





### The DMSAT-1 mission: primary instrument - Polarimeter characteristics and its earth observation applications

Alya AlMaazmi – Remote Sensing Applications Engineer

Mohammed Bin Rashid Space Center (MBRSC)





## MOTIVATION

**UAE DUSTY ATMOSPHERE** 

- Dust storms occur frequently in UAE, they most often hit during the summer and times of turbulent weather.
- Sandstorms typically contain silica crystals as well as viruses, bacteria, dust mites, fungi and even plant.
- Can cause vital problems linked to health, economy and environment.
- Space-based monitoring can supplement ground-based weather stations that are currently used for dust storm research.
- Map atmospheric aerosols, including their sources and transport, and study their influence within UAE.





## **DMSAT-1**



#### **Dubai Municipality Satellite**

Is a high-performance small microsatellite designed to perform multi-spectral multipolarization observations in visual and near-infrared bands, in addition to shortwave spectrum, for aerosol and greenhouse gas monitoring.





01 Mission Objectives	<b>Mission Objectives</b>		Investigations		Instruments
End-users tells us of what they want and need.	Map the distribution of atmospheric aerosols concentration (PM2.5, PM10) in the troposphere over Dubai land surface. Map greenhouse gases (CO2, CH4, H2O) concentration (ppm) over Dubai		Determine 2-D total column Aerosol Optical Depth (AOD) over Dubai urbanized areas at local noon.	/	Multi Spectral – Multi polarized Imager (Polarimeter)
02 Investigations Set of Constituents to be retrieved and			Determine total column Aerosol Effective Radius (AER) of PM2.5 and PM10 over Dubai urbanized areas at local noon.		Shortwave spectrometer 1 (1215 – 2000nm)
studied 03			Determine total column abundance and radiances of CO2, CH4, H2O over Dubai urbanized areas at local noon.		Shortwave spectrometer 2 (1730–2400nm)
Instruments					

Instrument design to aid the investigation





## **Instrument Design**

**Polarimeter instrument** 

- High transitivity telescope with a focal length of 150 mm and focal-length-to-aperture ratio of 2.8.
- Filter wheel used to divide the incoming light into three bands: 480-500 nm, 660-680 nm, and 860-880 nm.
- A "p-s" polarizing beam splitter, generates two identical images at two different polarizations 0° and 90°
- Kodak CCD detector at the focal plane.





## **Instrument Design**

**Spectral Bands and Polarization Measurements** 

- Chosen based on the mission's scientific requirement of retrieving aerosols properties.
- Each band has a polarization of 0° and 90°.
- The DMSAT-1 Polarimeter instrument is designed to measure the linearly polarized Earth-reflected radiance only.





## **Instrument Design**

**Geometry and spatial resolution** 

- --- Spatial resolution of the Polarimeter image is 43.8 m.
- --- Swath width of 107x38 km.
- Field of View (FOV) is 4.46 deg half diagonal, with an in-track look capability of less than 90° and an off-track look capability of 30°.





## **Operational Scenario**

730 km Altitude, SSO 12:00 LTDN

- Slewing maneuver at 7 different angles.
- Target observation at different reluctance and scattering angles.





#### **Primary Polarimeter Instrument Applications**

- Aerosol Optical Depth.
- Aerosol effective radius.
- Aerosol type PM 2.5.
- Aerosol type PM 10.

#### **Secondary Polarimeter Instrument Applications**

- Surface ALBEDO.
- Normalized Difference Vegetation Index (NDVI).
- Aerosol mass mixing ratio.

Credit : ICARE PARASOL Browse Products 01/OCt/2013

0.10

0.05

18 November 2019





**GRASP Algorithm for DMSAT-1 retrievals** 







**DMSAT-1** Calibrated

Image





Map Products

18 November 2019

www.mbrsc.ae

10







#### **GRASP Algorithm for DMSAT-1 retrievals**

INPUT	ALGORITHM	OUTPUT	
WEB APP	GRASP PROCESSING	KEPLER VIEWER	
User Input DMSAT-1	DMSAT-1 YAML and	Output format .CSV files,	
Image from processing	SDATA files, processing in	viewed in MBRSC Kepler	
	Ubuntu using Python	Viewer.	
	ubuntu®		

