

SOFTWARE

Open Access



OpenDragon: software and a programmer's toolkit for teaching remote sensing and geoinformatics

Kurt Rudahl*  and Sally E. Goldin

Abstract

Background: Continued progress in geoinformatics research and applications depends on educating new generations of theorists and practitioners. Effective education and training in turn depend on the availability of appropriate tools, including software.

The OpenDragon Project was established in 2004 to help satisfy this requirement. OpenDragon provides a simple yet powerful platform for hands-on lab exercises and student projects. The fact that it is available in multiple languages and can be internationalized without changing any code gives the system a world-wide audience. It is simple, portable, easy to use, and modest in its hardware requirements.

Results and conclusions: OpenDragon can be used as a turnkey platform to teach undergraduates and even secondary school students about geography, agriculture, geology, meteorology, forestry and other natural resource disciplines. OpenDragon is unusual in that it also provides support, through the Programmer's Toolkit, for students and researchers who need to program their own algorithms or who want to understand the internal workings of remote sensing and GIS software.

This paper describes OpenDragon, its history, and its capabilities. It also explains the system architecture and the organization of the Programmer's Toolkit, and includes some examples of how we have used these capabilities to support student research and to teach geoinformatics programming.

Keywords: Remote sensing, Programming, Teaching, Open Source

Background

Continued progress in geoinformatics research and applications depends on educating new generations of theorists and practitioners [1]. There are many open source projects that provide a framework for research in remote sensing and geoinformatics [2, 3] and can be potentially used for educational purposes [4], but none have been specifically engineered for educational purposes. Effective education and training in turn depend on the availability of appropriate tools, including software. The OpenDragon Project was established in 2004 to help satisfy this requirement. Details of the project can be found at <http://www.open-dragon.org>.

The primary software currently offered by the project is OpenDragon, which is derived from the commercial

Dragon/ips® package originally developed by Goldin-Rudahl Systems, Inc. [5]. OpenDragon is simple, portable, easy to use, and modest in its hardware requirements. Furthermore, it is fully internationalized, so that producing versions in languages other than English is a straightforward process of translating a few dozen pages of text. Currently, OpenDragon includes a variety of languages, including Thai, Traditional Chinese and Bahasa Indonesia.

OpenDragon can be used as a turnkey platform to teach undergraduates and even secondary school students [6] about geography, agriculture, geology, meteorology, forestry and other natural resource disciplines. In this role, it supports the education of practitioners and users of geoinformation. Due to its modular design and the developers' professional computing background OpenDragon can be used in conjunction with the Dragon Programmer's Toolkit to provide a tool for teaching about the underlying

* Correspondence: ktr@goldin-rudahl.com
King Mongkut's University of Technology Thonburi, Bangkok, Thailand

software. The combination allows students and researchers to explore new geoinformatics algorithms and data structures.

Figure 1 shows an example secondary school project.

This paper describes OpenDragon, its history and its capabilities. It also explains the system architecture and the organization of the Programmer’s Toolkit, and includes some examples of how we have used these capabilities to support student research and to teach geoinformatics programming.

History of OpenDragon

OpenDragon is based on Dragon/ips, which was the first commercial remote sensing image processing software package developed for off-the-shelf personal computers. The authors began the development of Dragon/ips in the mid-nineteen eighties after spending 2 years teaching remote sensing image processing at the Asian Regional Remote Sensing Training Center (ARRSTC), at the Asian Institute of Technology in Bangkok Thailand. For laboratory exercises, the courses offered at ARRSTC used expensive, proprietary minicomputer- and mainframe-based image processing software which required special graphics display hardware. Twenty to thirty students each term had to share five workstations, which meant that students had limited opportunities for hands-on practice. Recognizing the need for affordable software for remote sensing education, the authors decided to create remote sensing software that could be used with the IBM PC and similar computers.

Dragon/ips was released in 1987 by Goldin-Rudahl Systems, Inc. (GRS), who focused keeping the system cost low and the hardware requirements modest. In 2003, the authors returned to Thailand to take teaching positions at King Mongkut’s University of Technology Thonburi (KMUTT). The authors also convinced the university to

provide 3 years funding for the OpenDragon project, to create and disseminate a free version of the software for educational organizations in Southeast Asia.

