Sensagri

Sentinels Synergy for Agriculture

Deliverable D3.6

Software of the final operational SSM prototype exploiting S1&S2

V 1.0

User Manual

The research leading to these results has received funding from the European Union's Horizon 2020 Research and Innovation Programme, under Grant Agreement nº 730074

Document information

Project Number	730074	**Acronym**	SENSAGRI
Full Title	Sentinels Synergy for Agriculture		
Project URL	http://sensagri.eu		
Project Coordinator	José F. Moreno. IPL – University of Valencia (Spain)		
EU Project Officer	Massimo Ciscato		

Deliverable	**Number**	D3.6	**Title**	Software of the final operational SSM prototype exploiting S1&S2
Work Package	**Number**	WP3	**Title**	Surface Soil Moisture Service

Date of Delivery	**Contractual**	M30	**Actual**	M34	
Status	Version 0.1		Final ☐		
Type[1]	R ☒	DEM ☐	DEC ☐	OTHER ☐	ETHICS ☐
Dissemination Level[2]	PU ☒	CO ☐	EU-RES ☐	EU-CON ☐	EU-SEC ☐

Responsible partner	CNR-IREA			
Responsible Author	**Name**	Giuseppe Satalino,	**E-mail**	giuseppe.satalino@cnr.it
	Partner	CNR-IREA	**Phone**	+39 080 5929435
Other authors	Francesco Mattia, Anna Balenzano, Francesco Lovergine			

Abstract (for dissemination)	This document is the User Manual of the operational Surface Soil Moisture (SSM) prototype exploiting Sentinel-1 (S1) and Sentinel-2 (S2) data, described in the deliverable D3.5 “Final operational SSM prototype exploiting S1&S2”. The document reports on the procedure for the software installation, the preparation of the input/output data and the commands for processing a temporal series of S1 & S2-NDVI data.
Keywords	Soil moisture retrieval, Short Term Change Detection, Sentinel-1 & -2

Version Log			
Issue Date	**Rev. No.**	**Author**	**Change**
30 August 2019	1.0	Giuseppe Satalino	Final version

[1] R = Document, report; DEM = Demonstrator, pilot, prototype; DEC = Websites, patent fillings, videos, etc; OTHER; ETHICS = Ethics requirement

[2] PU = Public; CO = Confidential (Consortium and Commission Services); EU-RES = Restreint UE; EU-CON Confidentiel UE; EU-SEC = Secret UE (Commission Decision 2005/444/EC)

Table of Contents

Document information....3
List of Tables....5
List of Figures....5
1. Introduction....6
1.1. Scope of the document....6
1.2. Notations, abbreviations and acronyms....6
2. Software installation....7
2.1. Software and hardware requirements....7
2.2. Installation procedure....7
3. Input/output file specifications....9
4. Running the SSM prototype code....14
Reference documents....16
Web references....16

List of Tables

Table 1. Input file format for *script_SMOSAR_processings.bash* 11
Table 2. Input file format for *SMOSAR_PROCESSOR_V2_0.sav* 12

List of Figures

Figure 1. Directories of the SMOSAR folder 8

1. Introduction

1.1. Scope of the document

The Software User Manual provides a description of the installing and operating procedures of the operational Surface Soil Moisture (SSM) prototype code, referred to as "Soil MOisture retrieval from multi-temporal SAR data" (SMOSAR), developed in SENSAGRI. The Final operational SSM prototype is described in [D3.5], whereas its theoretical basis is reported in [D3.8].

In the following sections, firstly the installing and operating procedures of the prototype software and the input/output file are described. Then, an example of how to run the code, using the included test data set, is illustrated.

1.2. Notations, abbreviations and acronyms

CNR	Consiglio Nazionale delle Ricerche
EO	Earth Observation
ENL	Equivalent Number of Looks
ESA	European Space Agency
IREA	Istituto per il Rilevamento Elettromagnetico dell'Ambiente
S1	Sentinel-1
S2	Sentinel-2
SAR	Spaceborne Synthetic Aperture Radar
SMOSAR	Algorithm for Soil Moisture Retrieval using Sentinel 1 data
SSM	Surface Soil Moisture
STCD	Short Term Change Detection

2. Software installation

2.1. Software and hardware requirements

The SSM prototype code consists of IDL executable files and other additional bash scripts developed and tested under the Linux operating system (kernel 4.9.0-9-amd64), running IDL in 64-bit mode, and a platform using 2xCPU 14 core Intel Xeon E5-2660v4 2.0Ghz, 35M cache, 128GB RAM DIMM 2400MT/s RDIMM.

