CERTIFICATE

This is to certified that the project report title "Meteorological satellite: sensing principles, data collection and analysis" is work done by Mr. YOUNKAP NINA Duplex, Air Traffic Controller Officer/Computer Science Engineer, IMD Advanced Training in Meteorological Instrumentation & Information System Batch No. IX, ICITC. Under my supervision.

I recommend the project report for submission.

Place: Maroua TCHAMABO Urbain
Date : 22|02|2022

ACKNOWLEDGEMENT

I am grateful to Shri S. SENGUPTA, Scientist-E, who has guided me on this project with his immense knowledge and experience.

I am grateful to officers and staffs of training section for the support extended throughout the training period

INTRODUCTION

The general concept behind remote sensing is to monitor Geographical, Biological and Physical characteristics of Land, Water and Atmosphere. It can offer a snap shot of state of the system at a particular point in time and 2D space. A combination of such "observations" gives valuable Information on the resource under study. These Information flow from analysts to managers to decision makers for successful interventions. In this report our focus is remote sensing for weather observation

Chapter 1

BASIC PRINCIPLES OF SATELLITE REMOTE SENSING

1.1 Definition

"Remote sensing is the science (and to some extent, art) of acquiring information about the Earth's surface without actually being in contact with it. This is done by sensing and recording reflected or emitted energy and processing, analysing, and applying that information." (f)

In much of remote sensing, the process involves an interaction between incident radiation and the targets of interest. This is exemplified by the use of imaging systems where the following seven steps are involved. However that remote sensing also involves the sensing of emitted energy and the use of non-imaging sensors.

Step 1-Energy Source or Illumination (A) : the first requirement for remote sensing is to have an energy source which illuminates or provides electromagnetic energy to the target of interest.

Step 2-bandRadiation and the Atmosphere (B) band : as the energy travels from its source to the target, it will come in contact with and interact with the atmosphere it passes through. This interaction may take place a second time as the energy travels from the target to the sensor.

Step 3-Interaction with the Target (C) : once the energy makes its way to the target through the atmosphere, it interacts with the target depending on the properties of both the target and the radiation.

Step 4-Recording of Energy by the Sensor (D) : after the energy has been scattered by, or emitted from the target, we require a sensor (remote - not in contact with the target) to collect and record the electromagnetic radiation.

Step 5-Transmission, Reception, and Processing (E) : the energy recorded by the sensor has to be transmitted, often in electronic form, to a receiving and processing station where the data are processed into an image (hardcopy and/or digital).

Step 6-Interpretation and Analysis (F) the processed image is interpreted, visually and/or digitally or electronically, to extract information about the target which was illuminated.

Step 7-Application (G) the final element of the remote sensing process is achieved when we apply the information we have been able to extract from the imagery about the target in order to better understand it, reveal some new information, or assist in solving a particular problem.

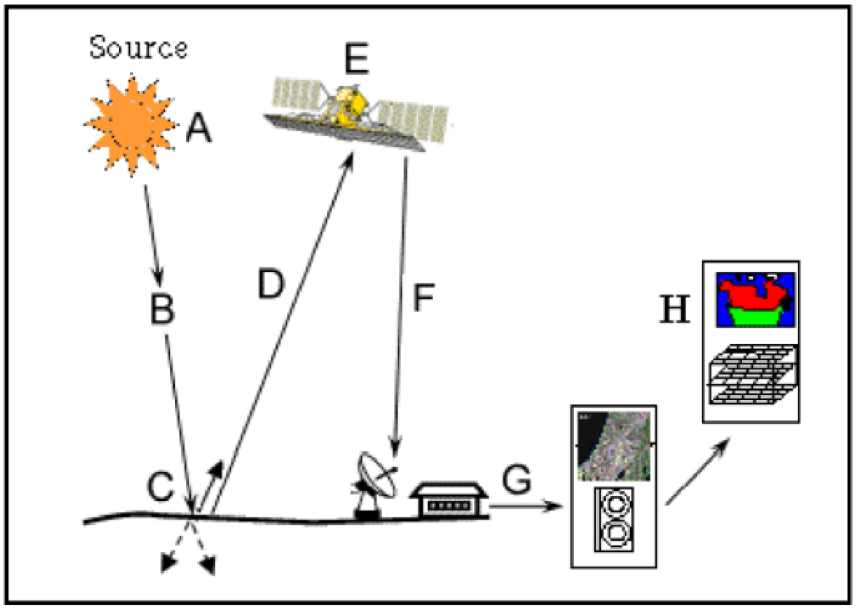


Figure 1.1: Remote sensing process

1.2 Electromagnetic Spectrum

Solar radiation can be divided into different wavelengths call EM spectrum:

