loading and merging data from multiple sources;
- supporting the snapshot mechanism for saving and packaging the results of a geovisual analytics reasoning process – the foundation for visual storytelling;
Chapter 4

Geovisual Analytics for Multivariate Spatio-Temporal Data

This chapter will present how geovisual analytics is applied to a number of specific domains. More specifically, it summarizes a number of geovisual analytics applications developed in this thesis work for multivariate spatio-temporal data and summarizes the contributions of the appended papers, each corresponds to a specific domain. The chapter includes four sections. Section 4.1 presents the approaches and geovisual analytics techniques for time-varying multivariate volume data and summarizes the main contributions of paper III. Section 4.2 presents the approaches and geovisual analytics techniques for multivariate spatio-temporal self-organizing network data and the main contributions of paper IV. Section 4.3 presents the approaches and geovisual analytics techniques for time-varying spatial interaction data and summarizes the main contributions of paper V. Section 4.4 presents the approaches and geovisual analytics techniques for multivariate spatio-temporal flood data and summarizes the main contributions of paper VI.
4.1 Geovisual analytics for time-varying multivariate volume data

The large and ever-increasing amounts of multi-dimensional, multi-source, time-varying and geospatial data have been a major challenge for visualization community. The need for the analysis of this type of data in many cases demands the combination of different techniques from different visualization fields. Scientific visualization (SciVis), information visualization (InfoVis) and geovisualization (GeoVis) have developed almost parallel to each other. The combination and integration of the techniques from these fields have still been rare and have not gained enough attention. This part of the research, therefore, has concentrated on the integration of different techniques from SciVis, InfoVis and GeoVis to enhance the ability of the visual exploration and analysis of time-varying multivariate volume data. It is demonstrated through a prototype for the visual exploration and analysis of ocean temperature and salinity volume data.

4.1.1 Objective

The overall goal of this part of the research has been to encourage and promote synergies between well-known information- and volume data visualization methods applied in a multiple coordinated and linked views interface. The objective has been:

- to extend the GAV C# framework with visualization methods for 4D volume data;
- to develop visual user interface (VUI) and interaction techniques for the exploration of time-varying multivariate volume data;
- to develop a prototype to support the visual exploration and analysis of time-varying multivariate volume data; and
- to demonstrate the synergy of integrating volume visualization with common information visualization and geovisualization interaction techniques and to exploit joint usage in an intuitive and usable manner;

4.1.2 Approaches and results

Time-varying multivariate volume data in the context of this research is data of points in a three-dimensional (3D) space. Each point has a number of attribute values (e.g. temperature, pressure and salinity) and these values are time series. To support the visualization of this type of data, the following visualization techniques (see Figure 4.1) are selected to implement in the GAV C# framework:

- iso-surface;
- 3D orthogonal cutting planes;
4.1 Geovisual analytics for time-varying multivariate volume data

- 2D cutting planes;
- 2D and 3D cursors;
- 2D line graph; and
- volumetric cut-away (or cropping), as illustrated in Figure 4.2 (the middle view).

Figure 4.1: An illustration of the multiple coordinated and linked views of the prototype. The screen usage is also optimized for visual representations. Less than 5% of the screen pixels are devoted to traditional GUIs.

To allow the user to easily interact with and explore data, these visualizations are enhanced with various interaction techniques. For example, the 3D visualizations are integrated with zoom, pan and rotation techniques. The 3D cutting planes can be selected by clicking, moved by dragging-and-dropping or by scrolling with mouse wheel. They can also be moved through linked views such as through 2D cutting planes or sliders (to the right edge of Figure 4.1). The volumetric cut-away technique is enhanced with a selection technique which allows the user to select one of eight sub volumes to be cut-away by just
one click. The 2D cutting planes are enhanced with the techniques: zoom, pan, rotation, horizontal/vertical flipping and selection. All 2D and 3D visualizations are integrated with techniques: brushing and brushing-and-selection (i.e. after brushing the cursor will be moved to the last position of brushing, and this position will be selected and displayed in all linked views). To further facilitate the exploration and analysis of data, a number of linking mechanisms are implemented, such as linking through (1) selection, (2) sliding (i.e. moving cutting planes), (3) time animation, (4) sharing the same colour scheme and (5) data cropping. For example, when a 3D cutting plane is moved, all views linked to that 3D cutting plane are updated; when time period is changed through moving the time slider (the top right slider in Figure 4.1) or through moving the red dot in the time graph (the bottom left view in Figure 4.1), all linked views are updated with the new time period; or when a sub data volume is cut-away (through the volumetric cut-away technique), all linked views are updated with the new data volume (see Figure 4.2).

To allow the user to switch among different modes of interaction, right-click context menus and shortcut keys are implemented. An example of a right-click context menu is illustrated in Figure 4.4.

A dynamic layout mechanism is also implemented to allow the user change the size of views or move around views by dragging-and-dropping.