The routines in IDL have been developed by using the IDL version 8.6.0 [W1]. The IDL executable files (*.sav files) have been obtained by using the IDL SAVE procedure, which saves all IDL routines by using the XDR (eXternal Data Representation) format. To execute the IDL *file.sav* under the linux platform, the command is: *idl –rt=file.sav.* To execute the file without an IDL license, the user has to install the IDL Virtual Machine (IDL VM) [W1]. The IDL Virtual Machine is free. However, for its downloading, it is required to be registered on the ENVI website. In this case, the command is: *idl -vm=file.sav.*

The bash script contains calls to GRASS GIS 7.2 [W2] and GDAL tools [W8], which need to be installed on the working platform. To work properly, the environment of the bash script has to be initialized and for this task, an initialization script is supplied (see file *sm-tools/INSTALL.md*).

The SSM prototype code is delivered together with one test data set already prepared for an experiment of data processing. All files are supplied in a single compressed UNIX-TAR file (file *SMOSAR_SENSAGRI_20190730.tar.gz*).

2.2. Installation procedure

For its installation in a Linux platform, the user has to copy the .tar.gz file into a working directory and then extract all the files with the linux shell command:

tar –xvzf SMOSAR_SENSAGRI_20190730.tar.gz

This command will create the SMOSAR folder with all the directories, as shown in Figure 1.

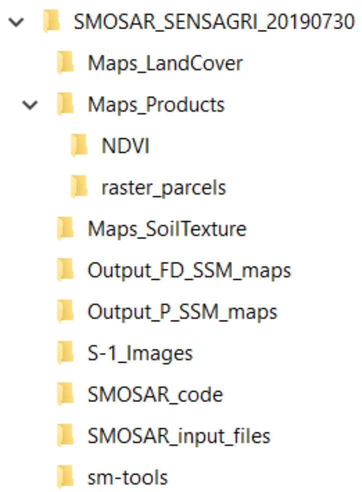

Figure 1. Directories of the SMOSAR folder

The directory ***SMOSAR_code*** contains the routines implementing the Final operational SSM prototype code, which consists of IDL executable files and other additional bash scripts, and input files:

- *Calibration_Mat_v20190730.sav*
- *input_script_SMOSAR_processings.txt*
- *make_easy_hybrid_ssm_map.sh*
- *script_SMOSAR_processings_v2_0.bash*
- *SMOSAR_MAPS_PREPROCESSING_V2_0.sav*
- *SMOSAR_PROCESSOR_V2_0.sav*

The *script_SMOSAR_processings_v2_0.bash* is the main script designed to manage the processing chain for a complete processing of a temporal series of S-1 and S2-NDVI data. The processing chain consists of three processing blocks, as described in [D3.5], i.e. Block1, Block2 and Block3, and it is executed by three routine calls:

1) *SMOSAR_PROCESSOR_V2_0.sav*
2) *SMOSAR_MAPS_PREPROCESSING_V2_0.sav*
3) *make_easy_hybrid_ssm_map.sh*

The routine *SMOSAR_PROCESSOR_V2_0.sav* implements the SMOSAR online, offline refinement and low pass filter modules, whereas the *SMOSAR_MAPS_PREPROCESSING_V2_0.sav* performs the masking of the SSM maps [D3.5]. Finally, the bash script *make_easy_hybrid_ssm_map.sh* implements the averaging of SSM at field scale and the fusion of SSM maps at different spatial resolutions and other tasks, such as the export of SSM maps in the PNG format by applying a colour-table useful for a quick map display.

The directory ***SMOSAR_input_files*** contains the input files required by the *SMOSAR_PROCESSOR_V2_0.sav* routine. They specify all input/output parameters and images required by the SSM retrieval algorithm. Moreover, this directory will contain subfolders to save the log files produced during the processing. The detailed description of these files is reported in the next section.

The directory ***S-1_Images*** contains the time series of S1 images (both co- and cross-polarized) and the local incidence angle images.