1. Ultra violet (UV) [3-400 nm]
2. Visible (Vis) [0.4-0.7 µm]
3. Infrared Red (IR) [0.7-100 µm]
   (a) Reflected IR [0.7-3 µm]
      - Near IR [0.7-1.5 µm]
      - Shortwave IR [1.5-3 µm]
   (b) Emitted IR or TIR [3-100 µm]
      - Midwave IR [3-8 µm]
      - Longwave IR [8-15 µm]

- Far IR [15-100 µm]

4. Microwave (MW) [1 mm – 1m]

   P band : 0.3 - 1 GHz (30 - 100 cm)

   L band : 1 - 2 GHz (15 - 30 cm)

   S band : 2 - 4 GHz (7.5 - 15 cm)

   C band : 4 - 8 GHz (3.8 - 7.5 cm)

   X band : 8 - 12.5 GHz (2.4 - 3.8 cm)

   Ku band : 12.5 - 18 GHz (1.7 - 2.4 cm)

   K band : 18 - 26.5 GHz (1.1 - 1.7 cm)

   Ka band : 26.5 - 40 GHz (0.75 - 1.1 cm)

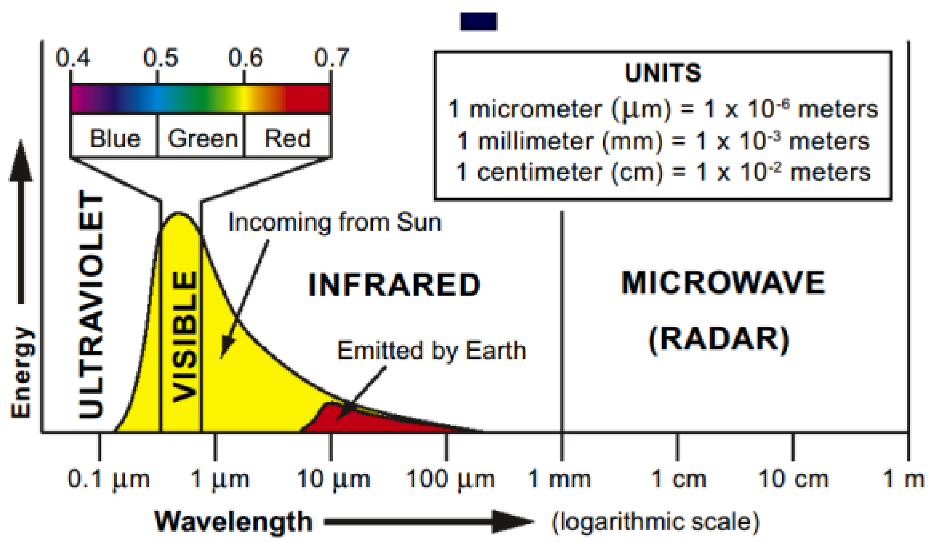


Figure 1.2: Electromagnetic spectrum

Most Multispectral satellite can acquire data in visible and infrared region (including the thermal infrared region). Now days there is an increasing demand for data in microwaves region, because the microwave energy can penetrate through the cloud.

1.3 Source of radiation

We have two majors source of radiation: Sun and Earth.

Sun is the mayor source of radiation with highest energy in visible, near infrared and short-wave infrared region as shown on picture above.

Earth surface can also emit energy usually in thermal Infrared and microwaves region.

