



Orfeo ToolBox: open source processing of remote sensing images

Manuel Grizonnet, Julien Michel, Victor Poughon, Jordi Inglada, Mickaël Savinaud, Rémi Cresson

► To cite this version:

Manuel Grizonnet, Julien Michel, Victor Poughon, Jordi Inglada, Mickaël Savinaud, et al.. Orfeo ToolBox: open source processing of remote sensing images. Open Geospatial Data, Software and Standards, 2017, 2 (15), pp.8. 10.1186/s40965-017-0031-6 . hal-01774927

HAL Id: hal-01774927

<https://hal.science/hal-01774927>

Submitted on 24 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

SOFTWARE

Open Access



Orfeo ToolBox: open source processing of remote sensing images

Manuel Grizonnet^{1*} , Julien Michel¹, Victor Poughon¹, Jordi Inglada^{1,2}, Mickaël Savinaud³ and Rémi Cresson⁴

Abstract

Orfeo ToolBox is an open-source project for state-of-the-art remote sensing, including a fast image viewer, applications callable from command-line, Python or QGIS, and a powerful C++ API. This article is an introduction to the Orfeo ToolBox's flagship features from the point of view of the two communities it brings together: remote sensing and software engineering.

Keywords: Remote sensing, Image processing, Open source

Orfeo ToolBox (OTB) [1] is an open-source project for processing satellite imagery. Built on the shoulders of the open-source geospatial community, it can process high resolution optical, multispectral, hyperspectral and radar images up to the terabyte scale. A wide variety of applications are available that include: ortho-rectification or pan-sharpening, supervised or unsupervised classification, feature extraction, SAR processing.

All of OTB algorithms are accessible from Monteverdi, Quantum GIS [2], Python, the command-line or C++. Monteverdi is an easy to use visualization tool with an emphasis on hardware accelerated rendering for high resolution imagery. With it, end-users can visualize huge raw imagery products and access all of the applications in the toolbox. From resource limited laptops to high performance MPI [3] clusters, OTB is available on Windows, Linux and Mac OS X. OTB is community driven, extensible and has a comprehensive documentation.

OTB in a nutshell

With more than 10 years of development and 28 major releases, Orfeo ToolBox has evolved from a very-high-resolution specific toolbox driven by the CNES Pleiades mission to a multi-sensor, multi-purpose remote sensing toolbox.

Thanks to its modular architecture, OTB allows fast prototyping and covers the full spectrum of algorithms for remote sensing image processing from pre-processing to advanced feature extractions methods allowing one to go from raw data to value added products.

Here is an incomplete list of OTB capabilities:

- Read, write, convert, extract parts of remote sensing data,
- Pre-processing like ortho-rectification, radiometric calibration and pan-sharpening,
- Common image processing tasks (thresholding, Fourier or wavelets transform, etc.),
- Extract features (radiometric indices, textures, shapes, etc.),
- Morphological operators,
- Segment images and vectorize segmentation results (at image scale),
- Classify images in a supervised or unsupervised way,
- Object-based image analysis,
- Export results in Geographic Information System and pretty print for publishing.

Main characteristics

The core OTB library is written in C++ and based on the Insight Toolkit (ITK) [4]. ITK is an open-source software toolkit for medical imaging, including registration and segmentation.

As ITK, OTB implementation style is referred to as generic programming (C++ templates). This means that the code is highly efficient, and that many software

*Correspondence: manuel.grizonnet@cnes.fr

¹CNES, 18 avenue Edouard Belin, Cedex 09, 31401 Toulouse, France
Full list of author information is available at the end of the article

problems are discovered at compile-time, rather than at run-time during program execution. It also enables to work with different types of images (different number of bands, dimensions, pixel type, etc.).

OTB algorithms covers a large number of features needed to process remote sensing images from basic pre-processing to high performance analysis. OTB's mission statement is to provide a free software end-to-end solution for the Earth Observation image information extraction pipeline based on a generic, high performance C++ library. On top of it, applications and processing chains can be built to fulfill the needs of users from ground segment processing chains to single user desktop applications.

Code reuse

Remote sensing image processing often leads to the combination of specialized methods available in dedicated software. The idea in OTB is to provide a common interface to these software libraries.

Therefore, OTB depends on Geospatial Data Abstraction Library (GDAL) [5] for raster and vector formats support; OSSIM for geometric sensor modeling; libSVM [6], OpenCV [7] and Shark [8] for machine learning; and MuParser [9] for instance for high performance mathematical expression parsing.