![Figure 4.2: An illustration of the volumetric cut-away technique and the linking mechanism based on data cropping](image)

To allow the user to get an immediate interactive response (i.e. less than 200 ms on desktop computers) when interacting with data, a number of techniques are implemented...
4.1 Geovisual analytics for time-varying multivariate volume data

to increase the performance of visualizations. For example, data items that are re-used many times (e.g. the min/max values of datasets or the results of comparing an isovalue against values at grid points of a volume data) are pre-calculated. A ‘lazy calculation mechanism’ is also implemented to avoid executing a process several times. The DirectX ‘mesh’ object is used to calculate the normals of an isosurface.

To support the user in storing interactive events in an analytical reasoning process, a simple snapshot mechanism is implemented. The highlighted data points of coordinated views, the status of data volume, etc. can be captured, saved and reused at any time during the EDA process. This feature aids the visual communication of gained insight and constructed knowledge.

To demonstrate a synergy and joint usage among different techniques from SciVis, InfoVis and GeoVis, a prototype is developed based on all techniques mentioned above and other techniques from InfoVis such as scatter plot. The prototype includes four modules as illustrated in Figure 4.3. The module ‘Volume Data Explorer’ includes six linked views: (1) three views each for a 2D cutting plane, (2) one 3D view that integrates an iso-surface and three 3D cutting planes, (3) one time graph view and (4) one depth profile graph view. This module has a main layout as illustrated in Figure 4.1. The views can be moved around by dragging-and-dropping. The module ‘Volume Data Cut-Away’ includes three 2D cutting plane views as mentioned above and a 3D view that implements the volumetric cut-away technique. This module has a main layout as illustrated in Figure 4.2. These two modules are used for the exploration of 4D volume data.

The module ‘Volume Data Correlator’ is designed for finding correlation between two variables (e.g. temperature and salinity) in 5D volume datasets. It includes five linked views as illustrated in Figure 4.4. Two 2D cutting planes are put next to each other and act like two map views, each for one variable. These two views are linked to each other through various interaction techniques such as selection, sub-dataset selection, zoom, pan, brushing and flipping. They are also linked to a scatter plot and line graphs through sub-dataset selection. This means when the user selects an area of interest in one of the two maps, this area will be displayed in the scatter plot and two line graphs. The scatter plot is also linked back to other views through single item selection.

The prototype has been applied to the exploration of ocean temperature data [1], which is 4D volume data. The result from the exploration process shows that while the 3D visualizations allow the user to gain an overall understanding of the data space, the 2D visualizations allow the user to access precise data; interaction techniques and linking mechanisms allow the user to interact with data through different ways and to gain better understanding of data (see Figure 4.1 and Figure 4.2).

The prototype has been also applied to the exploration of ocean temperature and salinity data, which is 5D volume data. Figure 4.4 shows an area of interest containing a small spot which is outstanding from its vicinity.
4.1.3 Conclusions and contributions

This part of the research introduced a geovisual analytics approach to the exploration of time-varying multivariate volume data. It focuses on visual user interface techniques through dynamic data manipulation and multiple coordinated and linked views applied to a large ocean volume dataset. It encourages the synergy among SciVis, InfoVis and GeoVis techniques to enhance the performance of the exploration and analysis of time-varying multivariate volume data. It also takes into account the performance issue by implementing different techniques to improve the performance of visualizations and the prototype.

The main contributions of this part of the research can be summarised as follows:
4.1 Geovisual analytics for time-varying multivariate volume data

Figure 4.4: A screenshot of five coordinated and linked views for comparing temperature and salinity using 2D image maps, 2D scatter and 2D depth graph views. A comparison of temperature vs. salinity inside an area of interest is presented for time step August (7). The user has selected an area of special interest with high salinity values outside the coast of South America. Volume data inside the selected area are shown simultaneously in five coordinated and linked views. Correlation between the two attributes for a time step is analysed in a 2D scatter and two depth profiles. The value at position (0.86, 0.69) is highlighted in all views. A right-click context menu pops up after a mouse double click and reveals possible interactive operations available for current view (the right 2D image map). The user has selected the operation ‘select area’ which is then applied until a new operation is requested. Shortcut keys are provided for the same purpose. A dynamic time slider updates all linked views with immediate response time.

- means to interact dynamically with large-scale 4D or 5D volume data with immediate response time (i.e. less than 200 milliseconds on desktop computers); interaction mechanisms to support the visual analysis of time-varying multivariate volume data;
- a snapshot mechanism for capturing, saving and reusing interactive views;
- demonstration of the synergy of integrating volume visualization with common information visualization and geovisualization interaction techniques to facilitate exploration and discovery;
- a research toolkit platform for evaluating and implementing synergies between Info-
Vis, GeoVis and SciVis methods; and

- a prototype for the visual exploration and analysis of time-varying multivariate volume data;

The prototype can be further extended to include other visualization methods such as a parallel coordinate plot and map glyph layers to provide the user with more means to explore and analyse time-varying multivariate volume data.
4.2 Geovisual analytics for self-organizing network data