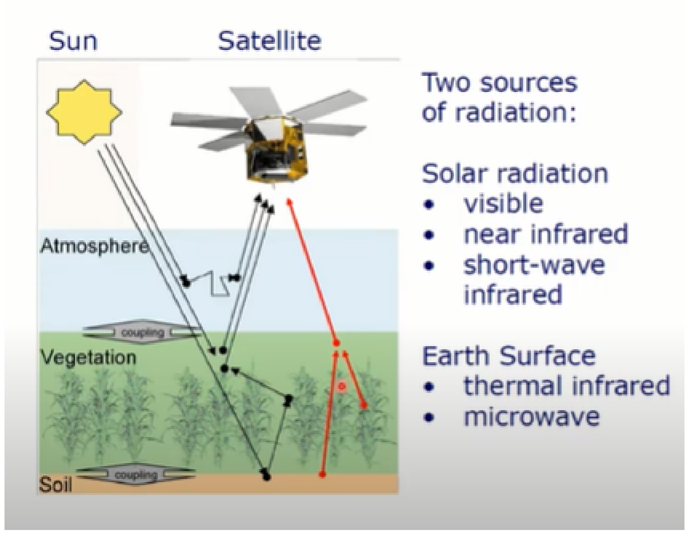


Figure 1.3: Source of radiation

1.4 Interaction in the atmosphere

Before radiation used for remote sensing reaches the Earth's surface it has to travel through some distance of the Earth's atmosphere. Different processes happening due to the presence of particles and gases in the atmosphere. Some get **scattered**, some get **absorbed**, some managed to transmit through the atmosphere and get **absorbed** by the earth, and some get **reflected** . This reflected energy is capture by the sensor on board of satellite. They are some energies emitted from the earth which are directly proportional to the heat of that particular object. This emitted energy can also be recorded by sensor on board of satellite if that sensor has capabilities to record energy in thermal infrared region. A sensor on board a satellite can record **remitted** or **reflected** radiation depending on the capabilities of the sensors.

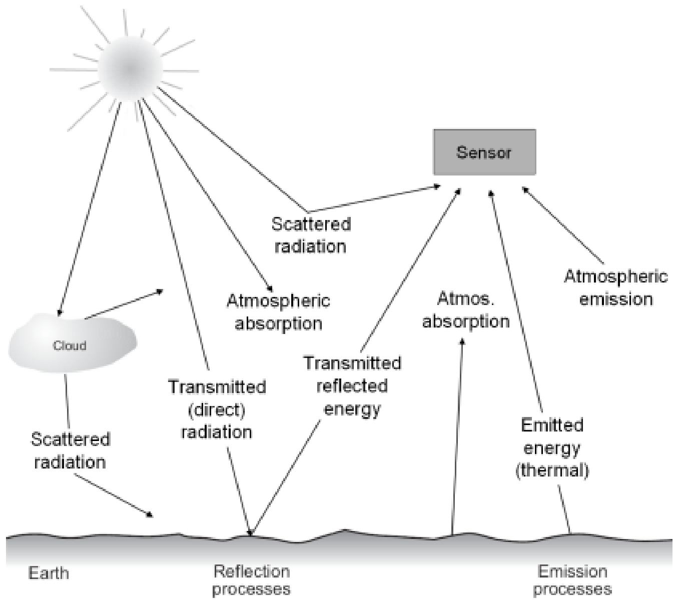


Figure 1.4: radiance at the sensor

1.5 How does it work ?

As we said above, the energy is either emitted or reflected. And this emission or reflection on energy depend on he targets of interest e.g. cloud, trees, water, etc. They all have different level of reflectance, and this radiation of energy from different targets type will be recorded by the sensor installed in the satellite. Many time what is record by satellite is directly proportional to top of the atmosphere (TOA) "level at which the effect of absorption and scattering is not relevant". The energy recorded at sensor is proportional to what could be at the top of atmosphere e.g. temperature and humidity recorded is equal to brightness temperature (TOA Temperature), and we have to apply some correction convert brightness temperature to surface temperature. The same go to reflectance, we have to convert TOA radiance energy to surface reflectance energy. This process is call atmospheric correction. This how satellite records energy and how it correspond to different targets of interest.

