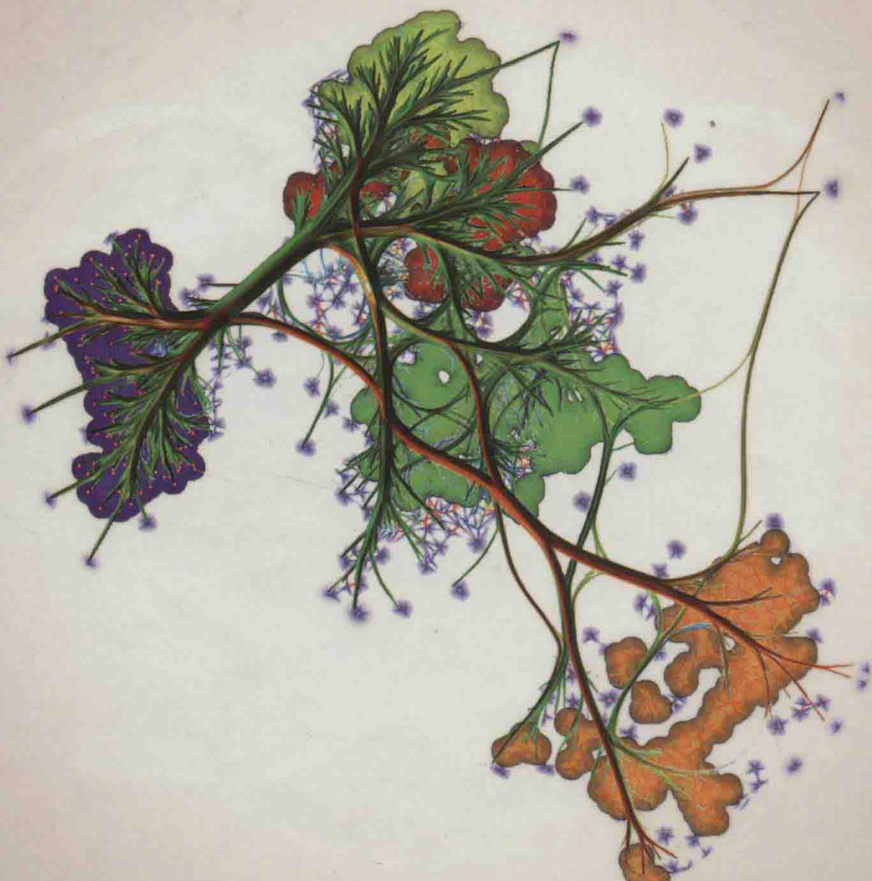
CRC Press
Taylor & Francis Group
AN A K PETERS BOOK

# DATA VISUALIZATION
## PRINCIPLES AND PRACTICE

SECOND EDITION



ALEXANDRU C. TELEA

# Data Visualization

## Principles and Practice

### Second Edition

Alexandru Telea



CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
AN A K PETERS BOOK

First edition published in 2007 by A K Peters, Ltd.

**Cover image:** The cover shows the combination of scientific visualization and information visualization techniques for the exploration of the quality of a dimensionality reduction (DR) algorithm for multivariate data. A 19-dimensional dataset is projected to a 2D point cloud. False-positive projection errors are shown by the alpha-blended colored textures surrounding the points. The five most important point groups, indicating topics in the input dataset, are shown using image-based shaded cushions colored by group identity. The bundled graph shown atop groups highlights the all-pairs false-negative projection errors and is constructed by a mix of geometric and image-based techniques. For details, see Section 11.5.7, page 524, and [Martins et al. 14].

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742



Printed and bound in India by Replika Press Pvt. Ltd.

Printed on acid-free paper
Version Date: 20140514

International Standard Book Number-13: 978-1-4665-8526-3 (Hardback)



**Library of Congress Cataloging-in-Publication Data**

Telea, Alexandru C., 1972-
Data visualization : principles and practice / Alexandru C. Telea. -- Second edition.
pages cm
Summary: "This book explores the study of processing and visually representing data sets. Data visualization is closely related to information graphics, information visualization, scientific visualization, and statistical graphics. This second edition presents a better treatment of the relationship between traditional scientific visualization and information visualization, a description of the emerging field of visual analytics, and updated techniques using the GPU and new generations of software tools and packages. This edition is also enhanced with exercises and downloadable code and data sets"-- Provided by publisher.
Includes bibliographical references and index.
ISBN 978-1-4665-8526-3 (hardback)
1. Computer graphics. 2. Information visualization. I. Title.

T385.T452 2014
001.4'226028566--dc23 2014008643

**Visit the Taylor & Francis Web site at**
**http://www.taylorandfrancis.com**

**and the CRC Press Web site at**
**http://www.crcpress.com**

# Data Visualization

To my family

# Preface to Second Edition

THIS is the second edition of the book *Data Visualization: Principles and Practice* first published in 2008. Since its first edition, many advances and developments have been witnessed by the data-visualization field. Several techniques and methods have evolved from the research arena into the practitioner's toolset. Other techniques have been improved by new implementations or algorithms that capitalize on the increased computing power available in mainstream desktop or laptop computers. Different implementation and dissemination technologies, such as those based on the many facets of the Internet, have become increasingly important. Finally, existing application areas have gained increasing importance, such as those addressed by information visualization, and new application areas at the crossroads of several disciplines have emerged.