After KMUTT funding ended, GRS took over support for OpenDragon, providing web hosting and continued development, and in 2010, GRS made OpenDragon free worldwide. Any user, anywhere in the world, can now freely download the system after completing a simple registration process.

To increase the utility of the software, in 2011 the authors published [7] which is available in ebook and print from Amazon.com. This text explains the basic theory underlying core remote sensing image processing operations and provides extensive hands-on exercises using OpenDragon.

Currently OpenDragon is free (but not open source). The package has been downloaded by users in more than 100 countries. If funding can be found, the intention is to merge the capabilities of Dragon/ips into OpenDragon, create versions for Linux and MacOS/X, and make the resulting combination both free and open source.

Implementation

OpenDragon functional capabilities

The OpenDragon turnkey application provides core remote sensing image processing functionality, as well as some raster GIS operations, in a simple, easy-to-use desktop package. Currently it runs only on Windows (Windows XP through Windows 8.1 have been tested), although all of its modules except for the image display component are platform independent. As discussed below, future plans include migration to Linux and to OS/X, as well as a possible web-based deployment.

OpenDragon’s functional capabilities include:

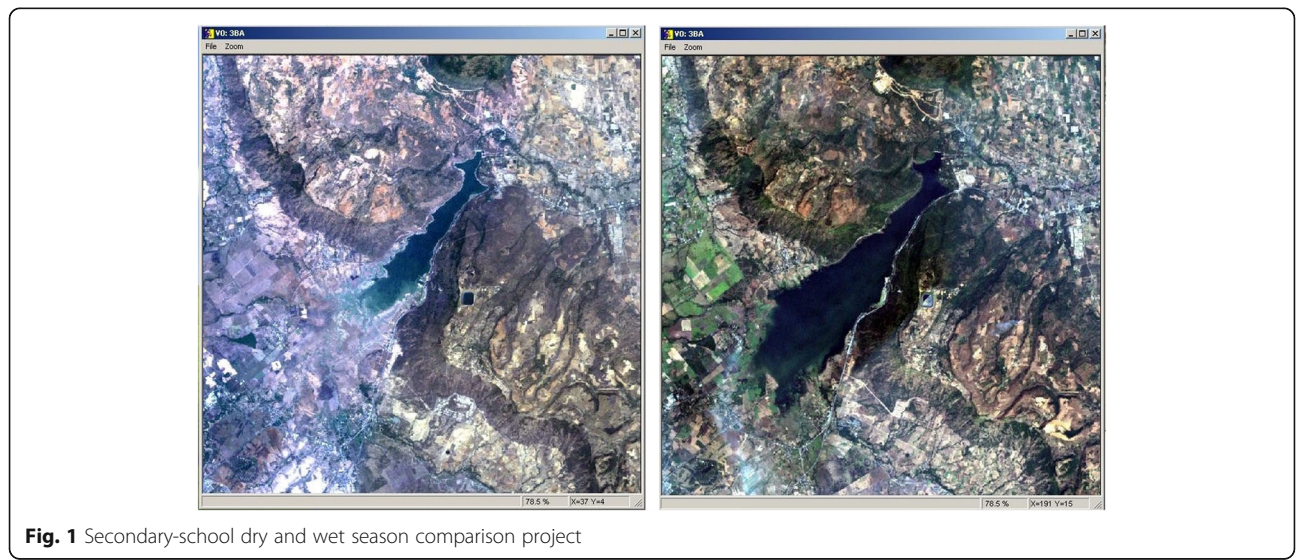


Fig. 1 Secondary-school dry and wet season comparison project

- Multi-image gray scale, pseudocolor and composite display, with Unicode image annotation
- Histogram and scatterplot display, metadata display and editing, interactive image data value exploration (see Fig. 2)
- Image arithmetic, convolution and median filtering, vegetation indices, masking, recoding, principal components
- Interactive training, signature display and editing, supervised and unsupervised classification, classification error analysis
- Geometric correction and image-to-image registration
- Vector feature import, display, capture, and export
- Image feature measurement; area histograms and line profiles; irregular subset extraction
- Slope and aspect calculations; raster buffers; rule-based multi-band combinations
- Context sensitive help and on-line manual.

