MODIS Active Fire Product (MOD14) Science Processing Algorithm
MOD14_SPA

General

The NASA Goddard Space Flight Center's (GSFC) Direct Readout Laboratory (DRL), Code 606.3 developed this wrapper software for the National Polar-orbiting Operational Environmental Satellite System (NPOESS) Preparatory Project (NPP) In-Situ Ground System (NISGS) and the International Polar Orbiter Processing Package (IPOPP).

Users must agree to all terms and conditions in the Software Usage Agreement on the DRL Web Portal before downloading this software.

Software and documentation published on the DRL Web Portal may occasionally be updated or modified. The most current versions of DRL software are available at the DRL Web Portal:

http://www.directreadout.sci.gsfc.nasa.gov

Questions relating to the contents or status of this software and its documentation should be addressed to the DRL via the Contact Us mechanism at the DRL Web Portal:

http://directreadout.sci.gsfc.nasa.gov/index.cfm?section=contact%20us

Algorithm Wrapper Concept

The DRL has developed an algorithm wrapper to provide a common command and execution interface to encapsulate multi-discipline, multi-mission science processing algorithms. The wrapper also provides a structured, standardized technique for packaging new or updated algorithms with minimal effort.

A Science Processing Algorithm (SPA) is defined as a wrapper and its contained algorithm. SPAs will function in a standalone, cross-platform environment to serve the needs of the broad Direct Readout community. Detailed information about SPAs and other DRL technologies is available at:

http://directreadout.sci.gsfc.nasa.gov/index.cfm?section=technology

Software Description

This DRL software package contains the Level 2 MODIS Active Fire Product (MOD14) SPA. This algorithm primarily uses brightness temperatures derived from MODIS 4 and 11 micrometer channels (bands 21, 22 and 31) to detect fires. MODIS bands 1, 2, 7 and 32 are used to reject false alarms and to mask clouds. The algorithm reads a MODIS 1-km Level 1B file along with the associated geolocation file and identifies active fires. The output is a two-dimensional fire mask in Hierarchical Data Format (HDF). Additionally the SPA creates a text file containing

useful information (latitude, longitude, acquisition date, channel 21/22 brightness temperature, Fire Radiative Power (FRP), and detection confidence) about each fire pixel detected. The MOD14_SPA functions in two modes: Standalone, or as an IPOPP plug-in.

Software Version

Version 1 of the DRL algorithm wrapper was used to package the SPA described in this document.

Credits

The MODIS Active Fire Product (MOD14) Production Code (Version 5.0.1, June 22, 2006) was provided to the DRL by the MODIS Land Rapid Response Team.

Prerequisites

To run this package, you must have the Java Development Kit (JDK) or Java Runtime Engine (JRE) (Java 1.5 or higher) installed on your computer, and have the Java installation bin/ subdirectory in your PATH environment variable.

Program Inputs and Outputs

This package uses the MODIS 1-km L1B Calibrated Geolocated Radiances (MOD021KM, MYD021KM) HDF product and MODIS Geolocation HDF product (MOD03, MYD03) as inputs. Recent MODIS L1B products over the eastern US region can be obtained from the DRL ftp site.

For Terra MODIS: ftp://is.sci.gsfc.nasa.gov/gsfcdata/terra/modis/level1/

For Aqua MODIS: ftp://is.sci.gsfc.nasa.gov/gsfcdata/aqua/modis/level1/

Datasets from other areas can be obtained either from the DAAC or from other Direct Readout stations.

Output is the MODIS Level 2 Fire product.

Installation and Configuration

This section contains instructions for installing an SPA in a standalone configuration. SPAs may also be installed dynamically into an IPOPP framework; instructions for this type of installation are contained in the IPOPP User's Guide.

Download the MOD14_5.0.1_SPA_1.0.tar.gz and
MOD14_5.0.1_SPA_1.0.testdata.tar.gz (optional) files into the same directory.

Decompress and un-archive the MOD14_5.0.1_SPA_1.0.tar.gz and
MOD14_5.0.1_SPA_1.0_testdata.tar.gz (optional) files:

```
$ tar -xzf MOD14_5.0.1_SPA_1.0.tar.gz
$ tar -xzf MOD14_5.0.1_SPA_1.0_testdata.tar.gz
```

This will create the following subdirectories:

    SPA

        mod14

            algorithm

            ancillary

            station

            wrapper

            testscripts

            testdata

For convenience, this package contains binaries statically pre-compiled on an Intel-compatible 32-bit computer running under Fedora Core 4, using gcc 4.0.2. The binaries should work on most Linux OS/platforms. If you get an error message while running the testscripts (refer to the next section, "Software Package Testing and Validation"), you may need to recompile the software for your platform/OS combination. Refer to the Appendix for instructions on recompiling the software.