**SRASP** 

18 November 2019





GRASP

# **Polarimeter Applications**

#### **GRASP Algorithm for DMSAT-1 retrievals**



18 November 2019







### GRASP

### **Polarimeter Applications**

#### **GRASP Algorithm for DMSAT-1 retrievals**

now here is a second of the second of the second second of the second of	aliananalal - 🖉
10 Upport sys 11 sys.path += ["grasp-dir/src/python"]	Name * Type Size Value *
12 import grasp.code as pygrasp	ix_x int64 (4,) [1122]
14 import datetime	ly_y int64 (4,) [1212]
15 (aciat/augraph unrelan()) if to text if the interface works correctly	i tot i i
17	designed and the second s
10 19 data = datatime datatime(2013 1 2 11 0) /	pixet_number United (4,) [0 1 2 3]
20 S_Altitude = solar.CetAltitude(24.416666, 54.640678, date)	x_x float64 (4,) [54.64067841 54.64067841 53.64067841 53.64067841]
21 S_Azimuth = solar.GetAzimuth(24.416666, 54.640678, date) 22 S_Zenith = 90 , S_Altitude	x_x2 float64 (4,) [54.64067841 54.64067841 53.64067841 53.64067841]
<pre>23 T = print(S_Zenith, S_Arimuth, S_Altitude)</pre>	y_y float64 (4,) [24.416666 23.416666 23.416666 23.416666]
24	The second secon
<pre>26 results = pyprasp.run("examples/DMSAT/settings_example_DMSAT_inversion.yml", 27 sdata = pyprasp.SDATA("examples/DMSAT/DMSAT_MASDAR_4pixels.sdat")</pre>	📁 JUpyter Untitled1 Last Checkpork: a tew seconds ago (unsaved changes) 🥐 Logost
29 #colors = ["red", "green", "blue", "purple"]	File Edit View Insert Cell Kemel Widgets Help Trusted 🖌 Python 3 🗣
$30 \ \pi (= 0)$	5 + 3× (2) 85 + 4 HRun B C 19 Code + 00
32 # print(colors[1])	
33.8 (++)	
<pre>35 sdata.pixel[0].ix # coordinate x in the current cell, storting at 1 (in</pre>	In [*]: # -*- coding: wtf-8 -*-
36 sdata.pixel[0].ly # coordinate y in the current cell, starting at 1 (in 37 sdata.pixel[0].cleady # cloud flag: 0 = cloud, 1 = clear [a]	Spyder Editor
30 sdata.pixel[0].icol # line of the pixel in its original grid or database	This is a temporary script file.
<pre>39 sdata.pixel[0].irow # column of the pixel in its original grid or databas d0 sdata.pixel[0].x # logaltude of the pixel, in decimal degrees, in the</pre>	
41 sdata.pixel[0].y # Lotitude	# cont torget to work at grasp directory ( change the path to grasp directory) import sys
42 sdata.pixel[0].meas[0].sza #solar zenith angle in decinal degri 43 sdata.pixel[0].meas[1].sza #solar zenith angle in decinal dear	from Pysolar import solar # Pysolar not pysolar, use solar as function import distant imp
44 sdata.pixel[0].meas[2].sza #solor zenith angle in decimal depr	sys.path += ["grasp-dir/src/python"]
45	import grasp.code as pygrasp
47 Bcolors = ["red", "green", "blue", "purple"]	
AB #1 = 0 AB Bubile 1 < len(colors):	# solar angles calculator #date = datetime.datetime(2017,1,30,8,9) # ( year, month, day, hour, minute) # UTC hour time 84M =12 FM
50 # print(colors[i])	#5_Altitude = solar.GetAltitude(24.416565, 54.640678, date)
51 # 1 ++ 1 52	#2, AZIMUTH = SOIAT.UETAZIMUTH(24.410000, 54.0400/8, 0010) #5, Zenith = 90 · 5, Alitikude
53	
54 pixel_number = np.asarray((0,1,2,3)) 55 ix x = np.asarray((1,1,2,2))	
56 (y_y = rp.asarray([1,2,1,2])	print(pygrasp.version()) # to test if the interface works correctly results = overasp.cm('examples/DEAT/settings examples/DEAT inversion.vml", print screensTrue)
57 trow_row = np.asarray([1,1,2,2]) 50 trol_rol = np.asarray([1,2,1,2])	/ access SultA parameters
59 x_x = np.asarray([54.64067840576172, 54.64067840576172, 53.64067840576172, 53	#changing variables sdata = pvgraso_SDATA(*examples/DMSAT/DMSAT MASDAR 4oixels.sdat*)
<pre>60 y_y = np.asarray([24.416666, 23.416666, 24.416666, 23.416666]) 61</pre>	# sdata.pixet(0)
62	<pre>sorta.pixet[0].ix # coordinate x in the current cell, starting at 1 (in the direction B) isdata.pixel[0].ix # coordinate y in the current cell, starting at 1 (in the direction B)</pre>
$(3) \mathbf{j} = 0$	sdata.pixel[0].cloudy # cloud flag: 0 = cloud, 1 = clour, (a)
The MILLE L. S. LEINBLACK INFINEL /1	status, puzzetto, izou e come puzzet un its original grid or database status, puzzetto, i e columo of the pixel in its original grid or database
	sdata.pixel[0].x # longitude of the pixel, in decimal degrees, in the range [-180, 180] sdata.pixel[0].x # latitude of the pixel, in decimal degrees, in the range [-00, -180]
	sdata.dupo["sdata.save2.sdat"]
	<pre># fixed variables sdata_bixellol.masl # altitude of the ground, in metres (MKSL; metres above sea level)</pre>
	sdata.pixel[0].land_percent # percentage of land, in the range [0 (sea) 100 (land)]. Intermediate values correspond
	source_pixet[0].nut # number of available wavelengths (nut) source_pixel[0].neas[0].nut # wavelength 1, in micrometers
	sdata.pixel[0].meas[1].wl # unvelocity 7, in sicconsters
	source pixet(u).mess(z).vv  # wave(engr) , in micrometers solata.pixet(0).mess(a).np # number of types of measurements for wavelength 1 (nip)
	<pre>sdata_pixel[0].meas[1].nip # number of types of measurements for wavelength 1 (nip) sdata_pixel[0].meas[1].nip # number of types of measurements for wavelength 1 (nip)</pre>

18 November 2019





**GRASP Algorithm for DMSAT-1 retrievals** 



GRASP OPEN

**SRASP** 







**GRASP Algorithm for DMSAT-1 retrievals** 



**SRASP** 

18 November 2019





×





## **Polarimeter Applications**

#### **GRASP Algorithm for DMSAT-1 retrievals**



#### Export Data

Dataset								
Choose	the	datasets	yo					

DMSAT\_inversion\_output\_01\_17.csv

Data Type Choose the type of data



Filter Data

You can choose exporting origi data or filtered data



Cancel

18 November 2019

www.mbrsc.ae

16







**GRASP Algorithm for DMSAT-1 retrievals** 







18 November 2019





MOHAMMED BIN RASHID SPACE CENTRE

### Thank You

alya.almaazmi@mbrsc.ae