Cellular radio networks such as mobile networks are continually growing in both node count and complexity. It therefore becomes more difficult to manually configure the networks and necessary to use time and cost effective algorithms to automatically configure the networks. There have been a number of attempts to develop such automated algorithms. For instance, Ericsson [16], a Swedish telecommunication company, has developed an algorithm, called Automatic Neighbor Relations (ANR) [2], which automatically creates and updates neighbour cell relations as well as assigns and updates physical cell identities to avoid local conflicts. A network operated under an automated algorithm is called a self-organizing network. Network operators, however, may not trust automated algorithms because they need to have an understanding of their behaviour, reliability and efficiency, which is not easily perceived. The challenge, therefore, is to develop tools to support network operators to understand the way automated algorithms work. To provide better understanding of the approaches used in this part of the research, the next paragraph will briefly describe self-organizing network data.

Self-organizing network data mainly includes three datasets: a cell dataset, a relation dataset and a network event dataset. The cell dataset contains cell properties and cell counts such as cell global identity (CGI), physical cell identity (PCI), cell location, cell direction and number of call drops. The relation dataset contains property values and counts of cell relations such as source cell, target cell, number of handover successes and number of handover failures. A handover is a process of transferring an ongoing call from one cell to another cell. This may happen, for example, when people are talking on the phone on a moving train. The network event dataset is a collection of network events which describe how the network is configured or evolved under the operation of an automated algorithm. Network events include relations added, relations removed, relations pushed in a black list (i.e. relations should not be used), PCI-conflicts, PCI-conflicts detected and PCI updates (i.e. cells changing one PCI to other PCI). In these events, PCI-conflict detected events and PCI update events are the most important ones, since they show how well an algorithm detects conflicts in the network and reconfigures the network.

4.2.1 Objective

The overall goal of this part of the research is to support network operators to understand the behaviour, reliability and efficiency of the automated algorithms used to configure the cellular radio networks. The objective is to develop a geovisual analytics system toward to this goal. The system should be able to

- visualize the data described above;
- show the way how an automated algorithm such as ANR configures a cellular radio network; or in other word, how a cellular radio network is evolved under the operation of an automated algorithm;
support the network operator answering a number of questions such as why a cell changes its PCI, why a relation is added or removed, or how a PCI-conflict is detected; and

- detect abnormal or extreme cells and cell relations so that the network operator can pay more attention on these cells or cell relations.

### 4.2.2 Approaches and results

The system is developed based on the user-centred approach. To support network operators to easily and quickly identify what happen in the network, where they happen and why they happen, a map is suitable for this purpose. However, since the self-organizing network data contains two datasets: a cell dataset and a relation dataset, it is logical to have two maps, one for the cell dataset and the other for the relation dataset. This approach also helps to reduce the clutter which may happen if we use only one map for both datasets. Similarly, each map is divided into a number of layers, each layer representing for one property (or attribute), or one type of count, or one type of event. For property values or counts, coloured-polygon and glyph layers are used. For events which involve only one cell (e.g. PCI-conflicts and PCI-updates), it is good to use glyph layers to represent them. However, for events which involve more than one cell (e.g. PCI-conflicts detected), it needs a special method to represent. A PCI-conflict detected event normally involves three cells: two cells which have a PCI conflict and a third cell which detects this conflict. This type of event also contains the information of the method that is used to detect the conflict, for example, the method of CGI report (CGIR) or the method of mutual relation addition (MRA). For each event of this type, it is good to use two arrows and one glyph to represent it. Two arrows represent the relationship among three cells and the glyph represents the method used to detect the conflict. To distinguish different methods used for detecting conflicts, arrows are coloured differently based on the method used. Figure 4.5 illustrates a number of cell map layers in which PCI-conflict detected events associated with the method of MRA are represented by pink arrows and PCI-conflict detected events associated with the method of CGIR are represented by yellow arrows.

To reduce the overlapping among layers of the cell map, a polygon representing a cell is partitioned into areas, as illustrated in Figure 4.6; each area is reserved for a layer. Since a cell, in some cases, can be selected as a source cell and a target cell at the same time, to represent this cell, the polygon for the cell includes two borders with two different colours, one for the role of a source cell and the other for the role of a target cell, as illustrated in Figure 4.6.

To visualize relations, arrows are selected because of their simplicity. To distinguish different relation types such as relations added, relations removed or relations in a black list, arrows are enhanced with colours and glyphs, as illustrated in Figure 4.7. Figure 4.8 illustrates the relation map of the system.

In addition to these two maps, a number of other interactive visualizations are selected to facilitate the exploration and analysis of data. For example, two parallel coordinate
Figure 4.5: An illustration of network events occurring in a time interval

Figure 4.6: (a) a polygon partitioned into many areas, each for a layer, to reduce the overlapping among cell map layers; the polygon can have two borders, one for source cell (the red border) and one for target cell (the yellow border)
Figure 4.7: Arrows (with border), glyphs and colours are combined to represent various types of relations.