Furthermore, OTB also tries to keep track of up-to-date research about the latest developments and integrate reference implementation of algorithms after their publication. For instance Morphological profiles, dimensionality reduction methods (PCA, NA-PCA, ICA, etc.), Line Segment Detector (LSD) [10] fast implementation of Haralick textures and SURF keypoints matching. The results of the LSD algorithm on Pleiades image is illustrated in Fig. 1.

To maximize user outreach, and to ensure that OTB is both easy to use and easy to install, the team behind OTB maintains and distributes standalone binary packages for all major platforms: Windows, Linux and Mac OS X.

Community

OTB was created from its inception as a collaborative, community effort. As many open source projects there are number of ways to participate which do not all require programming capabilities: documentation, bug reports and feature requests are all very valuable.

The documentation for instance has always been an important way for the project to gather community support of an open source software. Furthermore, getting documentation is only a part of the problem, what is equally important is the type of documentation. OTB provides different typed of documentation depending on users needs:

- The Software Guide [11] is a comprehensive guide which comprises about 800 pages, detailing the steps

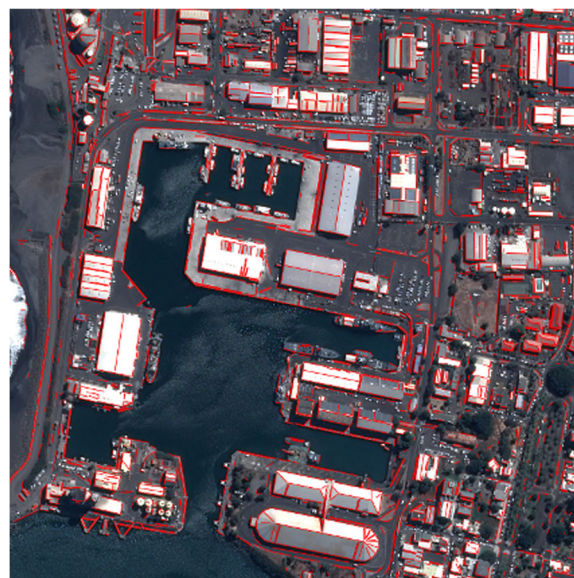


Fig. 1 Results of the Line Segment Detector (LSD) applied on a Pleiades pan-sharpened image

to install OTB and describing many of the features of the library. Most of the classes available are heavily illustrated with results from real remote sensing processing,

- The OTB Cookbook [12]: a guide for OTB-Applications and Monteverdi dedicated for non-developers. It is composed of a brief tour of OTB-Applications and Monteverdi, followed by a set of recipes to perform remote sensing tasks with both tools,
- Tutorials: the OTB website gather also free training materials shared by the community [13].

In early 2015, OTB set up an official Project Steering committee (PSC) to provide high level guidance and coordination [14]. The PSC provides a central point of contact for the project and arbitrates disputes. It is also a stable base of institutional knowledge to the project and tries its best to involve more developers and guarantees that OTB remains open and company neutral.

The OTB PSC was inspired by similar structures that exist in other geospatial software projects like GRASS GIS [15] or QGIS.

Finally, OTB is currently in the incubation stage of being part of the OSGeo foundation [16]. Within the Orfeo ToolBox community we act respectfully toward others in line with the OSGeo Code of Conduct [17] and we hope to be able to complete the incubation process in 2017.

From desktop to high performance computing

Orfeo ToolBox is designed to accommodate both interactive computing on users' desktops, and terabyte scale processing on many core architectures. This is achieved with a modular software architecture: the so-called OTB sandwich illustrates in Fig. 2.

At the core of OTB is the C++ API, which implements the pipeline. This core model of processing supports multi-threading, streaming and message passing. Thus all applications and filters can process images and scale to the available memory and CPU resources. With additional features like in memory application chaining (to minimize disk I/O), porting code from a development machine and scaling up to a high performance cluster is often trivial.

Monteverdi

Orfeo ToolBox ships with Monteverdi, a lightweight image rendering and processing tool written in Qt and OpenGL. Monteverdi makes use of the Ice rendering engine [18], also available in Orfeo ToolBox which offers:

- Smooth navigation in very large datasets using GDAL overviews capabilities,
- Reactive local and global rendering tools such as local contrast enhancement or color-mapping, based on advanced OpenGL features such as floating point textures and OpenGL Shading Language (this work was inspired by the authors of pvflip [19]),
- Multiple image display with on-the-fly rough registration of any image whose coordinate reference system is understood by Orfeo ToolBox, which include both ground projected and sensor geometry images.