Software Package Testing and Validation

The testscripts subdirectory contains a test script that can be used to verify that your current installation of the SPA is working properly, as described below. Note that the optional MOD14_5.0.1_SPA_1.0_testdata.tar.gz file is required to execute these testing procedures:

    Step 1: cd into the testscripts directory

    Step 2: Run the 'run-mod14' script by typing: $./run-mod14

A successful execution usually takes some time (approximately 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck, do not become impatient. If everything is working properly, the scripts will terminate with a message such as:

Output modis.firedetection is
/home/IPOPP/SPA/mod14/testdata/output/MYD14.07054183325.hdf

Output modis.fireloc.txt is
/home/IPOPP/SPA/mod14/testdata/output/MYD14.07054183325.txt

You can cd to the output directory to verify that the science product exists. If the product does exist, then the SPA works perfectly. Test output product(s) are available for comparison in the testdata/output directory. If there is a problem and the code terminates abnormally, then the problem can be identified using the log files. Log files are generated automatically within the directory used for execution.

They start with stdfile* and errfile*. Other problems may be caused by incompatibility between your system and the binaries provided with this software package. In that case you may need to recompile the software for your platform/OS combination. Refer to the Appendix for instructions on recompiling the software. Please report any errors that cannot be fixed to the DRL.

Program Operation

In order to run the package using your own input data, you can either use the run scripts within the wrapper directories, or modify the test scripts within the testscripts directory.

To Use the Run Scripts

Identify the 'run' script: The SPA/mod14/wrapper/mod14 directory contains the 'run' script. Execute the 'run' to execute the MOD14_SPA. Note that to execute 'run', you must have java on your path.

Specify input parameters using <label value> pairs: To execute the 'run' script, you must supply the required input and output parameters. Input and output parameters are usually file paths. Each parameter is specified on the command line by a <label value> pair. Labels are simply predefined names for parameters. Each label must be followed by its actual value. Each SPA has its own set of <label value> pairs that must be specified in order for it to execute. There are two types of <label value> pairs that the MOD14_SPA uses, as follows:

   a) Input file label/values.   These are input file paths. Values are absolute or relative paths to the corresponding input file.

   b) Output file labels.   These are output files that are produced by the SPA. Values are the relative/absolute paths of the files you want to generate.

The following table contains labels, and their descriptions, required by the MOD14_SPA.

Input File Labels	Description
modis.mxd021km	MODIS 1-km L1B Calibrated Geolocated Radiances HDF file (MOD021KM, MYD021KM)
modis.mxd03	MODIS Geolocation HDF file (MOD03, MYD03)
Output File Label	**Description**
modis.firedetection	MODIS L2 Active Fire product
modis.fireloc.txt	Fire Information text file

Note: The Fire Information text file will contain latitude, longitude, acquisition date, channel 21/22 brightness temperature, Fire Radiative Power (FRP), and detection confidence information about each fire pixel detected. The value for FRP is extracted directly from SDS FP_power. For the Collection 5 MOD14/MYD14 product this is valid; for Collection 4 it is not. The dumped value must be multiplied by the pixel area to obtain the actual FRP. See the "MODIS Collection 4 Active Fire Product User's Guide" for details.

Execute the 'runs': The following is an example of a command line to run MOD14_SPA from the testscripts subdirectory. You can run it from the directory of your choice by using the correct paths to the 'run' scripts and your datasets.

$ ..SPA/mod14//wrapper/mod14/run \

    modis.mxd021km ../testdata/input/MYD021KM.07054183325.hdf \

    modis.mxd03 ../testdata/input/MYD03.07054183325.hdf \

    modis.firedetection ../testdata/output/MYD14.07054183325.hdf

Output modis.firedetection is
/home/IPOPP/SPA/mod14/testdata/output/MYD14.07054183325.hdf

Output modis.fireloc.txt is
/home/IPOPP/SPA/mod14/testdata/output/MYD14.07054183325.txt

A successful execution usually takes some time (approximately 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck, do not become impatient. If execution fails, you will see an error message indicating

the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct the problem and run again. The problem can also be identified using the stdfile* and errfile* log files. Log files are generated automatically within the directory used for execution. Other problems may be caused by incompatibility between your system and the binaries provided with this software package. In that case you may need to recompile the software for your platform/OS combination. Refer to the Appendix for instructions on recompiling the software. The executions will create some temporary files (or symbolic links) in your execution directory. You can delete them after the run.

To Use the Script in the testscripts Directory

One simple way to run the algorithms from any directory of your choice, using your own data, is to copy the run-mod14 script from the testscripts directory to the selected directory. Change the values of the variables WRAPPERHOME, L1HOME and OUTPUTHOME to reflect the file paths of the wrapper directories and the input/output file paths. Then modify the input/output file name variables. Run the script to process your data.

Appendix

Instructions for Recompiling the MOD14_SPA Software

If you are going to build the binaries in this package, you will need to install the Hierarchical Data Format (HDF) library, HDF4.2rX on your system. (The DRL recommends using version HDF4.2r1.) The HDF4.2rX library can be obtained in precompiled binary form at http://hdf.ncsa.uiuc.edu. Download the library for your platform and install it. Make sure to get the JPEG, ZLIB and SZIP libraries for your platform and place all of the library and include files under the lib/ and include/ directories within the HDF install directory. Then cd into the SPA/mod14/algorithm directory. Modify "Makefile" to use the correct C compiler, and modify any additional compilation options that may be necessary. Edit the HDFHOME variable to point to the correct HDF install directory. First run "make clean" to remove any existing binaries. Next run "make" to build the binaries.