Figure 4.8: An illustration of using arrows with borders together glyphs and colours to represent various types of relations.

plots, one for cell data and the other for relation data, are used to support filtering data, finding extreme cells and relations, finding correlation among cell counts or relation counts, and so on. Similarly, two time graphs are used to allow the network operator to supervise the performance of selected cells and relations over time. A special time slider enhanced
with glyphs (see Figure 4.9) is designed to allow network operators to control the selection of time interval, time animation and especially to highlight time intervals of change (i.e. intervals in which cells change their PCIs or their relations) of selected cells or the whole network. This can help network operators to easily follow and analyse the changes in the network. Tooltips are also used to provide detailed information of objects which the user hovers on.

Figure 4.9: A time slider enhanced with glyphs to allow network operators to quickly identify time intervals in which network events of a certain type happen. Yellow squares represents time intervals of relation updates and red squares represent time intervals of PCI updates.

To enhance the performance of visual data analysis processes, the multiple coordinated and linked views approach is applied. A number of linking mechanisms are implemented, such as linking based on selection, filtering, colouring and animation. Among these linking

Figure 4.10: The layout with seven coordinated and linked views of the system: two maps are used to visualize, interact with and analyze cells and relations; two PCPs are used to explore and examine cell and relation attributes/counts in each time interval; two graphs are used to explore and examine cell and relation attributes/counts over time; a time slider is used to move back and forth and highlight “intervals of change” of selected cells.

To enhance the performance of visual data analysis processes, the multiple coordinated and linked views approach is applied. A number of linking mechanisms are implemented, such as linking based on selection, filtering, colouring and animation. Among these linking
mechanisms, the most difficult ones are those which link views of cell data with views of relation data and vice versa; for example, when a cell is selected in a view of cell data (e.g. the cell map in Figure 4.10), how the relation map should respond; and when a relation is selected in a view of relation data (e.g. the relation-based PCP in Figure 4.10) how the cell map should respond. The solution suggested in this part of the research is that when a cell is selected in a view of cell data, relations of this cell are selected in all views of relation data; and when a relation is selected in a view of relation data, the source cell of this relation is selected in all views of cell data.

The system has been used to analyse the self-organizing network data in three cases that are selected to simulate the behaviour of the ANR algorithm; each case includes 225 cells prepared. The simulation is done for one day, including 96 intervals of 15 minutes.

The result shows that the system allows the user to supervise changes in the network; for example, what cells change their relations, their PCIs, what cells have a PCI conflict, how a PCI conflict is detected, and so on. Figure 4.11 shows a situation in which a cell has 3 handover failures in time interval 5 and then a PCI conflict involving this cell is detected in time interval 8 through the method of CGI report (CGIR). The result also shows that the system allows the user to detect notable cells or cell relations such as cells with a large number of call drops, relations with a large number of handover failures or relations with long distances, as illustrated in Figure 4.12, which need more information to explain.

The system has been also evaluated by five network domain experts. The feedback shows that they like the concept of multiple coordinated and linked views which allows them to supervise changes in the network. They also like the ability to find ‘problem’ cells and relations. However, they also suggest a number of features which should be considered for further development; for example, the ability to pick lines on time graphs and link them to other views, and the ability to find extreme values in all time intervals instead of in only one interval.

4.2.3 Conclusions and contributions

This part of the research presented the approaches and geovisual analytics techniques for self-organizing network data which includes three multivariate spatio-temporal datasets. A geovisual analytics system has been developed to meet the objectives described above. The system has been evaluated by network domain experts and received positive feedbacks.

The main contributions of this part of the research can be summarized as follows:

- the concept of a multi-border and multi-sub-area polygon for displaying various types of cell information and data and reduce the overlapping among map layers;
- a visualization method which combines an arrow with different glyphs to represent different types of cell relations;
- a visualization method for representing PCI-conflict detected events;
- a time slider enhanced with glyphs to highlight notable time intervals;
4.2 Geovisual analytics for self-organizing network data

Figure 4.11: (a) Intervals of change of cell 210; (b) after 3 handover failures in interval 5 from cell 224 to cell 225, a PCI conflict between cells 210 and 225 detected in interval 8 by cell 224

• mechanisms for linking between visualization components for cell data and visualization components for cell relation data and vice versa; and

• a geovisual analytics system which combines all interactive visualizations and interaction techniques mentioned above for the visual analysis of self-organizing network data.
The research has been continued with further evaluation and improvement. More requirements from the users are added into the system. The system has also been updated to reflect the improvements of the GAV Flash framework based on which the system is developed. Figure 4.13 shows a screenshot of the new system in which glyph shapes and the colours are improved to meet requirements of the users; the two arrows of PCI-conflict detected events are also connected through a curve to indicate that they are of the same event.

