



Short communication

SEVIRI PrePro: A novel software tool for the pre-processing of SEVIRI geostationary orbit EO data products

George P. Petropoulos^{a, *}, Vasileios Anagnostopoulos^{b, c}^a Department of Geography & Earth Sciences, Aberystwyth University, SY23 2EJ Wales, United Kingdom^b Distributed and Knowledge Management Systems Lab, National Technical University of Athens, Greece^c InfoCosmos, Pindou 71, 13341 Athens, Greece

ARTICLE INFO

Article history:

Received 10 February 2016

Received in revised form

13 March 2016

Accepted 30 March 2016

Available online 25 May 2016

Keywords:

Earth observation

SEVIRI

Pre-processing

Operational products

Software tool

ABSTRACT

The Spinning Enhanced Visible and Infrared Imager (SEVIRI) is a geostationary orbit multispectral sensor on-board the Meteosat second Generation (MSG) platform, acquiring Earth Observation (EO) data over Earth's land surface from the optical to infrared parts of electromagnetic spectrum every 15 min. From the sensor a series of operational products are also provided to the user's community at no cost via EUMETSAT or LSA SAF portals.

Herein, an open access stand-alone software product developed in Java programming language is presented for automating key pre-processing steps to all the SEVIRI operationally distributed products. The software tool, named Seviri PrePro, makes use of present day multi-core processors and is able to process very large datasets in a short time period, making it appropriate as well for use in a High Performance Computing (HPC) environment. The practical usefulness of the toolkit is also demonstrated herein using as a case study the SEVIRI evapotranspiration (ET) product.

The development of SEVIRI PrePro is of significant importance to the SEVIRI users' community and is also very timely given that, to our knowledge, no similar software tool is freely distributed at present. Its use is anticipated to make a significant contribution to a large number of practical applications requiring use of SEVIRI data, including but not limited, weather forecasting and global climate monitoring at a range of geographical scales.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

Monitoring Earth's land surface and the inter-related processes, feedbacks and exchanges between the biosphere, atmosphere and hydrosphere that inherently shape the physical environments within our planet is a topic of key research priority today to be addressed (Mason and Calow, 2012; North et al., 2015). These complex processes are important drivers of weather and climate systems and govern the terrestrial hydrological, energy and biogeochemical cycles which are responsible for the existence of life in the planet (Wood et al., 2011; Destouni et al., 2013). In the last decades, Earth Observation (EO) technology has played an increasingly important role in determining various parameters characterising the land surface from space. This is because of its advantages related to the speed at which can be acquired (often

instantaneous updating of information) over large areas (synoptic views in a spatially contiguous fashion). Also, EO overcomes many problems inherent to conventional data acquisition methods (e.g. access to otherwise inaccessible areas). Additionally, remote sensing provides an efficient and cost-effective approach for the systematic repeat-coverage of areas of interest without a disturbing influence on the area (Kalivas et al., 2013). The advent of satellite-based EO over the last few decades, has reached a significant milestone in maturity that has allowed obtaining several land surface parameters from spaceborne EO systems at an operational level (Srivastava et al., 2015). Geostationary orbit satellites are of particular interest to studying and monitoring Earth's physical environment and the changes which occur by either anthropogenic activities or natural hazards. The principal advantage of those satellites is that they are able to provide a width of spectral information over large areas and at a time step compatible with the dynamics of several physical processes of the Earth system.

The Spinning Enhanced Visible and Infrared Imager (SEVIRI) is a geostationary orbit imaging radiometer which serves as the main

* Corresponding author.

E-mail address: george.petropoulos@aber.ac.uk (G.P. Petropoulos).

payload on-board the Meteosat Second Generation (MSG) satellite. This is a co-funded space mission between the European Space Agency (ESA) and EUMETSAT. SEVIRI has 12 spectral channels, consisting of three Visible and Near-InfraRed (VNIR) channels (centred at 0.6, 0.8 and 1.6 μm), eight InfraRed (IR) channels (centred at 3.9, 6.2, 7.3, 8.7, 9.7, 10.8, 12.0 and 13.4 μm) and one visible broadband channel (at 0.5–0.9 μm) called the High Resolution Visible channel (HRV). The satellite provides image data at 3 km spatial resolution at the sub-satellite point (nadir) for standard channels, and down to 1 km for the High Resolution Visible (HRV) channel, over 4 specific geographical regions (Europe, Africa - North_Africa and South_Africa- and South America), with a baseline repeat cycle of 15 min. A series of operational products from SEVIRI are provided by EUMETSAT and distributed by the Satellite Application Facility (SAF) on Land Surface Analysis (LSA) (<http://landsaf.meteo.pt/>). SEVIRI data have been used in a range of applications including for example solar modelling (Bojanowski et al., 2013), agriculture studies (Eerens et al., 2014), wildfires (Carvalho et al., 2010) and meteorological studies (Ginoux et al., 2004). Various software tools have also been developed so far for using various SEVIRI data or products (e.g. Cermak et al., 2008; Bojanowski et al., 2013). Yet, despite the large geographical coverage acquisition of SEVIRI, to our knowledge, there is no software platform available at present to support the pre-processing of those data including as well operational products.