The second edition of this book provides a revised and refined view on data visualization principles and practice. The structure of the book in terms of chapters treating various visualization techniques was kept the same. So was the bottom-up approach that starts with the representation of discrete data, and continues with the description of the visualization pipeline, followed by a presentation of visualization techniques for increasingly complex data types (scalar, vector, tensor, domain modeling, images, volumes, and non-spatial datasets). The second edition revises and extends the presented material by covering a significant number of additional visualization algorithms and techniques, as follows. Chapter 1 positions the book in the broad context of scientific visualization, information visualization, and visual analytics, and also with respect to other books in the current visualization literature. Chapter 2 completes the transitional overview from graphics to visualization by listing the complete elements of a simple but self-contained OpenGL visualization application. Chapter 3 covers the gridless interpolation of scattered datasets in more detail. Chapter 4 describes the desirable properties of a good visualization mapping in more detail, based on a concrete example. Chapter 5 discusses colormap design in additional detail, and also presents the enridged plots technique. Chapter 6 extends the set of vector visualization techniques discussed with a more detailed discussion of stream objects, including densely seeded streamlines, streaklines, stream surfaces, streak surfaces, vector field topology, and illustrative techniques. Chapter 7 introduces new examples of combined techniques for diffusion tensor imaging (DTI) visualization, and discusses also illustrative fiber track rendering and fiber bundling techniques. Chapter 8 introduces additional techniques for point-cloud reconstruction such as non-manifold classification, alpha shapes, ball pivoting, Poisson reconstruction, and sphere splatting. For mesh refinement, the Loop subdivision algorithm is discussed. Chapter 9 presents six additional advanced image segmentation algorithms (active contours, graph cuts, mean shift, superpixels, level sets, and dense skeletons). The shape analysis discussion is further refined by presenting several recent algorithms for surface and curve skeleton extraction. Chapter 10 presents several new examples of volume rendering. Chapter 11 has known arguably the largest expansion, as it covers several additional information visualization techniques (simplified edge bundles, general graph bundling, visualization of dynamic graphs, diagram visualization, and an expanded treatment of dimensionality reduction techniques). Finally, the appendix has been updated to include several important software systems and libraries. Separately, all chapters have been thoroughly revised to correct errors and improve the exposition, and several new references to relevant literature have been added.

Additionally, the second edition has been complemented by online material, including exercises, datasets, and source code. This material can serve both to practice the techniques described in the book, but also as a basis to construct a practical course in data visualization. Visit the book's website: http://www.cs.rug.nl/svcg/DataVisualizationBook.

# Contents

# Chapter 1

# Introduction

THIS book is targeted at computer-science, mathematics, and engineering-sciences students in their last undergraduate years or early postgraduate phase. A second audience is practitioners in these fields who want to develop their own data-visualization applications but who have not had extensive exposure to computer graphics or visualization lectures. The book strives to strike an effective balance between providing enough technical and algorithmic understanding of the workings and often subtle trade-offs of the most widespread data-visualization algorithms while allowing the reader to quickly and easily assimilate the required information. We strive to present visualization algorithms in a simple-to-complex order that minimizes the time required to get the knowledge needed to proceed with implementation.

Data visualization is an extensive field at the crossroads of mathematics, computer science, cognitive and perception science, and engineering. Covering every discipline that shares principles with visualization, ranging from signal theory to imaging and from computer graphics to statistics, requires in itself at least one separate book. Our goal is to provide a compact introduction to the field that allows readers to learn about visualization techniques. Hence, several more specialized visualization algorithms and techniques have been omitted. On one hand, we have chosen to focus on those techniques and methods that have a broad applicability in visualization applications, occur in most practical problems in various guises, and do not demand a specialized background to be understood. On the other hand, we have also included a number of less mainstream research-grade visualization techniques. With these methods, we aim to give the reader an idea of the large variety of applications of data visualizations, illustrate the range of problems that can be tackled by such methods, and also emphasize the strong connections between visualization and related disciplines such as imaging or computer graphics.

Whenever applicable, existing commonalities of structure, principles, or functionality between the presented visualization methods are emphasized. This should help the reader better understand and remember a number of underlying fundamental principles and design issues that span the visualization field. First, these principles allow one to design and use visualization applications for a problem domain or data type much easier than if one had to learn the required techniques anew. Second, this helps students understand the nature of such cross-domain principles as sampling, interpolation, and reconstruction, and design issues such as optimizing the trade-off between speed, memory consumption, and data representation accuracy. We believe this approach of understanding mathematics and software design by seeing their concrete application in a practical domain may benefit computer-science students, in particular, who have a less extensive mathematical background.