Monteverdi is a day-to-day tool for fast visualization of processing results, which can display images in sensor geometry. Monteverdi also facilitates processing using the applications, which will be the focus of the next section. It does not intend to replace GIS software such as QGIS which are more suitable to edit, display and relate different sources of geographic information (both raster or vector).

Figure 3 shows an example of the kind of rendering you can get by using Monteverdi.

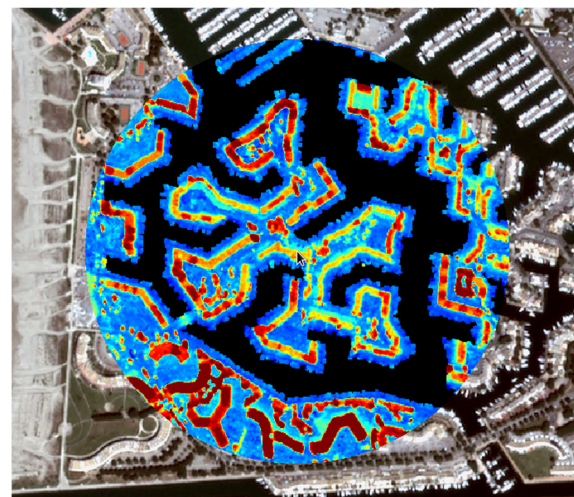


Fig. 3 Monteverdi software visualisation panel displaying ortho-image with local translucency and underlying digital surface model with jet color-mapping

OTB applications

The cross platform, ready-to-use OTB package ships with more than 90 applications. Applications expose existing processing functions from the underlying C++ library, or compose them into high level pipelines. This results in straightforward interfaces for many complex remote sensing algorithms.

For example, the *Orthorectification* application includes a set of complex pipelines with gives access to a number of rectification functions and parameters which can be easily configured by the user: output cartographic projection, external Digital Elevation Model file, interpolation mode, etc. Other flagship OTB applications follow a similar approach and are designed to work well together.

For instance, a recent addition is a new framework for pixel based classification. This new set of applications clarifies the different steps required to train and fit a classification model. In short they are: select samples in a reference image to be used for learning (and how to select them), extract pixel values from the image, and train

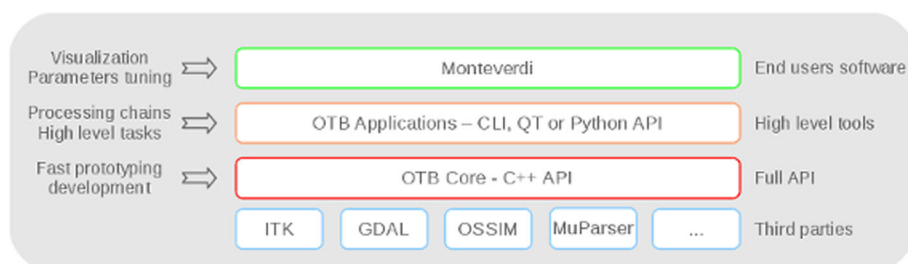


Fig. 2 The difference software layers in Orfeo ToolBox: dependencies, C++ API, applications, and GUI. a.k.a. the "OTB sandwich"

a supervised classifier (Support Vector Machines, Random forests, etc.). This way, control is given to the user over each critical step in the design and tuning of a classification pipeline. This is achieved without an excessive increase in complexity, to maintain both rapid prototyping capability and high performance.

Another important design characteristic of applications consists in the independency of their code and user interface. This is achieved with the so-called *Application Framework*. The framework allows multiple interfaces to be provided for each application without code duplication.

Today, OTB applications are available through the following interfaces:

- The command-line,
- A GUI interface based on Qt,
- Python for high level programming and connection with NumPy array,
- Monteverdi for interactive processing of viewed images,
- QGIS via the Processing plugin,
- Zoo-Project [20] through a Web Processing Service (WPS).

Here is one example of how to use Python to run the Smoothing application, changing the algorithm at each iteration:

Additionally, the OTB applications are available in QGIS via a Python plugin which is built on top of the QGIS processing framework [21] making spatial analysis tasks more productive and easy to accomplish.

Two additional features work together to decrease friction between the multiple interfaces: saving application parameters to XML files, and automatic conversion of GUI parameters to Bash for easy copy-pasting.