OpenDragon’s most significant limitation is the fact that display and processing are restricted to images no larger than 1024 × 1024 pixels. This limited image resolution is appropriate for teaching and lab exercises but it may not be adequate for research activities. In this case, students may need to use the full Dragon/ips system (see Future developments, below).

All OpenDragon operations can be executed interactively via the menu-based user interface. In addition, OpenDragon provides a scripting capability which allows instructors or students to automate sequences of frequently executed operations. OpenDragon also includes a variety of stand-alone utility programs and sample imagery from various sources. Additional images are available for use with the exercises in the *Learning Remote Sensing* textbook.

The architecture of OpenDragon

Normally, end users do not need any knowledge about the internals of a software system. However, OpenDragon is used not only to teach about geoinformatics applications but also about geoinformatics programming. For this latter objective, a basic understanding of the software architecture is required.

OpenDragon is a modular system that consists of four basic components, as shown in Fig. 3.

The **User Interface** component, written in Java, provides flexible and convenient access to OpenDragon functionality using familiar graphical conventions. The contents of the User Interface screens are generated at run time, based on a set of XML description files plus language-specific message files. Thus, it is possible to add new user interface screens with very little programming.

The **Image Processing Engine** handles all manipulation of image and vector data. It receives its instructions via command files, and thus can be run in a batch mode without the user interface.

The **Viewport** (of which there are multiple instances) is responsible for image display and real time interaction (e.g. interactive measurement or heads-up digitizing). It provides scrolling, zooming, capturing and printing of images as well supporting image annotation (legends and text).

The **Dragon Executive** maintains application state, coordinates communication between the other components, and supports scripting.

Presently all components except for the Viewport are platform-independent and can run on Windows, Linux and Mac OS/X. Re-implementing the Viewport using platform-independent technologies is the critical need before we can make the project open source.

For research and for studying geoinformatics software development, the most important component is the Image Processing Engine. As shown in Fig. 4, this