The directories ***Maps_LandCover*** and ***Maps_SoilTexture*** contain ancillary data required by the SMOSAR processor (i.e., land cover maps and soil texture maps).

The directory ***Maps_products*** contains *NDVI* maps (required by the *SMOSAR_MASK_PREPROCESSING_V2_0.sav* routine), a *raster file* derived from a shapefile of field parcel borders (required by the *make_easy_hybrid_ssm_map.sh* script) and temporary products produced during the processing.

The directories ***Output_FD_SSM_maps*** and ***Output_P_SSM_maps*** *are* used to write the temporary Output *FD-* and *P-SSM* maps (i.e., Fast Delivery and Precision SSM maps). The final SENSAGRI *P_SSM_hybrid* maps will be written in the *Output_P_SSM_maps* directory.

All input images require their own ENVI header file *.hdr [W3]. All the output images are written with the correspondent ENVI header file. This file system structure is not mandatory. The user could use another directory structure provided that the path and names are properly addressed into the input files.

3. Input/output file specifications

The input/output images managed by the three routines of the SSM prototype code are firstly described. Then, the prototypes of the input files are illustrated.

1) Routine *SMOSAR_PROCESSOR_V2_0.sav*:

- *Input: S1 images (N- VV and VH images, and the correspondent local incidence angle images).* It is expected that the input data are co-registered stack of geocoded, calibrated and speckle filtered backscatter (linear) and angles (radiant) images, written with a binary float*4bytes format. Image dimensions and pixel size must be equal to those reported in the input file. The coordinate system adopted is the geographic Lat/Lon (units in degree), with WGS84 datum. Information of each image are reported in the correspondent ENVI header file.

- *Input: Land cover maps.*
  The expected image content is an integer (representing the class code) written as binary integer*2bytes. The land cover map requires the correspondent look up table, containing a list of pairs (LC_class_code (integer) LC_class_type (string)). Among them, the list of selected land cover classes has to be supplied.
  It is expected that the map extent is equal to the one of the S1 geocoded image, i.e. the pixel size and image dimensions can be equal or different with respect the S-1 image, provided that the area covered is the same. If they are different, an image resampling is performed. Moreover, a check on the temporal difference between the land cover map date with respect to the last SAR image acquisition date is performed. This difference must be smaller or equal than 5 years.

- *Input: Soil Texture maps.*
  The clay and sand maps are required. The expected image content is an integer value ranging in [0,100] expressing a fraction of the soil component. Silt map is derived as a complement to 100. The data type of these maps is binary 1*byte integer.
  As for land cover maps, a map extent equal to the S-1 geocoded image one is required.

- *Input: Calibration parameters for the STCD algorithm.*
  The file *Calibration_Mat*_v20190730*.sav* (in the *SMOSAR_code* directory, which is in the idl *.sav* format) contains calibration parameters for the STCD algorithm found on the base of the experimental test sites employed to set up this version of the software. This file is read automatically by the *SMOSAR_PROCESSOR_V2_0.sav* routine. These parameters could be updated when additional test sites will be available. In this case, a new *Calibration_Mat.sav* file will be provided and the users will need just to substitute the file.

- *Output: FD_SSM and P_SSM maps.*
  Temporary files are written/deleted during the processing. Finally, the *FD_SSM* maps are obtained by the STCD retrieval algorithm, whereas the *P_SSM* maps are obtained by the off-line SSM refinement module, by averaging up to N=4 *FD_SSM_MAP* maps available for the same acquisition date [D3.5]. The final products are a spatial mean (*P_SSM_MAP.img_mean*), and standard deviation (*P_SSM_MAP.img_stddev*) maps obtained at a lower spatial resolution (i.e. 1km). During the processing, the percentage of not-null pixels within the averaging areas is computed and the result is used as a reliability map to mask (set to 0) both the maps to 0 when the percentage is lower than a given threshold (e.g. 33%).

2) Routine *SMOSAR_MAPS_PREPROCESSING_V2_0.sav*:

- *Input: NDVI maps.*
  S2-NDVI maps contain float*4byte values ranging in [-1, 1]. It is expected that *NDVI* maps are colocated in space and time with respect S1. For this reason, *NDVI* maps must have the same dimensions and pixel size of the correspondent S1 images. Moreover, it is expected that they contain a mosaic of *NDVI* maps derived from all the S2 tiles covering the S1 image AND acquired during a temporal window of +/-2 days (when both S2A and S2B are available) with respect the S1 acquisition date.

