Preface

This book is about open source geospatial command line software and has been written with a wide audience in mind that includes graduate students, scientific researchers, geospatial, and remote sensing analysts and practitioners. Its broad objective is to provide a clear and accessible introduction to the range of command line utilities supported by illustrative examples. One of the aims of the book is to demonstrate the power and efficiency of the command line to audiences that have predominantly used software packages driven by graphical user interfaces. It is hoped that this book will provide the novice command line user with sufficient insights to be able to integrate the tools into their day-to-day workflows. Nevertheless, we also anticipate that it will also serve as a reference guide to the more seasoned command line user wishing to review the specifics of a tool or refresh their understanding of the implementation of a tool.

On embarking on this book, we set ourselves an objective of *de-mystifying* the command line and showing its relevance to geospatial analysis. We appreciate that for those that have little or no experience with the command line, it can seem like a very daunting task to learn. In parallel, we acknowledge that there is a steep learning curve associated in getting to grips with it. We would like to emphasize that this early investment in time will soon be repaid many times when your familiarity with the command line increases and you can quickly re-run commands or sets of commands chained together in scripts.

While there are many packages currently available that can be used for geospatial analysis, this work focuses on a specific subset of them. This is due either to their unique functionality that they provide or the fact that they underpin other software packages. For vector and raster data processing, they are:

- GDAL/OGR
- pktools
- Orfeo Toolbox

In less detail, we also cover a number of packages that deal with three-dimensional point clouds obtained from LiDAR instruments. Rather than giving an

exhaustive description, we provide a brief overview with some usage examples of the following packages:

- liblas
- PDAL
- LAStools
- PulseTools
- SPDLib

A key motivating factor behind this undertaking is the capacity of the selected tools to provide highly effective capabilities for processing geospatial data. Another is the fact that while they continue to be developed at a rapid rate, the documentation and related training material associated with them are distributed on disparate websites such that the consolidation of the explanation of a particular tool and its command line options is deemed necessary. The tools that we cover in the book all have the following in common:

- free and open source;
- ability to operate in batch mode (command line processing);
- available on a Linux operating system and other systems.

These three requirements are described in more detail. First, free and open source software has gained in popularity over the last decade. The terms free and open source have been used in different contexts with different meanings. Without entering into the discussion, it is worth recalling the idea of the free software foundation (FSF), in the following terms: "Free as in free speech, not as in free beer." The philosophical view of freedom is ethical and goes beyond the idea of open source, which is rather focused on practical issues, such as making or having powerful, reliable software (Stallman 2009). Nevertheless, the two categories overlap. The unrestricted use of free software is particularly important for capacity building, in particular in developing countries. With the access to remote sensing data archives (Landsat, MODIS, Proba-V, and the upcoming Sentinel missions) becoming more open, both data and processing software are now available to a large community. This provides new opportunities for research in Earth system sciences and its related policy support.

Furthermore, command line processing is a powerful technique that is, in the view of the authors, currently underused. This is particularly true for geospatial data analysis where datasets typically require processing and manipulation. There are three ways in which users can interact with software. The first is a graphical user interface (GUI), which is well known in Windows environments and typically used in most proprietary software. Users interact with the software via menus, dialog boxes, and mouse actions (move, clicks, drag, and drop). In many cases the software combines a development environment with an image viewer in a so-called what-you-see-is-what-you-get (WYSIWYG) application. This allows the user to see and directly interact with the results of the actions performed on the geospatial data. Such applications are intuitive for novice users as they learn the application through explorative use. False user input can be easily avoided by letting the user choose

only from a predefined choice via menu options. However, repetitive tasks require many mouse clicks and are time-consuming as a result. For this reason it is argued that command line processing has a lot to offer both operational and non-operational contexts. For instance, scientists dealing with geospatial data develop prototypes when creating new algorithms. Clearly, it is a welcome feature to have a development environment underpinned by a viewer and a GUI since it allows modifications to parameters to be viewed immediately. However, prototyping often requires testing the effect of a parameter on many test cases (e.g., images). It is also common that the range of settings for a number of parameters need to be optimized. Command line processing is especially useful in these situations where repetitive tasks need to be carried out. They can be automated with scripts, including iterative loops and variables.

On the other side of the spectrum, users can interact with software via an application programming interface (API). This allows users (or programmers) to use predefined functions in their own programs. APIs typically handle lower level tasks such as reading and writing image formats so that programmers can concentrate on higher level (i.e., algorithmic) tasks. The latter is arguably the most flexible and sophisticated way of interacting with software. The analyst of geospatial data has complete control of handling the data by combining the basic instructions provided by the API with self-written code in a computer language of its choice (typically C++, Python, or Java). However, not all scientists or spatial data analysts are computer experts and there exists, therefore, a potentially significant gap between the two ends of the spectrum to interact with geospatial data (GUI and API). This book is intended for geospatial analysts seeking to take advantage of the power of command line processing and APIs. The book provides a number of ready to use tools written by analysts that were confronted with similar problems. As all sources are open, the committed reader can consult the programs and adapt them to his or her own needs, and the book also offers an introduction to GDAL/OGR API programming.

Third, the tools are available on a Linux operating system. While most are available on other operating systems (MS Windows, Mac OS, Unix), the choice of Linux is driven by the focus on command line processing. Linux has inherited a powerful shell scripting language (Bash), making it ideally suited to the tools analyzed in this book. As a free and open source operating system, it not only complies with our second requirement, but it is also consistent with the overarching aim of the book.

This book is structured into three parts covering different types of geospatial analysis. First, it offers an introduction to geospatial data and analysis with an emphasis on Earth observation data. The introduction also serves to outline the software packages and libraries that are dealt with throughout the book. Part I focuses on processing vector and raster data. It first introduces the OGR Simple Features library and its command line tools that can be used to manipulate, query, and analyze vector features and their attributes. It is continued with raster processing using the GDAL command line utilities. Third party command line tools are introduced in Part II. We cover pktools and the Orfeo Toolbox as well as developing

new utilities with the API. Rather than providing a complete overview, the third-party tools are introduced to cover complementary functionality to GDAL/OGR. Laser scanning data are are also covered there, using a variety of open source software packages. Part III provides some practical applications of the tools for real-world examples. We conclude our book with Chap. 19 that offers a somewhat personal perspective on the future of geospatial analysis, Earth observation data, and the emerging tools that will develop to visualize and analyze these data.

On page xxiii, you can find an overview of the utilities covered in this book with a short summary on their use and a reference to the page where they are explained within the book. A list of acronyms and a glossary are provided on pages xxi and 349 respectively. Finally, the Appendices offer supplementary information that can facilitate the installation of software relevant to this book and as well as information on the data and tools used throughout the book.

County Clare, Ireland, August 2014 Daniel McInerney
Balen, Belgium Pieter Kempeneers

http://www.springer.com/978-3-319-01823-2

Open Source Geospatial Tools
Applications in Earth Observation
McInerney, D.; Kempeneers, P.
2015, XXVII, 358 p. 96 illus., 50 illus. in color., Hardcover
ISBN: 978-3-319-01823-2