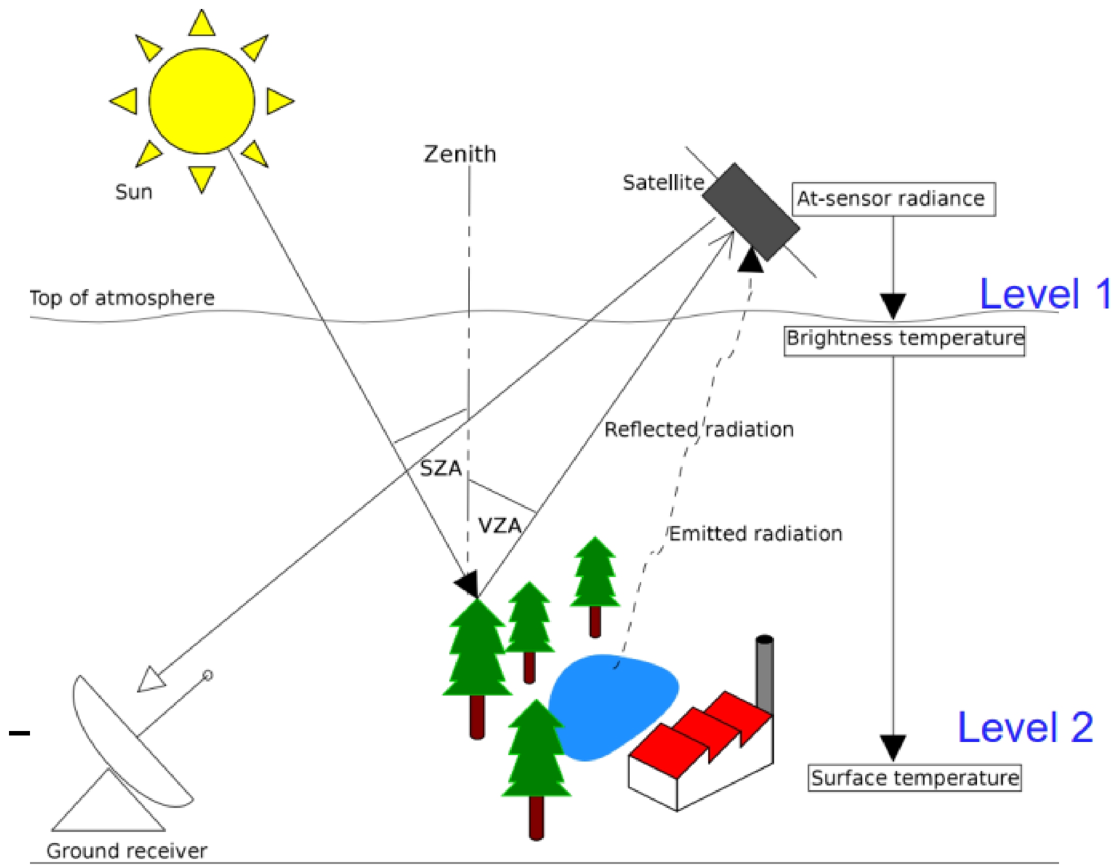


Figure 1.5: Sattelite remote seinsing

1.6 Category of satellite

There are different category of satellite based on:

Source of energy : passive and active Sensors.

Orbit : polar, geo-stationary and sun-synchronous.

Resolutions : spatial, spectral, radiometric and temporal.

Objectives : communication, weather, navigation, earth observation, etc.