- *Input: temporary SSM maps and masking images.*
  These data are temporary produced by *SMOSAR_PROCESSOR_V2_0.sav* and are used to mask SSM maps by abrupt changes of roughness or vegetation.

- *Input: working parameters for maps preprocessing.*
  The file *Calibration_Mat* v20190730*.sav* contains the parameters needed by the routine for the SSM masking.

- *Output: preprocessed SSM maps.*
  These temporary data are used by the script *make_easy_hybrid_ssm_map.sh*.

3) Script *make_easy_hybrid_ssm_map.sh*:

- *Input: Low Resolution P_SSM maps, preprocessed SSM maps.*
  These maps, which have a different pixel size, are resampled to a pixel size of 0.0005 degree.

- *Input: raster file derived from a shapefile of field parcels.*
  This map has the same pixel size of S1 and can have dimensions equal or lower compared to those of S1, with a format integer*4. It is obtained from a shapefile of field parcels, whenever available, and each area contains a field identifier integer value. They are used to make averages at field scale. The averaging process produces a spatial mean and standard deviation layers and also a percentage of not-null pixels within the averaging areas, as seen for the *P_SSM* maps. Both the mean and stddev maps are set to 0 when the percentage is lower than a given threshold (e.g. 33%).

- *Ouput: P_SSM_hybrid maps.*
  The output SENSAGRI *P_SSM_hybrid* maps come from a fusion process: the background is the *Low Resolution P_SSM* map and the foreground is the *preprocessed SSM* map averaged at field scale. The format is geocoded three band TIF images (projection: geographic lat/lon, pixel:0.0005 deg, datum: WGS84, see note a1) for the naming convention), whose bands are:

  - BAND 1 = SSM_mean, i.e. SSM averages at field scale using the information of parcel borders, wherever available, or SSM averages at a resolution of 1km [Unit: m^3/m^3*10000];
  - BAND 2 = SSM_stddev, i.e. standard deviation associated to the averaging [Unit: m^3/m^3*10000];
  - BAND 3 = CV, i.e. Coefficient of Variation defined as stddev/mean*10000 [dimensionless].

Input files prototypes

Before executing the main *script_SMOSAR_processings.bash,* the user has to specify the field values of the next two input files:

a) SMOSAR_code/input_script_SMOSAR_processings.txt

The format of this input file required by the *script_SMOSAR_processings.bash* is reported in Table 1.

Table 1. Input file format for *script_SMOSAR_processings.bash*

Name=value	Description
track_dir=<*string*>	Used as folder name of S-1 data processed, e.g. *ApulianTav_S1_A146*
date_from_proc=*yyyymmdd*	S-1 Starting date of processing e.g. *20180401*
date_to_proc=*yyyymmdd*	S-1 Stopping date of processing e.g. *20180530*
string_processing={*online offline make_hybrid*}	Processing chain for the images included in temporal window from *date_from_proc* to *date_to_proc*
label_run=<*string*>	String added to the default folder name of the output maps
list_img_combined=<*string*>	Name of the list of images which is automatically filled with proper image names used by *COMPUTE_CHANGE_ROUGHNESS_VEG.sav*

rasterized_fields_map=<*name_img*>	Image used by *make_easy_hybrid_ssm_map.sh*

Notes:

a1) the output *P_SSM_hybrid* maps are written into the default subfolder:
Output_P_SSM_maps/< track_dir>/STCD_run_"label_run"_hybrid);
Default_names for the *P_SSM_hybrid* maps:
SSM_P_STCD_ yyyymmdd _S1x_IW_GRDH_1SDV_yyyymmddThhmmss_MNxx_RLxxxx.vv.GTC_MAP.img.tif,
where:
-*SSM_P_STCD* = Precision Surface Soil moisture product obtained by the STCD algorithm;
-*yyyymmdd* = Map date: year, month, day;
-*S1x_IW_GRDH_1SDV_yyyymmddThhmmss* = Sentinel-1 product type (string derived from the S-1 naming convention, i.e. *x*=A,B*; IW*=Interferometric Wide swath; *GRDH*=Ground Range Detected High resolution; *1SDV*=Level 1 Standard product, dual VV+VH polarisation; *yyyymmddThhmmss*=sensing date and time) (see [W4] for specifications);
-*MNxx_RLxxxx* = nr of frames into the S-1 mosaic and Reference Latitude of the northern frame;
-*vv.GTC* = S-1 product (vv polarization, Geocoded Terrain Corrected);
-*MAP.img.tif* = TIFF format of the SSM product.