Scaling up

The same Orfeo ToolBox applications that are available for desktop remote sensing will scale up seamlessly to process larger datasets. Whenever possible, algorithms implemented in Orfeo ToolBox are performing piecewise processing, which means that if the data are larger, they will take longer to process, with a constant memory budget. This piecewise processing is a key aspect of the ITK processing pipeline on which Orfeo ToolBox strongly relies (piecewise processing is sometimes called streaming in the ITK world). This ability is retained even when combining several algorithms in the pipeline. When there is no obvious way of computing some algorithms piecewise, Orfeo ToolBox also offers adapted versions of these methods, such as for MeanShift segmentation [22] or large scale region growing segmentation [23] which ensures

```
# Example on the use of the Smoothing application
#

# We will use sys.argv to retrieve arguments from the command line.
# Here, the script will accept an image file as first argument,
# and the basename of the output files, without extension.
from sys import argv

# The python module providing access to OTB applications is otbApplication
import otbApplication

# otbApplication.Registry can tell you what application are available
print "Available applications : "
print str( otbApplication.Registry.GetAvailableApplications() )

# Let's create the application with codename "Smoothing"
app = otbApplication.Registry.CreateApplication("Smoothing")

# We print the keys of all its parameter
print app.GetParametersKeys()

# First, we set the input image filename
app.SetParameterString("in", argv[1])

# The smoothing algorithm can be set with the "type" parameter key
# and can take 3 values : 'mean', 'gaussian', 'anidif'
for type in ['mean', 'gaussian', 'anidif']:

    print "Running with " + type + " smoothing type"

# Here we configure the smoothing algorithm
app.SetParameterString("type", type)

# Set the output filename, using the algorithm to differentiate the outputs
app.SetParameterString("out", argv[2] + type + ".tif")

# This will execute the application and save the output file
app.ExecuteAndWriteOutput()
```

stable results with piecewise or tile-wise computation processing illustrated in Fig. 4.

However, being able to accommodate a memory budget regardless of the size of the data is only a part of the problem. Users usually also want to take advantage of modern CPU architectures as well as High Performance Computing infrastructures (HPC), and Orfeo ToolBox will also do that for them. Whenever possible, algorithms implementation are threaded, mostly using OpenThreads [24], even if specific parts of the code will also use OpenMP [25] directives. Orfeo ToolBox will seamlessly use all cores of the CPU. But as the dataset gets bigger, applications generally moves to an HPC architecture sharing many similar nodes with a shared high-bandwidth storage, and Orfeo ToolBox can also do that. It allows for MPI [3] parallel processing, meaning that the whole pipeline will be replicated across nodes which will produce a piece of the resulting image [26]. An example of use of this capability is the pan-sharpening of a whole Pleiades image (1.6 gigapixels). Spread across 560 mono-threaded nodes, the processing time is cut down to 4.3 min.

One last capability that is very useful to scale up to your data is the inner pipeline connection between applications. Building a processing chain from several applications is very convenient, and will scale-up nicely. But intermediate data must be written to disk and read again by next application. This results in an unnecessary overhead of I/O time if those intermediate results are not meant to be kept. In this case, Orfeo ToolBox allows to connect the inner pipeline between applications, so that piece-wise processing is enabled throughout the chain of applications.

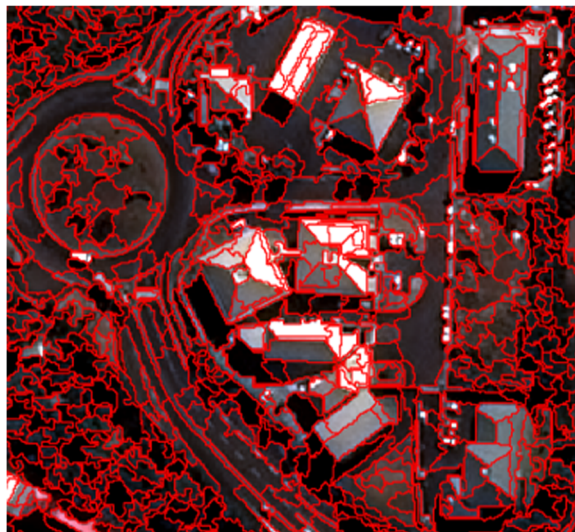


Fig. 4 Result of large scale region-merging segmentation performs on very high resolution optical image at full resolution

Success stories

Beyond the integration of OTB in third party tools such as OSGeo4W, QGIS and Zoo, the library and the applications have been also transferred successfully from research activities to operational environments for mass production of Earth Observation derived products.