Chapter 2

CASE STUDY: GOES-R Series

2.1 GOES project current status

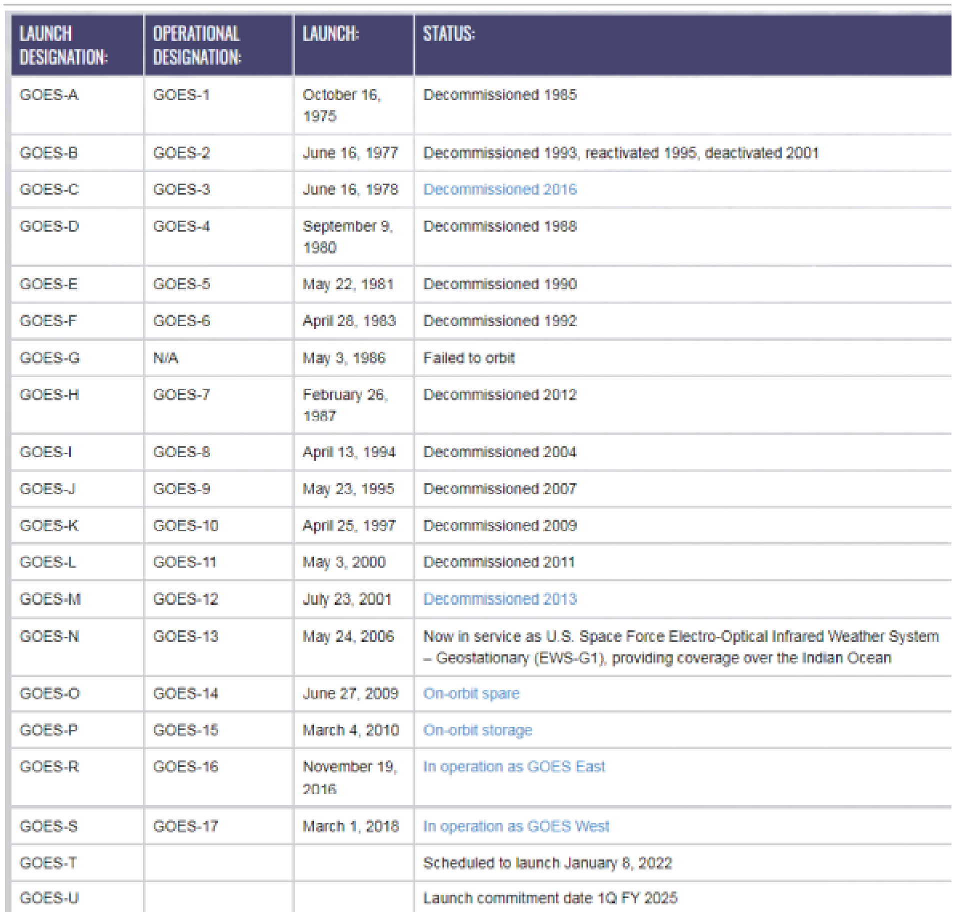
The Geostationary Operational Environmental Satellite Program (GOES) is a joint effort of NASA and the National Oceanic and Atmospheric Administration (NOAA). The GOES system currently consists of GOES-13, operating as GOES-East, in the eastern part of the constellation at 75 degrees west longitude and GOES-15, operating as GOES-West, at 135 degrees west longitude. The GOES-R series will maintain the two-satellite system implemented by the current GOES series. However, the locations of the operational GOES-R satellites will be 75 degrees west longitude and 137 degrees west longitude. The latter is a shift in order to eliminate conflicts with other satellite systems. The GOES-R series operational lifetime extends through December 2036. These spacecraft help meteorologists observe and predict local weather events, including thunderstorms, tornadoes, fog, hurricanes, flash floods and other severe weather. In addition, GOES observations have proven helpful in monitoring dust storms, volcanic eruptions and forest fires.The benefits that directly enhance the quality of human life and protection of Earth's environment include:

- Supporting the search-and-rescue satellite aided system (SARSAT).
- Contributing to the development of worldwide environmental warning services and enhancements of basic environmental services.
- Improving the capability for forecasting and providing real-time warning of solar disturbances.
- Providing data that may be used to extend knowledge and understanding of the atmosphere and its processes.

The next series of GOES satellites includes GOES-R, S, T and U.

Figure 2.1: Goes-R Series

2.2 GOES Project History



LAUNCH DESIGNATION:	OPERATIONAL DESIGNATION:	LAUNCH:	STATUS:
GOES-A	GOES-1	October 16, 1975	Decommissioned 1985
GOES-B	GOES-2	June 16, 1977	Decommissioned 1993, reactivated 1995, deactivated 2001
GOES-C	GOES-3	June 16, 1978	Decommissioned 2016
GOES-D	GOES-4	September 9, 1980	Decommissioned 1988
GOES-E	GOES-5	May 22, 1981	Decommissioned 1990
GOES-F	GOES-6	April 28, 1983	Decommissioned 1992
GOES-G	N/A	May 3, 1986	Failed to orbit
GOES-H	GOES-7	February 26, 1987	Decommissioned 2012
GOES-I	GOES-8	April 13, 1994	Decommissioned 2004
GOES-J	GOES-9	May 23, 1995	Decommissioned 2007
GOES-K	GOES-10	April 25, 1997	Decommissioned 2009
GOES-L	GOES-11	May 3, 2000	Decommissioned 2011
GOES-M	GOES-12	July 23, 2001	Decommissioned 2013
GOES-N	GOES-13	May 24, 2006	Now in service as U.S. Space Force Electro-Optical Infrared Weather System – Geostationary (EWS-G1), providing coverage over the Indian Ocean
GOES-O	GOES-14	June 27, 2009	On-orbit spare
GOES-P	GOES-15	March 4, 2010	On-orbit storage
GOES-R	GOES-16	November 19, 2016	In operation as GOES East
GOES-S	GOES-17	March 1, 2018	In operation as GOES West
GOES-T			Scheduled to launch January 8, 2022
GOES-U			Launch commitment date 1Q FY 2025