a2) the file exchanges among the three routines (including the list on S2-NDVI data) are managed automatically by the *script_SMOSAR_processings.bash.*

b) SMOSAR_input_files/input_SMOSAR_processor.txt

Table 2 reports the content of the input file read by the *SMOSAR_PROCESSOR_V2_0.sav.*
The structure is fixed, with two fields, i.e. the field name and the field value. The lines starting with the symbol "#" are comments. The ranges of the parameters are also reported.

Table 2. Input file format for *SMOSAR_PROCESSOR_V2_0.sav*

Name	Value	Description
#SMOSAR_SSM_ PROCESSOR		
Retrieval_Algorithm	*string*	{STCD {online | offline} 1
Run_log_filename	*string*	Name for the log_file
Parameter_N_VV_S-1	*integer*	Number of S-1-VV images (N=4)
#S-1_AND_INC_IMG_INFO		
S-1_IMG_samples	*integer*	Columns of S-1 & INC images (m)
S-1_IMG_lines	*integer*	Rows of S-1 & INC images (n)
S-1_IMG_PixelSize	*float*	Pixel size of S-1 & INC images (pixsiz>0, degree)
S-1_IMG_dir_path	*string*	Pathname of S-1 & INC images
S-1_IMG_filenameVV_1	*string*	Filename of the 1st S-1 image VV
S-1_IMG_filenameVH_1	*string*	Filename of the 1st S-1 image VH
S-1_IMG_filenameINC_1	*string*	Filename of the 1st S-1 image INC
S-1_IMG_date1	*yyyymmdd*	Acquisition date of the 1st S-1 image
....	....	....
S-1_IMG_filenameVV_N	*string*	Filename of the Nth S-1 image VV
S-1_IMG_filenameVH_N	*string*	Filename of the Nth S-1 image VH

S-1_IMG_filenameINC_N	*string*	Filename of the Nth S-1 image INC
S-1_IMG_dateN	*yyyymmdd*	Acquisition date of the Nth S-1 & INC image
#LANDCOVER_MAP_INFO		
LC_MAP_samples	*integer*	Columns of the Land cover map (r)
LC_MAP_lines	*integer*	Rows of the Land cover map (s)
LC_MAP_PixelSize	*float*	Pixel size of the Land cover map
LC_MAP_dir_path	*string*	Pathname of the Land cover map
LC_MAP_filename	*string*	Filename of the Land cover map
LC_MAP_date	*yyyymmdd*	Acquisition date of the Land cover map
LC_MAP_look_up_table_filename	*string*	Filename of the land cover lookup table (Note b1)
LC_MAP_class_codes	*T integers*	Codes of the selected classes, T >=1 and class_id=1,…,Nr_of_classes
#SOILTEXTURE_MAP_INFO		
ST_MAP_samples	*integer*	Columns of the Soil texture map
ST_MAP_lines	*integer*	Rows of the Soil texture map
ST_MAP_PixelSize	*float*	Pixel size of the Soil texture map
ST_MAP_dir_path	*string*	Pathname of the Soil texture map
ST_MAP_filename_clay	*string*	Filename of the Soil texture map clay
ST_MAP_filename_sand	*string*	Filename of the Soil texture map sand
#OUTPUT_SSM_FD_PROCESSOR		
SSM_FD_ MAP_dir_path	*string*	Pathname of SSM FD maps
#SSM_FD_MV_MAP_filenames:default_names		(Note b2)
#OUTPUT_SSM_P_PROCESSOR		
SSM_P_ MAP_dir_path	*string*	Pathname of SSM P- and Error maps
#SSM_P_MAP_filename:default_name		(Note b3)