ESA Sentinel-2 ground segment

Sentinel-2 mission offers a systematic global coverage with a high spatial resolution, a high revisit and a wide range of spectral bands from visible to short wave infrared. Therefore, the flow of data to process is quite huge (1.2 TBytes of compressed raw data per day) and requires to design an operational Payload Data Ground Segment (PDGS) with efficient image processing capabilities to produce end users Level 1C products (orthorectified data at Top of Atmosphere level) in near real time. In order to develop the image processing module of the PDGS, European Space Agency (ESA) has selected OTB library as main component of the Instrument Processing Facility (IPF) module. More precisely, OTB filters have been used for all radiometric corrections (denoising, defective pixel detection and correction, quality mask and TOA conversion) and resampling operations. The integration of the OTB in this operational framework has been successful both in terms of quality and performance. Since July 2016, every Sentinel 2 data available has been processed by OTB filters. Moreover the OTB library is used in the Mission Performance Center (MPC) for some Calibration and Validation activities.

MACCS

Optical remote sensing from space in the reflective range of the optical domain is a powerful tool for studying the state and the evolution of land surfaces. However, optical observations from space are significantly disturbed by the atmosphere: clouds, gas molecules and aerosols scatter and absorb the light emitted by the Sun or reflected by the Earth's surface. As a result, the operational processing of remote sensing image time series requires preliminary correction steps, such as the detection of clouds and the correction for atmospheric effects. These tasks are particularly difficult above land surfaces, because of two main issues: the identification of cloud-free pixels and, then, the separation of surface and atmospheric effects. The cloud detection problem has already been addressed elsewhere [27] and is not within the scope of this article.

Regarding atmospheric correction, two effects must be taken into account:

- The absorption by atmospheric gases (especially water vapor, ozone, oxygen and carbon dioxide): Absorption has a predominant effect within specific absorption bands, but the spectral bands for land

surface observations are usually designed to avoid strong absorption lines. In these bands, gaseous absorption can be accurately corrected using meteorological analyses and simple analytic models, such as the Simplified Model for Atmospheric Correction (SMAC) [28] or 6S radiative transfer model [29],

- The scattering by air molecules and aerosols: scattering in the atmosphere is very accurately modeled and can be adequately accounted for, provided the composition of the atmosphere is sufficiently well known. This is the case for the air molecules, but the difficulty lies in the knowledge of the aerosol properties, which are very variable, both in location and time.

With good knowledge of the aerosol optical thickness (AOT) and of the aerosol model, and using radiative transfer codes, one can correct for the aerosol effects and convert the satellite top-of-atmosphere (TOA) reflectances into surface reflectances [30].

Multi-temporal Atmospheric Correction and Cloud Screening (MACCS) is a processing chain [31] which implements all these steps. It was designed at CNES and the operational system was entirely developed based on the OTB.

MACCS is used in the The French Theia Land Data Centre (THEIA) [32] to produce and distribute in near-real time the Sentinel-2 data acquired on an area of $5Mkm^2$. The processing starts with France and is progressively extended to the other selected regions at the end of 2016 and beginning of 2017.

Theia OSO land cover product

A detailed and accurate knowledge of the land cover is crucial for many scientific and operational applications, and as such, it has been identified as an Essential Climate Variable [33]. This accurate knowledge needs a frequent update of the information.

In addition to the distribution of surface reflectance products, the French Theia Land Data Centre has also set up a Scientific Expertise Centre whose aim is to implement an operational fully automatic land cover map production system using mostly Sentinel-2 image time series. The product will be updated once a year and will contain 20 thematic classes mapped at 10 meter resolution.

The open source system [34] based on OTB developed in the Centre d'Etudes Spatiales de la Biosphere lab (CES-BIO) allows to produce these types of maps in a consistent and reproducible way. Experimental and operational classifications are two different approaches for large area land cover mapping and monitoring [35, 36]. The former concentrates on the development and performance testing of novel algorithms and models, the latter focuses on the