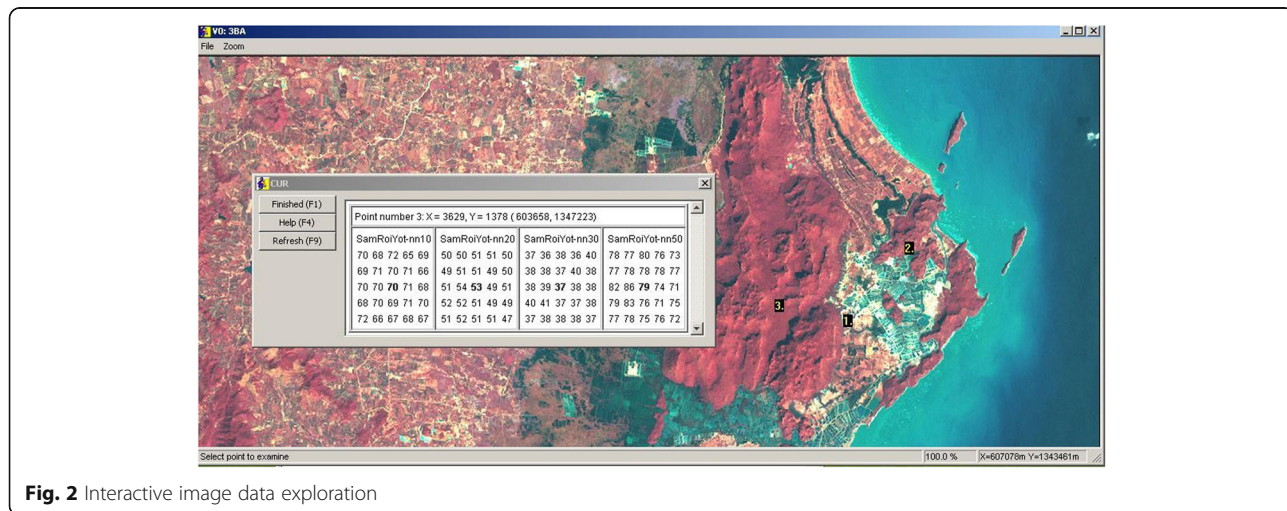
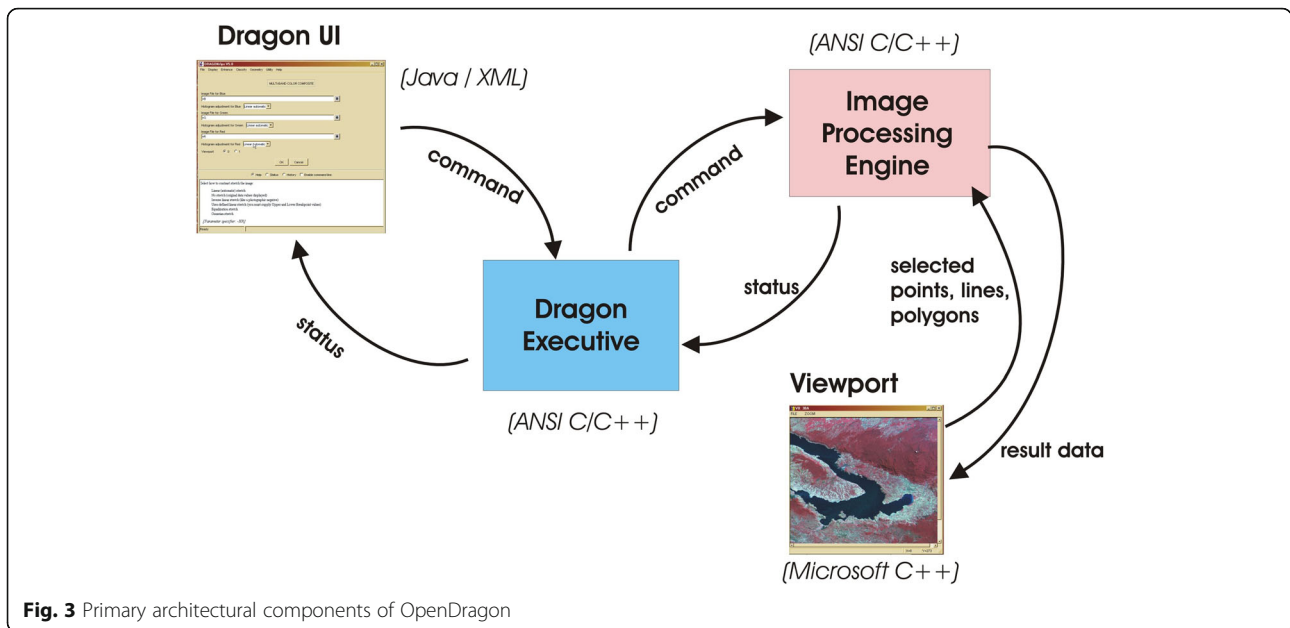


Fig. 2 Interactive image data exploration



component consists of operation-specific application code, functions to handle communication with other components of the system, and a set of core functions for reading, writing, and querying geographic data. The core functions provide access to raster and vector data sets and their metadata, as well as some ability to manipulate other categories of information such as training signatures and image statistics.

Students and researchers need access primarily to the core functions in order to start writing their own geoinformatics modules. The core libraries, a simplified set of header files, HTML documentation, sample programs, and build scripts are packaged together into the OpenDragon Programmer's Toolkit. With the toolkit, students and researchers can write, compile and link code that will

read and write OpenDragon images and vector data, and take advantage of other data management facilities.

Note that at present OpenDragon is not open source. Educational organizations can request source code and the project will evaluate each request individually. However, the project is still in the process of reorganizing, simplifying and rationalizing the source code to make it comprehensible to a non-expert programmer. If the project receives long-term funding to support this activity, OpenDragon will move to a true open source model.