Throughout the book, we illustrate algorithmic and software design issues by providing (pseudo)code fragments written in the C++ programming language. Exercises covering the various topics discussed in the book, including datasets and source code, are also provided as additional online resources. (See Section 1.5.) These can be of help for experimenting with the various techniques and methods discussed in the main text, but also for organizing educational material for courses being taught based on this book. The reader is assumed to have an average understanding of the C++ language, i.e., be familiar with the language syntax and have basic knowledge of data structures and object-oriented programming. Whenever possible, the examples are described in terms of plain structured programming. Object-oriented notation is used only when it simplifies notation and helps understand the described algorithms. No particular software toolkit, library, or system is used to support this description. There is a single exception to this rule: in a few instances, we make use of a small set of concepts present in the OpenGL programming library, such as graphics operations and data types, to illustrate some visualization techniques. OpenGL is one of the best-known and most well-supported graphics libraries in use, has a quite easy learning curve, and provides a compact and concise way to express a wide range of graphical operations. Knowledge of OpenGL is not required to follow the material in this book. However, the provided code fragments should allow and encourage readers who are interested in implementing several of the presented techniques to get a quick start. For both a quick start in programming OpenGL applications as well as an in-depth reference to the library, we strongly recommend the classics, also known as the Red Book [Shreiner et al. 03] and the Blue Book [Shreiner 04].

We have decided to follow a toolkit-independent exposition of visualization principles and techniques for several reasons. First, we believe that understanding the main principles and working of data-visualization algorithms should not be intermixed with the arduous process of learning the inevitably specific interfaces and assumptions of a software toolkit. Second, we do not assume that all readers have the extensive programming knowledge typically required to master the efficient usage of some visualization software toolkits in use nowadays. Finally, different users have different requirements and work in different contexts, so the choice of a specific toolkit would inevitably limit the scope of the presentation.

Last but not least, designing a complete visualization system involves many subtle decisions. When designing a complex, real-world visualization system, such decisions involve many types of constraints, such as performance, platform (in)dependence, available programming languages and styles, user-interface toolkits, input/output data format constraints, integration with third-party code, and more. Although important for the success of a system design, such aspects are not in the realm of data visualization but of software architecture, design, and programming. All in all, we believe that presenting the field of data visualization in a manner as independently as possible from a toolkit choice makes this book accessible to a broader, less specialized audience.

## 1.1 How Visualization Works

The purpose of visualization is to get *insight*, by means of interactive graphics, into various aspects related to some process we are interested in, such as a scientific simulation or some real-world process. There are many definitions of visualization. Following Williams et al., visualization is "a cognitive process performed by humans in forming a mental image of a domain space. In computer and information science it is, more specifically, the visual representation of a domain space using graphics, images, animated sequences, and sound augmentation to present the data, structure, and dynamic behavior of large, complex data sets that represent systems, events, processes, objects, and concepts" [Williams et al. 95].

In most applications, the path from the given process to the final images is quite complicated and involves a series of elaborate data-processing operations. Ultimately, however, the visualization process produces one or several images

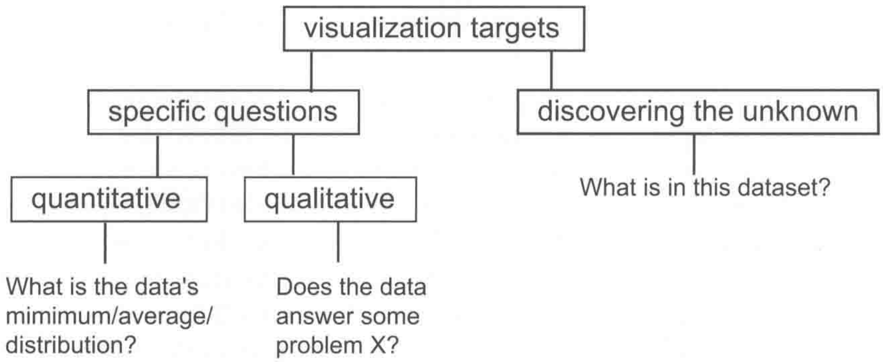



**Figure 1.1.** Types of questions targeted by the visualization process.

that should be able to convey insight into the considered process. In the words of pioneers in the field, visualization is "the use of computers or techniques for comprehending data or to extract knowledge from the results of simulations, computations, or measurements" [McCormick et al. 87].

Visualization and insight. The term "insight" is used very frequently in visualization-related disciplines and texts. However, what exactly does insight mean in this context? Visualization can help obtain several types of insight by answering several types of questions (see Figure 1.1). In the current context, we use the word "insight" to describe two types of information we get from a visualization application:

1. *answers* to concrete questions about a given problem;

2. *facts* about a given problem that we were not aware of.

*Concrete questions:* In the first case, we have some concrete questions about a given phenomenon, process, or dataset.[1] The purpose of visualization in this context is to answer these questions as well, and as quickly, as possible. Such questions can be *quantitative*, e.g., "given a two-dimensional (2D) land map, how high are the highest points?" In this case we are interested in a measurable answer on a given scale of reference, e.g., "the highest peak is 2500 meters above sea level." A sample list of quantitative questions targeted by visualization applications includes:

[1]By *dataset*, we mean a collection of data values that describe some given phenomenon or structure. A more formal definition of a dataset is given in Chapter 3.