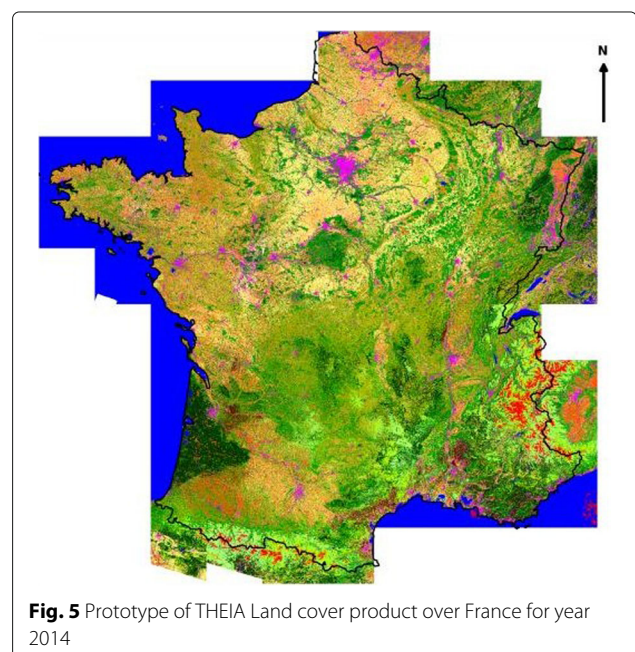
development and delivery of reliable data products within a pre-defined time schedule [37]. This system aims at filling the gap between these two approaches by assessing the performances of a novel strategy in the context of an operational map production system.

The land cover map for the year 2014 over France is illustrated in Fig. 5.

The procedure aims to be portable to other regions of the globe and to allow evolution in terms of nomenclature and update frequency. To fulfill these goals, all the steps of the methodology are independent of the mapping nomenclature or the landscape characteristics. All these characteristics can nevertheless be taken into account by the methodology through the input data (Earth observation, reference data, etc.), but no modification of the workflow is needed for that.

Geoinformation for sustainable development (GEOSUD)

Spatial information of ecosystems, agricultural systems and territories is crucial for environmental and agronomic research, as well as public policies. GEOSUD aims to transfer methods for environment and territories management, and ease the access of spatial information to scientific community and public actors [38]. Currently, this access is guaranteed by a geospatial data infrastructure delivering very high resolution remote sensing products (RapidEye, Pléiades, Spot 6 and 7). It is achieved through the use of Open Geospatial Consortium (OGC) Web Services for data access, visualization and catalog. The incoming extension of the infrastructure implementing the OGC Web Processing Service (WPS) standard



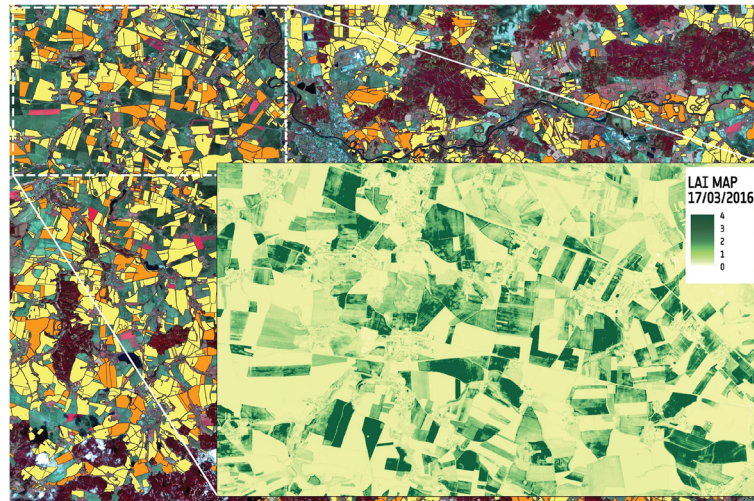


Fig. 6 Extract of a Crop Type map and LAI map produced by Sentinel-2 for Agriculture project over Czech Republic in March 2016 (in orange winter rapeseed, in yellow winter cereals and in Grey fodder crops). The full map can be found at [39]

enables the interactive on-demand remote processing of geospatial data. For this purpose, the Orfeo ToolBox has been deployed on cluster computing architectures, allowing the speedup of data-intensive processes. The large number of available OTB applications fulfills a wide range of needs regardless the users expertise.

Sentinel-2 for agriculture

Developing better agricultural monitoring capabilities based on Earth Observation data is critical for strengthening food production information and market transparency. As previously described, the Sentinel-2 mission has the optimal capacity for regional to global agriculture monitoring. In this context, the European Space Agency launched in 2014 the “Sentinel-2 for Agriculture” [39] project, which aims to prepare the exploitation of Sentinel-2 data for agriculture monitoring through the development of open source processing chains for relevant products. These processors are based on OTB applications framework and library to efficiently generate Level 2A data from Sentinel-2 (and LandSat 8):

- Cloud-free Reflectance Composite,
- Vegetation Status Indicators (LAI, NDVI and phenology metrics),
- Dynamic Cropland mask,
- Crop Type map.

at regional or national scale (cf. Fig. 6).