Figure 4.13: The updated system with a new layout, a new colour scheme and new glyph shapes
4.3 Geovisual analytics for spatial interaction time series data

Spatial interaction data (or movement data) such as trade flows, commuting flows and migration flows may contain information or movement trends which are valuable to policy makers. Understanding these trends as well as the reasons behind these trends can help them have better policies to improve the society. Communicating the gained insight to the citizens is also important, since it can help the citizens understand their society better. Since there has been a lot of research focusing on giving an overview of spatial interactions, this part of the research has focused more on the aspects of user interactions, visual exploration, analysis and knowledge communication.

4.3.1 Objective

The objective of this part of the research has been to develop and evaluate a flow map demonstrator that is web-enabled and should support:

- both exploration and presentation of flow data;
- interaction with spatial interaction time series data with interactive performance;
- animation of time series data;
- different background map layers, e.g. Google map or Bing map, for identifying the name of geographic locations;
- dynamic queries and filtering operations;
- time series glyphs to show changes over time;
- a dynamic bar chart to show order and magnitudes of flow values;
- a parallel coordinates plot for detecting correlations and filtering data;
- answering various questions about flow data, such as:
  - Which are the dominant flows (or the trend of movement) in a certain year?
  - Which are the top municipalities in Norway to which people living in Swedish border municipalities tend to commute or migrate?
  - What is the net migration i.e. the difference between the outflow and the inflow of a region for another region?
  - How do flows vary over time?
- gaining insight why people tend to move from one location to another;

The requirements listed above are based on the needs of the target end users (statistic analysts and policy makers) of the system.
4.3.2 Approaches and results

To visualize flow data, a straightforward approach used in this part of the research is to use directed weighted arrows, where each arrow represents a movement from an origin to a destination and the thickness of the arrow represents the number of people. Arrow thickness can be dynamically scaled to make arrows more readable; however, they always reflect the values they represent. To avoid overlapping, arrows are displayed as quadratic Bezier curves whose curvatures can be dynamically adjusted (see the right view in Figure 4.14).

![Figure 4.14: Migration flows from Oslo to Swedish counties and vice versa during the period 2006-2009. The bar chart shows (1) the volume of migration flows to Oslo (blue bars) to Sweden, (2) the volume of migration flows from Oslo to Sweden (green bars), and (3) the difference (or net value) between outflows and inflows (red bars) which shows that there is a positive trend of migrating to Oslo from Swedish counties. The blue arrows (right part) show top five migration flows from Swedish counties to Oslo. The green arrows show inverse flows from Oslo to Swedish counties. The bar chart glyphs (blue ones) show time-series values of top five migration flows from Swedish counties to Oslo. The polygon layer (or region layer) is coloured according to migration volumes from Swedish counties to Oslo.](image)

The second approach is to use coloured-shaded regions. This approach can be applied to the case in which the user wants to explore flows originating from an origin. In this case, each flow corresponds to a destination and therefore can be visualized by colouring the region representing that destination (see the right view in Figure 4.14).
4.3 Geovisual analytics for spatial interaction time series data

To facilitate the exploration of flow data as well as answering questions such as what are the top flows in global and what are the top outflows or inflows of a region, the system is designed to have three modes: a global top flow mode, a local top flow mode and a free flow mode. The global top flow mode allows the user to find a range of flows (e.g. top five largest or smallest flows) in a time period. The local top mode allows the user to find a range of flows coming to or going out from a region (e.g. top five largest or smallest coming in or going out from a region) in a time period. The free flow mode allows the user to select any flow to examine; for example, the user can select the export flows among America, Germany and Japan for comparison. Flows can be selected through the map by clicking an origin and then a destination or through a data table of flows.

Tooltips and information panels are also used to provide detailed information for flows hovered or selected.

To visualise time series flow data, the first approach is to use a time chart glyph layer. A time chart glyph can be a bar chart or a line chart, as illustrated in Figure 4.14 and Figure 4.15. In addition to the time chart glyph layer, a trend glyph layer is also implemented. In a trend glyph, the glyph size represents the average amount of people moving or commuting and the glyph slope shows the trend of the movement - increasing or decreasing. The second approach to the visualization of time series flow data is through simultaneous animation of arrows, coloured regions and linked views as mentioned below. The animation can be started and stopped at any time point, and the animation speed can be also controlled to allow the user to observe changes easily.

![Figure 4.15: Time line chart glyphs represent time-series inflows and outflows of regions](image)

To support the user in exploring further and gaining insight into flow data, a number of interactive visualizations are added into the system such as a bar chart integrated focus-and-context technique and a parallel coordinate plot, which are inherited from the GAV Flash framework. The approach of multiple coordinated and linked views is also applied to enhance the performance of exploration and analysis process. The linking mechanisms from the GAV Flash framework such as linking based on selection, filtering, colouring and
animation are inherited. For example, when a region is selected in the map view, values of inflows and outflows of this region are displayed in other views. The net values (i.e. the difference between outflows and inflows), total values (e.g. total inflow values, total outflow values) are also dynamically calculated and displayed in all other views to provide the user with additional data for the analysis of flow data. Two circle glyph layers are implemented to represent total inflow values and total outflow values (see Figure 4.17).