Notes:

b1) The Land cover map requires the correspondent look up tables, containing a list of pairs
(LC_class_code(integer) LC_class_type(string));

b2) Default_names for SSM_FD_ MAP filenames:
SSM_FD_STCD_ yyyymmdd _S1x_IW_GRDH_1SDV_yyyymmddThhmmss_MNxx_RLxxxx.vv.
GTC_MAP_<Map_index>.img_{mean|stddev},
where:
-*SSM_FD_STCD* = Fast Delivery Surface Soil moisture product obtained by the STCD algorithm;
-*yyyymmdd* = Map date: year, month, day;
-*S1x_IW_GRDH_1SDV_yyyymmddThhmmss* = Sentinel-1 product type (see a1);
-*MNxx_RLxxxx* = see a1);
-*vv.GTC* = see a1);
-*Map_index* = number in the range [1,...,N];

b3) Default_name for SSM_P_ MAP_filename:
SSM_P_STCD_ yyyymmdd _S1x_IW_GRDH_1SDV_yyyymmddThhmmss_MNxx_RLxxxx.vv.
GTC_MAP.img_{mean|stddev}
where:
-*SSM_P_STCD* = Precision Surface Soil moisture product obtained by the STCD algorithm.

4. Running the SSM prototype code

To run the prototype code under a Linux operating system the user has to:

A. set the two input file prototypes shown in the previous section;
B. execute the script *script_SMOSAR_processings.bash.*

Examples of input files are given in the following and the names used refer to files of the test data set included. The data set is composed of a stack of 7 S-1 images (S-1A and S-1B) acquired over the Apulian Tavoliere (Italy) from 20180304 to 20180409, ASC pass, track A146, and of the following ancillary data:

- Land Cover: CCI land cover [W5];
- Soil Texture: ISRIC soil texture [W6];
- NDVI mosaicked images derived from S-2 L2A data [W7].

A. Setting input files

a) SMOSAR_code/input_script_SMOSAR_processings.txt

```
track_dir=ApulianTav_S1_A146
date_from_proc=20180301
date_to_proc=20180403
string_processing=online offline make_hybrid
label_run=test1
list_img_combined=../SMOSAR_input_files/ApulianTav_S1_A146/inputs/list_img_combined.txt
rasterized_fields_map=../Maps_Products/raster_parcels/ApulianTav_S1_A146/italy_fully_merged_fields_2017
```

b) SMOSAR_input_files/input_SMOSAR_processor.txt

```
#INPUT_SMOSAR_PROCESSOR
Retrieval_algorithm          STCD online 1
Run_log_filename             input_SMOSAR_processor_STCD_ApulianTav_S1_A146_20180304
Parameter_N_VV_S-1           4
#S-1_AND_INC_IMG_INFO
S-1_IMG_samples              9442
S-1_IMG_lines                8536
S-1_IMG_PixelSize            0.0004
S-1_IMG_dir_path             ../S-1_Images/ApulianTav_S1_A146
S-1_IMG_filenameVV_1         S1A_IW_GRDH_1SDV_20180304T164905_MN02_RL4200.vv.GTC
S-1_IMG_filenameVH_1         S1A_IW_GRDH_1SDV_20180304T164905_MN02_RL4200.vh.GTC
S-1_IMG_filenameINC_1   reference_inc/ref_S1A_IW_GRDH_1SDV_20170414T164854_MN02_RL4200.vv.GTC.inc
S-1_IMG_date1                20180304
S-1_IMG_filenameVV_2         S1B_IW_GRDH_1SDV_20180310T164824_MN02_RL4200.vv.GTC
S-1_IMG_filenameVH_2         S1B_IW_GRDH_1SDV_20180310T164824_MN02_RL4200.vh.GTC
S-1_IMG_filenameINC_2   reference_inc/ref_S1A_IW_GRDH_1SDV_20170414T164854_MN02_RL4200.vv.GTC.inc
S-1_IMG_date2                20180310
S-1_IMG_filenameVV_3         S1A_IW_GRDH_1SDV_20180316T164905_MN02_RL4200.vv.GTC
S-1_IMG_filenameVH_3         S1A_IW_GRDH_1SDV_20180316T164905_MN02_RL4200.vh.GTC
S-1_IMG_filenameINC_3   reference_inc/ref_S1A_IW_GRDH_1SDV_20170414T164854_MN02_RL4200.vv.GTC.inc
S-1_IMG_date3                20180316
```