As OTB application, the Sentinel-2 for Agriculture processors can be interfaced with other standard open-source processing frameworks such as the Sentinel Application Platform (SNAP) [40].

Conclusion

In ten years of its life, Orfeo ToolBox has evolved from a C++ only, VHR specific library to a versatile toolkit addressing most of remote sensing imagery needs. Orfeo ToolBox is now part of several large, operational projects while still being a tool of choice for the day-to-day image processing in labs and a great environment to write new algorithms and methods. Scalability, versatility and openness are the key assets of Orfeo ToolBox.

The software is constantly evolving to enhance performances, to offer more state-of-the-art algorithms and to provide the best interoperability between them. Orfeo ToolBox roadmap is user-driven, and its young open project steering committee ensures that everyone willing to get involved will receive equal attention. In the short to medium term, next OTB versions will include a better integration of non supervised classification algorithms in the applications framework as well as improvements of the integration of applications in Quantum GIS. Finally, the project will continue to try to foster more contributors. With around ten active remote modules already contributed, the project is gaining momentum.

Acknowledgments

The authors would like to thank all the people who contribute in some way to the OTB development, authors of bug reports, of bug fixes and enhancements as well as people who have been working on coordination, documentation and testing of the library and who supported the project over the years.

Authors' contributions

MG, JM, VP, JI, MS and RC are all members of the OTB development team and wrote the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹CNES, 18 avenue Edouard Belin, Cedex 09, 31401 Toulouse, France. ²CESBIO, 13 avenue du Colonel Roche, 31401 Toulouse, France. ³CS Communication & Systèmes, 5 rue Brindejonc des Moulinais, 31500 Toulouse, France. ⁴IRSTEA, Maison de la Télédétection, 500 rue Jean-François Breton, 34090 Montpellier, France.