To support the user in answering questions such as why people tend to move from one location to another, the system supports the use of additional regional datasets. For example, the user can use a regional dataset of income level, employment rate and higher education rate to have a better understanding of the trend that more people commute from Sweden to Norway for working rather than the opposite. This regional dataset is then merged with the data dynamically calculated above (net values, total inflow values, total outflow values) and can be displayed in all interactive visualizations.

To support the presentation and communication of the gained insight, the system inherits the mechanisms of snapshot, storytelling and publishing of the GAV Flash framework. It implements the snapshot mechanism for the flow layers and glyph layers which are developed in this part of the research. It also extends the snapshot mechanism to store two different datasets (the flow dataset and the regional dataset) into snapshots.

Figure 4.16: Top ten global commuting flows among Norwegian and Swedish municipalities in 2008
4.3 Geovisual analytics for spatial interaction time series data

Figure 4.17: Top five hubs-in (blue circles) and hubs-out (blue circle) for commuting in 2008; the size of circles represents the volume of commuters.

The developed system has been applied to two different flow datasets to find trends of movement and gain insight into these flow datasets. The first flow dataset contains data of commuting across the border between Norway and Sweden from year 2001 to year 2008 at municipality level. Based on using the global top flow mode to find dominant flows in 2008 it is found that (1) top five flows are from municipalities of Sweden to Oslo of Norway; (2) in top ten flows there is only one flow from Norway to Sweden: the flow from Halden (Norway) to Strömstad (Sweden) (see Figure 4.16); the map shows that these two municipalities are neighbours of each other, which may explain this fact. This trend is still repeated in all time periods from 2001 to 2008.

Similarly, based on filtering on total inflow values and total outflow values it is found that (1) Göteborg, Filipstad, Karlstad, Strömstad, Stockholm and Torsby of Sweden are the biggest hubs of people commuting out and (2) Oslo of Norway is the biggest hubs of people commuting in (see Figure 4.17).

The second flow dataset contains data of commuting among 290 municipalities in Sweden in 2008. In addition, a regional dataset containing indicators such as population, average income, high education level and unemployment rate is also used to support the analysis of the flow data. The result of the analysis process shows that Stockholm and Göteborg are two destinations of largest commuting flows, as illustrated in Figure 4.18. Based on the PCP (as shown in Figure 4.18) it is also found out that Stockholm and Göteborg have high higher education rates and high income levels which may explain this fact.

The system has been evaluated by a number of statistics experts at Statistics Sweden, Linköping municipality, Region Västra Götaland and the Norwegian county Ostfold Fylke who used it to analyse in depth the geographic structures and correlations in commuting and migration flows in Sweden and between Sweden and Norway and has resulted in good stories for presentation and communication. The general feedback is positive, especially the
combination of linked views, a ‘heat map’ with overlaid weighted arrows and a dynamic bar chart integrated with focus-and-context technique, which allows a comprehensive insight into the number of people commuting among Swedish regions.

Figure 4.18: Top ten commuting flows among Swedish municipalities in 2008; regions are coloured according to higher education rate

Figure 4.19: Stockholm and Gøteborg have high income and higher education rate and attract a large number of commuters.

4.3.3 Conclusions and contributions

This part of the research presented the approaches and geovisual analytics techniques for spatial interaction dat. It has focused on the aspects of user interactions, visual exploration, analysis and knowledge communication. A demonstrator has been developed and meets the objectives described above. The demonstrator has been used by many statistics analysts for various spatial interaction datasets such as commuting data, migration data and trade flow data.

The main contribution of this part of the research can be summarized as follows:
interaction techniques for the exploration of flow data;

interactive flow and glyph layers;

the use of additional regional datasets in combination with flow datasets to support the analysis of the flow datasets;

an implementation of the snapshot mechanism and the use of storytelling and publishing techniques for flow data which can be enhanced with regional data; this encourages the collaboration between statistics analysts and users of statistics as well as supports the presentation and communication of gained insight to a broad range of users through the web platform.

a web-enabled demonstrator which can be customized to analyse various flow datasets.

The flow map demonstrator has been further developed and evaluated after this part of the research work is published. A number of new features are suggested; for example, (1) there should be an option for the user to decide the layout of views; and (2) scatter plots and time graphs for flow data and regional data should be added into the system to enhance the ability to explore the data. It is also used to analyse global trade flow data provided by OECD, as illustrated in Figure 4.20.
4.4 Geovisual analytics for flood data

The large and ever-increasing amounts of multi-dimensional, multi-source, time-varying and geospatial digital information represent a major challenge for the analyst. The need to analyse and make decisions based on these information streams, often in time-critical situations, demands efficient, integrated and interactive tools that help the user to explore, present, collaborate and communicate visually large information spaces. A serious flood event which took place on 29 October 2010 in Lisbon and caused a lot of damage to buildings, especially historical buildings, presents an example of this need. Planners, policy makers and insurance companies would like to have an interactive tool to explore the flood event data, estimate the damage caused by the flood event, support making plan for rescuing or evacuating people and communicate the analytical results. This part of the research addresses this need and proposes a system for the visual analysis of flood data developed based on the GAV Flash framework.