In purview of the above, is presented herein an open source stand-alone software tool, developed in Java, which allows performing basic, yet of key importance, pre-processing steps to the SEVIRI operationally distributed products. The software platform, named “SEVIRI PrePro”, makes also use of present day multi-core processors and is able to process fast very large datasets, making it also suitable for use in a High Performance Computing (HPC) environment. Furthermore, the practical applicability of the software toolkit is demonstrated using as an example an operationally distributed product from SEVIRI, although equally any other operational product may be used.

2. The SEVIRI pre-pro toolkit

2.1. Introduction to the toolkit

The SEVIRI PrePro tool is a Java 8 based cross-platform tool that enables the user to extract information (i.e. geophysical quantities of interest) from a SEVIRI product file. SEVIRI product files come in .HDF5 hierarchical format and each granule contains an image but instead of typical RGB imaging which assigns a vector of R, G and B values per image site, a vector of values corresponding to geophysical parameters is contained in this image. These values are sampled per image site which is part of the image grid. There can be four grids, one per region. The sites on the grid correspond to pairs of latitude and longitude and are indexed as in a typical image by image coordinates constrained by the width and height of the image. The translation from the grid (image) co-ordinates to region specific WGS84 coordinates is implemented through a list of region specific files that one has to download from the LSA SAF site after registration. The user of the tool can select various geographical sites from which can export, in batch mode, from near image sites, the corresponding image vector irrespective of operational product. This is facilitated by a convenient GUI. For HPC reasons, a headless option has also been developed and is also available which comes through a Command Line Interface (CLI). Moreover the user can export via a Region of Interest (ROI) parts of the image. In SEVIRI PrePro the exporting is done in the native format of BEAM/VISAT open source software platform (Brockmann, 2004), namely BEAM DIMAP format, for compatibility reasons. The tool automates the

batch exporting through a ROI defined by a shape file (.shp). The user workflow is also easy to implement. Thanks also to recent Java releases, SEVIRI PrePro is able to take advantage of all processors installed in a given computer system. This way, the pre-processing is multithreaded and efficient in performance.

2.2. Installation of the SEVIRI PrePro toolkit

The SEVIRI PrePro has been developed as open source software, released under the terms of the GNU General Public Licence, distributed with a complete documentation for its use by Aberystwyth University, United Kingdom (www.aber.ac.uk/seviriprepro). The software is under GPL and the user can either download the source code or build it using Eclipse (for Java developers) or download a binary release from the same site. In order to build it from source, the user needs a BEAM/VISAT compiled in Java 8 by using the corresponding build. The preferred way is to use a pre-compiled binary, which is available as a .7z file download. The toolkit download contains three launchers, the main.jar file, a sample file in the Comma-Separated Values (.CSV) format for pixel extraction, a folder with .jar dependencies and a folder with shapefiles (.shp)(Fig. 2).

To execute SEVIRI PrePro, first the static .HDF5 (Folk et al., 2011) files for the region of interest need to be downloaded, which are provided by LSA-SAF (Cihlar et al., 1999), subject to registration to their site. Once this step is completed, LSA-SAF will need to authorise their account before they can request the necessary static files. Using this authorisation the user can download the necessary files located in the “Auxiliary Data” tab. On the website is a list of .bz2 files in .HDF5 format. The user is advised to download all available files and store them in a folder since it can be used for further reference. The structure of the folder is illustrated in Fig. 1. After the .HDF5 data have been acquired, Java 8 needs to be installed to take advantage of speed and Graphical User Interface (GUI) available through Java FX improvements. The installation has been tested with the corresponding JDK (at time of writing JDK 8u60 was the most up to date version and used in implementation).

2.3. SEVIRI PrePro implementation

2.3.1. The graphical user interface (GUI)

The user should first update (if necessary) the run GUI.bat file. Java location and assigned memory can be modified. The defaults are a user wide installation of Java and 2048 MB of assigned memory. If the environment is Linux, the user can copy and paste the instruction at the shell prompt and execute it. A GUI appears as shown in Fig. 3. One can immediately see that the tabbed interface at the bottom offers two utilities. The first is the “Country Export” and the other one is “Pixel Extraction”. The generic part of the application allows one to re-project the whole products in the Equirectangular, plate carrée projection (Snyder, 1993). In the remainder of the section each of the three main interfaces of the software toolkit menu (shown in Fig. 3) are explained.

2.3.2. Inputs requirements

In the generic part of the toolkit application the user needs to fill some input parameters (Fig. 4). The user must first specify the folder where the reprojected BEAM DIMAP files will be saved. The user has also to provide the folder of the LSA SAF static files in order to achieve the grid translation. Then the user must specify where the SEVIRI product files reside. The folder must have a flat structure as no recursive searching has been implemented. Finally, the region the files belong must be provided. The assumption is that all the files in the folder have to do with the same region. One can start the reprojection process. Procedure previously implemented in the

Download English Version:

<https://daneshyari.com/en/article/6962476>

Download Persian Version:

<https://daneshyari.com/article/6962476>

[Daneshyari.com](https://daneshyari.com)