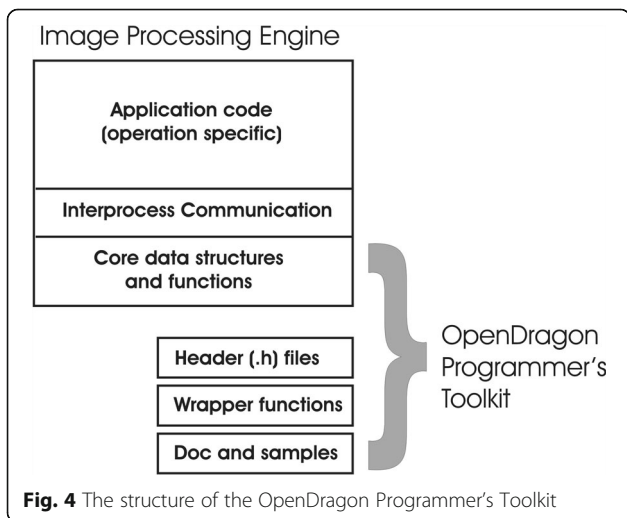
Results

Using the OpenDragon Programmer's toolkit for projects

The Programmer's Toolkit exposes an Application Programming Interface (API) that allows a user with fairly basic C language programming skills to create new modules compatible with OpenDragon's data and file structures. The Toolkit can be used for a variety of purposes:

- To create new data import and output capabilities;
- To implement processing options not currently offered by the turnkey package;
- To implement novel or advanced image processing and vector processing algorithms that can be used in conjunction with the turnkey package.

During our tenure at KMUTT, a number of undergraduate and graduate student projects have used OpenDragon and the Toolkit. The fact that students do not need to develop new data formats and can use the OpenDragon application saves them time and allows them to focus on the novel aspects of their research.



This section summarizes some of the work our students have done with OpenDragon.

Undergraduate projects

Vector GIS using PostgreSQL (2005). Songpon Yookeaw and Nopporn Jinbunluphol created a prototype vector GIS system in order to evaluate the spatial database capabilities of the open source PostgreSQL database system. They used OpenDragon to process raster images to serve as backgrounds for vector display. Vector import and export used the OpenDragon vector file format.

Remote Sensing Image Classification using Neural Network (2005). Pawarat Kitmanomai and Prachya Lalit-norasate created a neural network classifier and compared its performance to the OpenDragon maximum likelihood classification. They used the Toolkit to read and write training and result images and to process signatures.

Generation and Animation of Perspective Views (2006). Korapin Atichaichotikul and Piyana Sukanghong developed an application to create a perspective view of a landscape based on a digital elevation model, draping the model with satellite image information in order to enhance the realism and information content. They used the Toolkit to read and process the DEM and image files.

Automatic Extraction of Road Networks (2006). Sasiwimon Ratchatasomboon 122795031 and Wittaya Unjai developed a system to identify roads in medium resolution imagery, using a combination of low-level (e.g. filtering) and high-level (e.g. context-based rules) approaches. They used OpenDragon and the Toolkit for all input, output and results display.

Image Processing in Hardware (2007). Kittituch Manakul and Surachai Chatchalernmpul developed a co-processor to execute time-consuming components of image processing operations, such as gray-level co-occurrence matrix calculations, in hardware using FPGAs (Field Programmable Gate Arrays). They used OpenDragon and the Toolkit to create a baseline, software-only implementation of their algorithms for comparison purposes.

Interactive Map and Directory for KMUTT Campus (2007). Aunyarath Prasittiarpa and Kanokporn Sudsang swang created a special-purpose GIS application to serve as an English/Thai interactive guide to the KMUTT campus. They used OpenDragon and the Toolkit to create a vector representation of campus roads and building footprints.

Travel-time GIS "Buffer" Operation (2009). Pirakorn Iamcharernying and Pawat Duantsatit developed algorithms to compute buffers that take travel time along road networks into account. They used OpenDragon for image input and display.

Viewshed Analysis (2016). For her 2-month internship project at KMUTT, Camille Aidat from ECE Paris used

the Toolkit to create a module for calculating what locations are visible from a selected viewpoint.

Morphological Operations (2016). For his 2-month internship project, Guilhem de la Ruelle from ECE Paris used the Toolkit to create a module that implements the morphological operations of erosion, dilation, opening and closing, within the OpenDragon framework.