4.4.1 Objective

The objective of this part of the research has been to develop an interactive system which should support:

- the exploration of flood data;
- estimating the damage caused by a flood event to buildings;
- planning routes to rescue or evacuate people;
- making-note for collaboration;
- presentation and communication of gained insight;

4.4.2 Approaches and results

The system is developed based on the GAV Flash framework in close collaboration with our research partners and domain experts from the Laboratory for Systems, Instrumentation and Modeling in Science and Technology for Space and the Environment (SIM) [77], UNINOVA [29] and King’s College London [41] in a project funded by the European Space Agency (ESA) [24]. A map view is selected as the central view of the system. Based on the characteristics of the flood data and the needs of the user, a number of new map layers and new map techniques are implemented and added into the map component. They include:

- a grid layer for flood water column (or water depth) data; grid cells are coloured based on their water depth values; and a colour scheme for flood data is designed based on the suggestion of domain experts;
- a building layer for buildings which allows the user to be able to interact with buildings and support calculating the damage caused by a flood event to the buildings;
4.4 Geovisual analytics for flood data

- a marker layer for marking special locations or locations of interest;
- a polyline layer for drawing routes and measuring the length of routes;
- a traffic route layer for finding traffic routes between locations; this layer is implemented based on the traffic route feature of online maps;
- a note layer for making note and supporting collaboration among analysts; the notes can be interactively moved around the location to which it links; the note width can be interactively resized, but the note height is automatically calculated to fit the note content;
- selection layers representing selected grid cells and selected buildings;
- a brushing layer for highlighting buildings under mouse hovering;
- a hyperlink layer for marking buildings which have a link to an external source (e.g. a website describing the architecture of a building);

A number of setting panels are implemented to allow the user to (1) change property values of layers such as note colour, note scale, route thickness, route colour and route opacity, (2) show or hide layers or only a number of items of layers, and (3) delete items from layers.

Figure 4.21: An illustration of a number of layers of the map view
A number of interaction techniques are designed to allow the user to dynamically and interactively add markers, routes and notes. This is implemented through a right-click context menu which allows the user to switch among various interaction techniques.

To support evaluating the effects of the flood event on buildings, a number of data indicators for buildings are calculated. For example, for each building, the intersection between the building and the flood grid is calculated. The water columns at the cells intersecting with the building are then used to calculate the average water column surrounding the building. This value is then used to calculate the constant damage to the building. The accumulated damage to the building is calculated based on constant damage values.

To facilitate the visual analysis of flood data and calculated building data, the system inherits from the GAV Flash framework the following features:

- interactive visualizations such as the time graph, the scatter plot and the bar charts for flood data and building data;
- linking mechanisms such as linking based on colouring, selection, focusing and animation;
- the snapshot mechanism for capturing, saving visual analysis processes; the snapshot mechanism is implemented for the new map layers that are added into the map component; and
4.4 Geovisual analytics for flood data

- the storytelling technique for sharing, presentation and communication of gained insight;

The system has been used to gain insight into the flood event which took place on 29 October 2010 in Lisbon. It allows the user to detect areas affected by the flood event and estimate the damage to buildings. It has been also evaluated by three users. The general feedback is positive. They highly evaluate the features of the system, especially the ability to dynamically make notes and storytelling technique. They like the animation mechanism which allows them to supervise the progress of the flood event and the linking mechanism through selection which allows them to view the data from several perspectives. After one hour using the system, one user has produced a short story presenting his gained insight into the flood event data. Figure 4.22 illustrates a snapshot in this story. The story is available at the link [20]. A video illustrating the features of the system is available at the link [21]. New features are suggested for further development such as the ability to dynamically update the formulas for damage estimation, (2) map layers representing police stations, schools, and (3) the ability to export a subset of data of interest.

4.4.3 Conclusions and contributions

This part of the research presented the approaches and geovisual analytics techniques for multivariate, spatio-temporal data flood data. A system has been developed to meet the objectives described above. It has been evaluated by domain experts and received positive feedbacks. This part of the research is believed to have the following contributions:

- interactive map layers for making notes and creating routes which are enhanced with labels displaying distances between points of the routes;
- interaction techniques for creating different map layers such as note layer, route layer and marker layer;
- a technique for dynamically calculating the damage caused by a flood event to buildings;
- a web-enabled system for the visual exploration and analysis of flood data and supporting storytelling, collaboration and dissemination of gained insight.

The next steps are to (1) further evaluate the system with planners and insurance companies in Lisbon, (2) test it with real time data which comes every 10 minutes, and (3) develop techniques to improve the performance for rendering large flood datasets.
Chapter 5

Conclusions and Future Research

This chapter will summarize the research work presented in this thesis including its main contributions and then discuss potential directions for future research.