Figure 2.2: Goes-R Series

2.3 GOES-R Series Instruments

GOES-R Series has six instruments on board:

Earth-pointing :

- Advanced Baseline Imager (ABI) - the primary instrument for imaging Earth's weather, oceans and environment.
- Geostationary Lightning Mapper (GLM) - a single-channel, near-infrared optical transient detector that can identify momentary changes in an optical scene, indicating the presence of lightning.

Sun-pointing :

- Extreme Ultraviolet and X-ray Irradiance Sensors (EXIS) - monitors solar irradiance in the upper atmosphere using two primary sensors: the Extreme Ultraviolet Sensor (EUVS) and the X-Ray Sensor (XRS).

- Solar Ultraviolet Imager (SUVI) - a telescope that monitors the sun in the extreme ultraviolet wavelength range, detecting solar flares and solar eruptions, and compiling full disk solar images.

In-situ :

- Magnetometer (MAG) - measures the space environment magnetic field that controls charged particle dynamics in the outer region of the magnetosphere
- Space Environment In-Situ Suite (SEISS) - monitors proton, electron, and heavy ion fluxes in the magnetosphere using four sensors: the Energetic Heavy Ion Sensor (EHIS), the High and Low Magnetospheric Particle Sensors (MPS-HI and MPS-LO), and the Solar and Galactic Proton Sensor (SGPS).

Only two sensors are used for atmospheric purpose, Advanced Baseline Imager (ABI) and Geostationary Lightning Mapper (GLM).

2.4 Advanced Baseline Imager (ABI)

ABI is a multi-channel passive imaging radiometer that images Earth's weather, oceans and environment with 16 spectral bands (2 visible, 4 near-infrared, and 10 infrared channels).

ABI Band	Central Wavelength(μm)	Type	Nickname	Best Spatial Resolution
1	0.47	Visible	Blue	1
2	0.64	Visible	Red	0.5
3	0.86	Near-Infrared	Veggie	1
4	1.37	Near-Infrared	Cirrus	2
5	1.6	Near-Infrared	Snow/Ice	1
6	2.2	Near-Infrared	Cloud particle size	2
7	3.9	Infrared	Shortwave window	2
8	6.2	Infrared	Upper-level water vapor	2
9	6.9	Infrared	Midlevel water vapor	2
10	7.3	Infrared	Lower-level water vapor	2
11	8.4	Infrared	Cloud-top phase	2
12	9.6	Infrared	Ozone	2
13	10.3	Infrared	"Clean" longwave window	2
14	11.2	Infrared	Longwave window	2
15	12.3	Infrared	"Dirty" longwave window	2
16	13.3	Infrared	CO_2 longwave	2

Schmit, T. J., S. S. Lindstrom, J. J. Gerth, M. M. Gunshor, 2018. Applications of the 16 spectral bands on the Advanced Baseline Imager (ABI) J. Operational Meteor., 6 (4), 33-46.

Figure 2.3: ABI optical components

2.5 ABI Data Collection Approach and Operations

There are three major components of ABI that work together to collect observations of the Earth, space, and calibration targets as a system. These are the scanning mirrors, the Four Mirror Anastigmat (FMA) telescope, and the Focal Plane Modules (FPMs).