Masters theses

Improving the Estimated Area of Rice Cultivation by Classification Techniques [8]. Ithiphol Ekahitanonda explored the efficacy of multi-temporal analysis, genetic algorithms, and fuzzy decision making approaches as ways to resolve ambiguities in identifying rice from Landsat ETM imagery. He used the OpenDragon Toolkit extensively, developing a wide range of new modules and utilities.

Change Detection in Natural Disasters: the Tsunami Case [9]. Supanee Tanathong implemented an object-based strategy for characterizing landscape change after natural disasters such as the December 2004 tsunami. She used the OpenDragon Toolkit to experiment with segmentation using texture features.

Texture Based Classification of High Resolution Remotely Sensed Imagery [10]. Decky Aspandi Latif explored the utility of several novel, computationally efficient texture metrics in classifying Quickbird imagery. He used the Programmer's Toolkit extensively so that he could write code for his own algorithms and then use standard Dragon functions, invoked from inside his programs, for image display, subsetting, error analysis and so on.

All of the above projects would have been far more difficult if students had been required to develop all software capabilities "from scratch". Although our students are studying Computer Engineering, they are not necessarily expert programmers. Nevertheless, they were easily able to apply the Toolkit to their projects.

Prior to the availability of OpenDragon, projects and theses related to remote sensing and GIS were rare in our department. The OpenDragon Project has improved student research quality, and has provided new visibility for the field of geoinformatics within our primarily engineering-oriented university.

Using OpenDragon to teach geoinformatics software development

In addition to encouraging students to use the Toolkit for research, we have also used OpenDragon to teach two different courses related to geoinformatics software development.

The first is a course for seniors and graduate students within our own Department of Computer Engineering, entitled "Algorithms and Architectures for Geoinformatics". This course introduces students to concepts of representing

spatial information in computers, including rasters, non-topological and topological vectors, quadtrees, sparse data arrays, TINs, etc. We use actual source code from OpenDragon to illustrate concepts of pixel-, neighborhood- and region-based algorithms. Students complete group projects during the semester, often using the OpenDragon Toolkit as a foundation. Projects have included slope and aspect calculation, region-growing, buffer generation, edge extraction, optimal path finding and texture-based segmentation. Since all students had substantial backgrounds in programming, we focused on issues such as efficiency in algorithm and data structure design, and software engineering principles such as modularity, separation of concerns, and consistent style.

The second course, “Computer Programming for Geoinformatics”, was created to introduce masters students from the Chiang Mai University Department of Geography to the basics of writing software in the geoinformatics domain. The course was later repeated for masters and PhD students in the Burapha University Department of Geoinformatics (Bang Saen, Chonburi, Thailand). The goal of this course is provide the students with sufficient background to understand and evaluate commercial or third-party software, write small programs to help integrate existing packages, and create scripts to automate work flows. In both universities, the students had little or no experience with programming. Thus the course began with the elements of programming: variables, conditional statements, loops, functions and so on.

About half of the time in this second course was spent in laboratory exercises. The labs relied heavily on the OpenDragon Toolkit for raster work, most of which was done using the C language. For labs illustrating vector concepts, we used the OpenDragon vector format in some simple Java applications.

In both the computer engineering and the geography courses, OpenDragon served as a core resource. A student studying software development will learn a great deal more from viewing real-world code than from reading a text book or listening to lectures. Writing code provides an even deeper understanding of the issues involved in software development. Our experience in Chiang Mai indicates that even non-technical people can benefit from well-written, well-organized code examples and exercises.

Most students in geography, planning, forestry, oceanography, and other natural resource related disciplines have very little exposure to programming concepts. Seeing the step-by-step operation of remote sensing or GIS processes, expressed succinctly in the form of a program, can help make the processing logic more concrete and easier to understand. In addition, of course, professionals with “spatial intelligence” are needed to work on future geoinformatics tools. The remote sensing and GIS developers of the future are at least as likely to come from the “softer” disciplines of geographic science as they are from the

engineering side. Tools like OpenDragon and its libraries facilitate professional development by letting students “get their hands dirty” working with real code.