5.1 Conclusions and contributions

The research work presented in this thesis has been inspired by the book Illuminating the Path: The Research and Development Agenda for Visual Analytics [78] which (1) addresses the needs as well as challenges for combining technologies from various areas to gain insight into data, which in many cases are very complex and dynamic, and (2) calls for moving research into practice. This research has focused on two subjects. The first subject is the development of a geovisual analytics framework which supports the integration of exploratory, analytical reasoning and communicative tools to facilitate the development of applications for geovisual analytics tasks. In this subject, it is targeted at the web environment to meet the growing needs for the exploration and analysis of data and the presentation and dissemination of gained insight in the form of interactive visualizations through web and internet to reach a broader user community. Although many geovisual analytics tools have been developed, it has still been a challenge to adapt these tools to the Internet.

The second subject of this research work is the development of applications which apply geovisual analytics techniques to solving different problems in a number of specific domains.

For the first subject, it has adopted the component thinking in building a web-enabled geovisual analytics framework. The challenges for building frameworks for web platforms have been addressed. Many visualizations have been integrated into the framework. Visualizations have been enhanced with many interaction techniques and interactive features to improve their ability of data analysis. The importance of multiple linked views has been highlighted. Many linking mechanisms have been integrated into the framework and new linking mechanism have been introduced. The need for capturing and sharing gained insight during the EDA process has been emphasized. A snapshot mechanism has been implemented to support this task. A number of applications have been developed to solve different problems in different specific domains. Storytelling techniques have been integrated into a number of developed applications to support telling stories of data. In summary, the main contributions of this research work can be considered as follows:
Conclusions and Future Research

- a collection of highly interactive visualizations which are enhanced with a rich set of interaction techniques and interactive features; especially an interactive map component which is extendable and configurable, including a rich set of interactive map layers and supporting the use of online web map services as a background map;

- mechanisms for linking views in multiple views applications;

- a library which simulates a multithreading environment in Adobe Flash ActionScript to improve interactivity;

- a snapshot mechanism for capturing and saving discoveries, which is the foundation for the visual storytelling technique;

- an effective web-enabled geovisual analytics framework which integrates all features mentioned above and facilitates the development of web-enabled geovisual analytics applications;

- demonstration of the synergy of integrating volume visualization with common information visualization and geovisualization interaction techniques to support visual exploration and analysis of data;

- an application for the visual exploration of 5D volume ocean data;

- techniques and an application for the visual exploration and analysis of spatio-temporal multivariate self-organizing network data to support the network operator have better understanding of the way an automated algorithm configures a network;

- techniques and an application for the visual exploration and analysis of spatial interaction time series data based on the needs of target end users; especially the use of additional regional datasets in combination with flow datasets to support the analysis of the flow datasets;

- techniques and an application for the visual exploration and analysis of spatio-temporal multivariate flood data;

The framework has been used to develop a number of web-enabled visualization applications which is being used daily by many users. For example, OECD eXplorer [25] is being used by OECD [53] for analysing data and presenting analytical results to the public through interactive attractive visualizations to capture their attention as well as to help them understand better our society and our world through statistic data. The RoadVis application [82] is being used by Swedish Meteorological and Hydrological Institute (SMHI) [70] for analysing and communicating information about road weather conditions, particularly during the Swedish winter months. The flow map application (named Flow map eXplorer) for spatial interaction time series data are being used by OECD and many other statistic organizations for the visualization and presentation of commuting data, migration data and trade flow data to the public. The framework and applications have been continuously improved and updated to meet more requirements from the user.
5.2 Future research

Although there are many visualizations and interaction techniques supported in the framework, users and analysts still want to have more visualizations and interaction techniques to do their jobs better and faster. This is one of the focuses of the future work.

The second focus is to adapt visualizations and interaction techniques to different screen sizes and devices such as laptops, desktops, tablets and smart phones, since there is an increasing trend in which the user uses mobile devices to access data and present data as well as analytical results to others. Therefore, visualizations should be designed to adapt well to different screen sizes, and interaction techniques for touch screen and mobile devices should be considered. Thorough user evaluation should be performed to determine what visualizations and interaction techniques are best suited to each user groups as well as to each type of devices (e.g. desktops, laptops, tablets, smart phones, touch screens).

The third focus is the visual analysis of large datasets. Data collected in many fields is getting larger and larger. Therefore, we need effective techniques to deal with large datasets. One possible solution is to combine the focus-and-context technique with effective downsampling techniques and rendering techniques that are based on the GPU technology. Another approach to deal with large datasets is to apply data mining techniques for extracting patterns and trends in large datasets. Parallel computing can also contribute to solutions for large datasets. Therefore, it should be considered in future research.
Conclusions and Future Research
Bibliography


[24] European Space Agency (ESA). http://www.esa.int/ESA.


BIBLIOGRAPHY


[70] SMHI. http://www.smhi.se/.


[82] RoadVis video. [http://vitagate.itn.liu.se/GAV/videos/VDA/RoadVis/RoadVis_full.wmv](http://vitagate.itn.liu.se/GAV/videos/VDA/RoadVis/RoadVis_full.wmv).