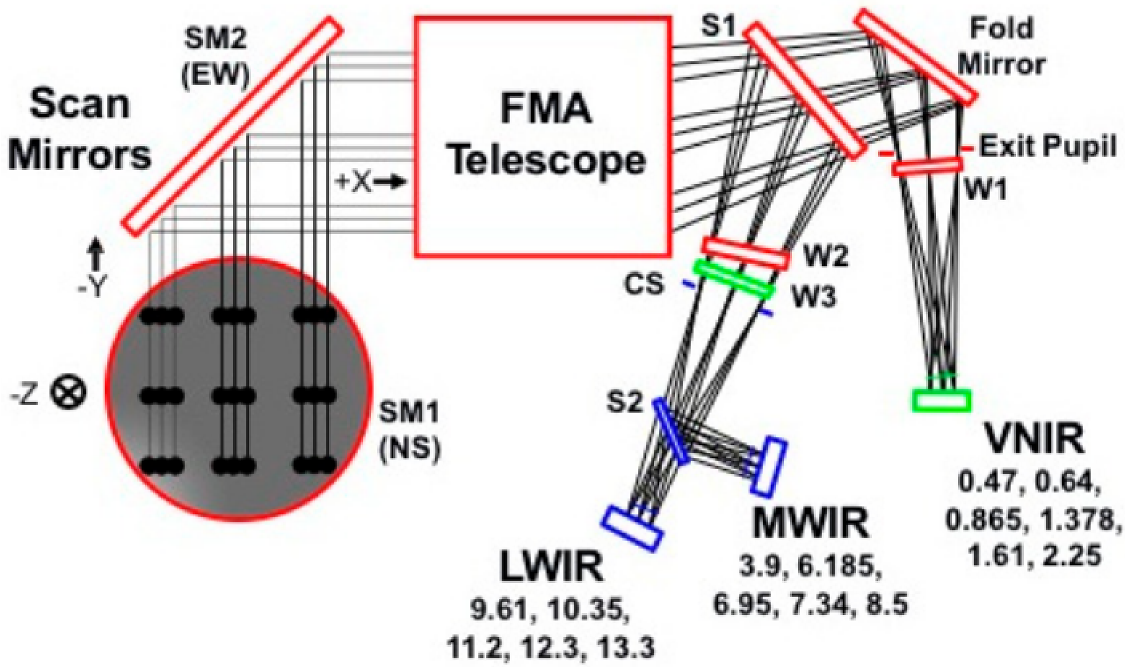


Figure 2.4: ABI optical components

SM=Scan Mirror	**EW**=East–West	**NS**=North–South
FMA=Four Mirror Anastigmat	**S1, S2**=Beamsplitter 1 and 2	**CS**=Cold Stop
W1, W2, W3=Windows 1, 2, and 3	**LWIR**=Longwave Infrared	**VNIR** =Visible and Near I
MWIR=Mid-Wave IR		

2.6 ABI modes of operation

Full Disk : Hemispheric Coverage of 83° local zenith angle, temporal resolution of 5-15 minutes, and spatial resolution of 0.5 to 2km.

Mesoscale : Provides coverage over a 1000x1000km box with a temporal resolution of 30 seconds, and spatial resolution of 0.5 to 2km.

Continental US/Pacific US : The CONUS and PACUS scans are performed every five minutes, providing coverage of the 5000km (east/west) and 3000km (north/south) rectangle over the continental United States (GOES-16) or the Pacific Ocean, including Hawaii (GOES-17). The spatial resolution is 0.5 to 2km.

Flex Modes : he flex modes provide a full disk scan every 10 minutes (mode 6) or every 15 minutes (mode 3), a CONUS/PACUS every five minutes, and two mesoscale domains every 60 seconds (or one sub-region every 30 seconds).

2.7 GOES-R Data Products

GOES-R Data products have three levels:

Level 0 (L0) : are observation data received directly from the 6 satellite instruments. The data is not meaningful to most users prior to processing by the ground system.

Level 1b (L1b) : are calibrated and, where applicable, geographically corrected, L0 data. This means that the data has been processed so that its values are in standard units of physical quantities. For ABI, the L1b product is Radiances. This is useful for users who require radiance units, instead of reflectance/brightness temperature (Kelvin) units. All of the instruments have L1b products available except GLM, which is only distributed as an L2+ product.