Received: 19 December 2016 Accepted: 6 June 2017

Published online: 29 June 2017

References

- ORFEO ToolBox - Open Source Processing of Remote Sensing Images. <https://www.orfeo-toolbox.org/>. Accessed June 2017.
- QGIS Development Team. QGIS Geographic Information System. Open Source Geospatial Foundation, 2009. Open Source Geospatial Foundation. <http://qgis.osgeo.org>. Accessed June 2017.
- Forum MP. MPI: A message-passing interface standard. Knoxville: Technical report; 1994.
- Ibanez L, Schroeder W, Ng L, Cates J. The ITK Software Guide, 1st edn. Kitware, Inc., 2003. Kitware, Inc.. ISBN 1-930934-10-6. <http://www.itk.org/ItkSoftwareGuide.pdf>. Accessed June 2017.
- GDAL Development Team. GDAL - Geospatial Data Abstraction Library, Version X.x.x. Open Source Geospatial Foundation, (201x). Open Source Geospatial Foundation. <http://www.gdal.org/>. Accessed June 2017.
- Chang CC, Lin CJ. LIBSVM: a Library for Support Vector Machines. 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>. Accessed June 2017.
- Itseez. Open Source Computer Vision Library. 2015. <https://github.com/itseez/opencv>.
- Igel C, Heidrich-Meisner V, Glasmachers T, Shark J. Mach Learn Res. 2008;9:993–6.
- Muparser - Fast Math Parser Library. <http://beltoforion.de/article.php?a=muparser>. Accessed June 2017.
- Grompone von Gioi R, Jakubowicz J, Morel JM, Randall G. LSD: a Line Segment Detector. Image Process Line. 2012;2:35–55.
- The ORFEO Tool Box Software Guide - Updated for OTB-5.10.0. <https://www.orfeo-toolbox.org/SoftwareGuide/index.html>. Accessed June 2017.
- OTB Cookbook: a Guide for OTB-Applications and Monteverdi Dedicated for Non-developers. <https://www.orfeo-toolbox.org/CookBook/>. Accessed June 2017.
- OTB Documentation. <https://www.orfeo-toolbox.org/documentation/>. Accessed June 2017.
- OTB Project Steering Committee. <https://wiki.orfeo-toolbox.org>. Accessed June 2017.
- Neteler M, Bowman MH, Landa M, Metz M. GRASS GIS: a multi-purpose Open Source GIS. Environ Modell Softw. 2012;31:124–30.
- The Open Source Geospatial Foundation. <http://www.osgeo.org/>. Accessed June 2017.
- OSGeo Code of Conduct. http://www.osgeo.org/code_of_conduct. Accessed June 2017.
- Ice: Lightweight Efficient Rendering for Remote Sensing Images. <http://fr.slideshare.net/otb/ice-lightweight-efficient-rendering-for-remote-sensing-images>. Accessed June 2017.
- Python Vflip: OpenGL Accelerated Image Viewer. <https://github.com/gfacciol/pvflip>. Accessed June 2017.
- Fenoy G BN, V R. Zoo-project: the open wps platform. Appl Geomatics. 2013;5(1):19–24.
- QGIS Processing Framework. https://docs.qgis.org/2.6/en/docs/user_manual/processing/index.html. Accessed June 2017.
- Michel J, Youssefi D, Grizonnet M. Stable mean-shift algorithm and its application to the segmentation of arbitrarily large remote sensing images. IEEE Trans Geosci Remote Sensing. 2015;53(2):952–64.
- Lassalle P, Inglada J, Michel J, Grizonnet M, Malik J. A scalable tile-based framework for region-merging segmentation. IEEE Trans Geosci Remote Sensing. 2015;53(10):5473–485.
- Haines M. On designing lightweight threads for substrate software. Institute for Computer Applications in Science and Engineering (ICASE); Washington, DC: National Aeronautics and Space Administration; 1997.
- Dagum L, Menon R. Openmp: an industry standard api for shared-memory programming. Comput Sci Eng IEEE. 1998;5(1):46–55.
- Cresson R, Hautreux G. A Generic Framework for the Development of Geospatial Processing Pipelines on Clusters. IEEE Geosci Remote Sens Lett. 2016;13(11):1706–10. doi:10.1109/LGRS.2016.2605138.
- Zhu Z, Woodcock CE. Object-based cloud and cloud shadow detection in Landsat imagery. Remote Sensing Environ. 2012;118:83–94. 00097. Accessed 13 Feb 2015.
- H.RAHMAN, G.DEDIEU. Smac: a simplified method for the atmospheric correction of satellite measurements in the solar spectrum. Int J Remote Sensing. 1994;15(1):123–43.
- The 6S Code Is a Basic RT Code Used for Calculation of Lookup Tables in the MODIS Atmospheric Correction Algorithm. <http://6s.ltdri.org/>. Accessed June 2017.
- Hagolle O, Dedieu G, Mougnot B, Debaecker V, Duchemin B, Meygret A. Correction of aerosol effects on multi-temporal images acquired with constant viewing angles: Application to Formosat-2 images. Remote Sens Environ. 2008;112(4):1689–701.
- Hagolle O, Huc M, Pascual D, Dedieu G. A multi-temporal and multi-spectral method to estimate aerosol optical thickness over land, for the atmospheric correction of formosat-2, landsat, vens and sentinel-2 images. Remote Sens. 2015;7(3):2668–91.
- The Theia Land Data Services Centre Is a French National Inter-agency Organization Designed to Foster the Use of Images Issued from the Space Observation of Land Surfaces. <https://www.theia-land.fr/>. Accessed June 2017.
- Bojinski S, Verstraete M, Peterson TC, Richter C, Simmons A, Zemp M. The concept of essential climate variables in support of climate research, applications, and policy. Bull Am Meteor Soc. 2014;95(9):1431–43.
- Infrastructure Pour l'Occupation des Sols Par Traitement Automatique Incorporant les Orfeo Toolbox Applications. <http://tully.ups-tlse.fr/jordi/iota2>. Accessed June 2017.
- Defries RS, Townshend JRG. Global Land Cover Characterization From Satellite Data: From Research To Operational Implementation. GTE/LUCC Research Review. Global Ecol Biogeograph. 1999;8(5):367–79.
- Hansen MC, Loveland TR. A review of large area monitoring of land cover change using landsat data. Remote Sens Environ. 2012;122(nil):66–74.
- Chen J, Chen J, Liao A, Cao X, Chen L, Chen X, He C, Han G, Peng S, Lu M, Zhang W, Tong X, Mills J. Global land cover mapping at 30m resolution: A pok-based operational approach. ISPRS J Photogramm Remote Sens. 2015;103(nil):7–27.
- GEOSUD (GEOinformation for Sustainable Development) Is a French Inter-agency Project Aiming to Ease the Access and the Use of Geospatial Information. <http://ids.equipex-geosud.fr/>. Accessed June 2017.
- ESA Sentinel-2 for Agriculture Project. <http://www.esa-sen2agri.org/SitePages/Home.aspx>. Accessed June 2017.
- SNAP - ESA Sentinel Application Platform. <http://step.esa.int>. Accessed June 2017.

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

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

Submit your next manuscript at ► springeropen.com