S-1_IMG_filenameVV_4 S1B_IW_GRDH_1SDV_20180322T164824_MN02_RL4200.vv.GTC
S-1_IMG_filenameVH_4 S1B_IW_GRDH_1SDV_20180322T164824_MN02_RL4200.vh.GTC
S-1_IMG_filenameINC_4 reference_inc/ref_S1A_IW_GRDH_1SDV_20170414T164854_MN02_RL4200.vv.GTC.inc
S-1_IMG_date4 20180322
#LANDCOVER_MAP_INFO
LC_MAP_samples 9442
LC_MAP_lines 8536
LC_MAP_PixelSize 0.0004
LC_MAP_dir_path ../Maps_LandCover/ApulianTav_S1_A146
LC_MAP_filename CCI_LCOVER_image_lc_sel.img
LC_MAP_date 20151231
LC_MAP_look_up_table_filename LC_lut.txt
LC_MAP_class_codes 1
#SOILTEXTURE_MAP_INFO
ST_MAP_samples 9442
ST_MAP_lines 8536
ST_MAP_PixelSize 0.0004
ST_MAP_dir_path ../Maps_SoilTexture/ApulianTav_S1_A146
ST_MAP_filename_clay isric_soil_texture_clay
ST_MAP_filename_sand isric_soil_texture_sand
#OUTPUT_SSM_FD_PROCESSOR
SSM_FD_MAP_dir_path ../Output_FD_SSM_maps/ApulianTav_S1_A146/STCD_run_test1
#SSM_FD_MAP_filenames:default_name_FD_MV
#OUTPUT_SSM_P_PROCESSOR
SSM_P_MAP_dir_path ../Output_P_SSM_maps/ApulianTav_S1_A146/STCD_run_test1

NB: this example contains the *input_SMOSAR_processor_20180304.txt* parameters and refers to the group of 4 S1 images (i.e. 20180304, 20180310, 20180316 and 20180322). The input files for the subsequent groups of 4 S1 images (i.e. *input_SMOSAR_processor_20180310.txt* and so on) are already prepared and they can be found in the folder:
SMOSAR_input_files/ ApulianTav_S1_A146/STCD_run.
The *script_SMOSAR_processings.bash* manages all these inputs making a copy of *input_SMOSAR_processor_yyyyddmm.txt* files into *input_SMOSAR_processor.txt* and starting each time a new processing (for both the *online* and *offline* processings).

B. Executing the software

The command is:

./script_SMOSAR_processings.bash

This command executes the processing chain (i.e. routines: 1) *SMOSAR_PROCESSOR_V2_0.sav* 2) *SMOSAR_MAPS_PREPROCESSING_V2_0.sav* and 3) *make_easy_hybrid_ssm_map.sh*) to obtain the temporal series of *P_SSM_hybrid maps* from 20180301 to 20180403, which are written into the output directory: *Output_P_SSM_maps/ApulianTav_S1_A146/STCD_run_test1_hybrid.*
The script automatically manages and updates the required intermediate input/output files.

Log files will be also written in the sub-directory of the folder:
SMOSAR_input_files/ApulianTav_S1_A146/STCD_run_test1.

Reference documents

[D3.5] Final operational SSM prototype exploiting S1&S2.
[D3.8] Final SSM Algorithm Theoretical Basis Document.

Web references

[W1] IDL – Interactive Data Language, © 2017 Exelis Visual Information Solutions, a subsidiary of Harris Corporation, http://www.harrisgeospatial.com/docs/using_idl_home.html
[W2] GRASS GIS - Geographic Resources Analysis Support System, https://grass.osgeo.org/#
[W3] ENVI – Envi image analysis software, EMVI https://www.harrisgeospatial.com/Software-Technology/ENVI
[W4] https://sentinel.esa.int/documents/247904/685163/Sentinel-1_User_Handbook
[W5[CCI land cover, https://www.esa-landcover-cci.org/
[W6] ISRIC soil texture, https://www.isric.org/
[W7] S-2 L2A data, https://earth.esa.int/web/sentinel/user-guides/sentinel-2-msi/product-types/level-2a.
[W8] GDAL tools - https://gdal.org/programs/index.html