Discussion

Future developments

As noted earlier, OpenDragon became free and universally available in 2010. Since then, the authors have been seeking sponsors who will support the continued availability and expansion of OpenDragon’s capabilities, in order to convert the package to a true open source distribution model and provide the greatest benefit for the widest audience. However, this cannot be done without financial backing. Though open source software itself is free to the users, open source projects are not sustainable without some external funds. The most successful open source efforts (for example, Linux, Apache, Perl) all have corporate support from companies such as IBM, Novell, and O’Reilly Publishing.

There are also many new features that should be incorporated into OpenDragon. In the area of the Programmer’s Toolkit, there should to be a mechanism allowing users to integrate their own geoprocessing modules into the OpenDragon user interface. A prototype of this capability already exists but needs to be made accessible to users.

Conclusion

OpenDragon provides a simple yet powerful platform for hands-on lab exercises and student projects. The fact that it is available in multiple languages and can be internationalized without changing any code gives the system a world-wide audience.

Although other software packages are available for teaching remote sensing and GIS at an introductory level, most if not all focus on application-oriented users. OpenDragon is unusual in that it also provides support for students and researchers who need to program their own algorithms or who want to understand the internal workings of remote sensing and GIS software.

The OpenDragon Programmer’s Toolkit provides a level of extensibility that is rare in the world of geoinformatics software. It offers a testbed for research as well as a framework for hands-on class projects. By revealing the inner workings of geoinformatics software to the students of today, OpenDragon helps build the skills of the people who will develop the spatial analysis software of tomorrow.

Availability and requirements

Project name: OpenDragon

Project home page: <http://www.open-dragon.org>

Operating system(s): currently Windows; eventually platform independent

Programming language: C, C++, Java, and others

Other requirements: minimum screen size of 1024 × 768 (so-called VGA size)

License: currently must register to download; eventually FreeBSD or similar

Restrictions to use by non-academics: no

Availability of data and materials

N/A

Authors' contributions

Written and edited by KR and SG. Illustrations selected and/or generated by KR and SG. Submission process handled by KR. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Received: 20 November 2016 Accepted: 22 February 2017

Published online: 13 March 2017

References

1. Konecny M, Nyapola H. Capacity Building for Geo-Information Development, E/CONF.97/6/IP.2. United Nations Economic and Social Council. Proceedings of the Seventeenth United Nations Regional Cartographic Conference for Asia and the Pacific. 2006.
2. Piragnolo M, Masiero A, Fissore F, Pirotti F. Solar irradiance modelling with NASA WW GIS environment. *ISPRS Int J Geo-Information*. 2015;4:711–24. doi:10.3390/ijgi4020711.
3. Pirotti F, Guarneri A, Vettore A. Collaborative Web-GIS design: a case study for road risk analysis and monitoring. *Trans GIS*. 2011;15:213–26. doi:10.1111/j.1467-9671.2011.01248.x.
4. Pirotti F, Brovelli MA, Prestifilippo G, et al. An open source virtual globe rendering engine for 3D applications: NASA World Wind. *Open Geospatial Data, Softw Stand*; (in Print).
5. Goldin SE, Rudahl KT. Dragon and Phoenix: low-cost image processing for developing countries. Proceedings of the South East Asian Regional Computer Conference. Bangkok: Computer Association of Thailand. 1986. 41–42.
6. Goldin SE, Rudahl KT. Report on the High School Workshop on Remote Sensing for Drought Assessment. Bangkok: King Mongkut's University of Technology Thonburi; 2005.
7. Rudahl KT, Goldin SE. Learning Remote Sensing Using Dragon/ips® and OpenDragon™. Hadley: Global Software Institute; 2011. ISBN 9781453862940.
8. Ekahitanonda I, Goldin S, Rudahl KT. Improving rice classification using multi-temporal data with fuzzy logic and genetic algorithms. Proceedings of the 28th Asian conference on remote sensing. 2007.
9. Tanathong S. Object-based change detection: the tsunami disaster case. Proceedings of the 28th Asian conference on remote sensing. 2007.
10. Latif DA, Goldin S, Sumaryono. Texture based classification of high resolution remotely sensed imagery using Weber Local Descriptor. Proceedings of 10th International Conference on Asia GIS. 2014.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