Level 2+ Products : contain environmental physical qualities, such as cloud top height or land surface temperature. Aside from the GLM Lightning Detection Product, the data source for these products is the ABI L1b data.

2.8 ABI meteorological product

The mission-critical ABI product is Cloud and Moisture Imagery (CMI), which utilizes all 16 ABI spectral bands, and is used to generate an array of products aiding forecasters in monitoring and predicting weather hazards. The following is a list ABI meteorological and space weather data products that are available to the user community.

- Aerosol detection (including smoke and dust)
- Aerosol optical depth (AOD)
- Aerosol particle size
- Clear sky masks
- Cloud layers/heights
- Cloud and moisture imagery

- Cloud optical depth
- Cloud particle size distribution
- Cloud top height
- Cloud top phase
- Cloud top pressure
- Cloud top temperature
- Derived motion winds
- Derived stability indices
- Downward shortwave radiation: surface
- Fire/hot spot characterization
- Hurricane intensity estimation
- Land surface albedo
- Land surface bidirectional reflectance factor
- Land surface temperature (skin)
- Legacy vertical moisture profile
- Legacy vertical temperature profile
- Radiances
- Rainfall rate/QPE
- Reflected shortwave radiation: TOA
- Sea and lake ice: age
- Sea and lake ice: concentration
- Sea and lake ice: motion
- Sea surface temperature (skin)
- Snow cover
- Total precipitable water
- Volcanic ash: detection and height

ls

Chapter 3

SATELLITE DATA ANALYSIS

3.1 Technique 1: access data files

3.1.1 Access data files from NOAA CLASS

- The NOAA Comprehensive Large Array-data Stewardship System (CLASS) repository is the official site for accessing all available GOES-R Series Products.

- A similar but easier website to navigate is NCEI's Archive Archive Information Request System (AIRS).

3.1.2 Access data files from Amazon, Microsoft, OCC

- Amazon Web Service (AWS) - ABI L1b and L2+, GLM L2+, and SUVI L1b products are available in AWS S3 Buckets. These open datasets can be accessed by the public from AWS for free.

- Microsoft Azure - Two GOES-16 ABI Full Disk products are stored in a Azure blob container. The products currently available are L1b Radiances and L2+ MCMI, which can be accessed from Azure for free.

- Open Commons Consortium (OCC) - Stores a 100 TB rolling archive of GOES-16 data (˜8 months), the products stored are ABI L1b and ABI L2+ CMI and MCMI. OCC recommends using the AWS CLI or the python boto library, to access the data.

3.1.3 Access data files from Google Cloud

- Google Cloud - ABI L1b and L2+, GLM L2+, and SUVI L1b products are available in two different buckets for GOES-16 and GOES-17.

- Google Earth Engine (GEE) - A cloud-based platform for geospatial analysis. This service runs through Google Cloud, and pulls GOES-R datasets from the Google Cloud buckets.

3.1.4 Use Python to retrieve data from AWS

Python is the most powerful programming language using in data science and geospatial data analysis. The following sample scripts demonstrate how to retrieve GOES-R files from AWS using Python.

```python
from goespy.Downloader import ABI_Downloader
### to use ABI_Downloader, you need 7 arguments:
import datetime as dt
### Getting the current date (in UTC coordinate)
utcDateTime = dt.datetime.utcnow()
## current year
year = utcDateTime.strftime("%Y")
# current month
month = utcDateTime.strftime("%m")
## current day
day = utcDateTime.strftime("%d")
## current hour in UTC
hour = utcDateTime.strftime("%H")
##Choose a channel from your preference (can be C01-C16)
channel = ["C01"]
## In GOES satellite they have 9 products
## 3 are L1b-Rad(M,C,F)
## 3 are L2-CMIP(M,C,F)
## 3 are L2-MCMIP(M,C,F)
### In your case we will get the CMIPF, F means FullDisk (all the projection by
    the satellite)
product = 'L1b-L2-CMIPF'
## The Bucket is the variable contains the name of dataset server from goes on
   AWS
Bucket = 'noaa-goes16' ## in the future on AWS they will have goes17.
## Now we will call the function ABI_Downloader:
Abi = ABI_Downloader('noaa-goes16',year,month,day,hour,product,channel)
```

After all the dataset is downloaded, they are in your home directory with that structure:

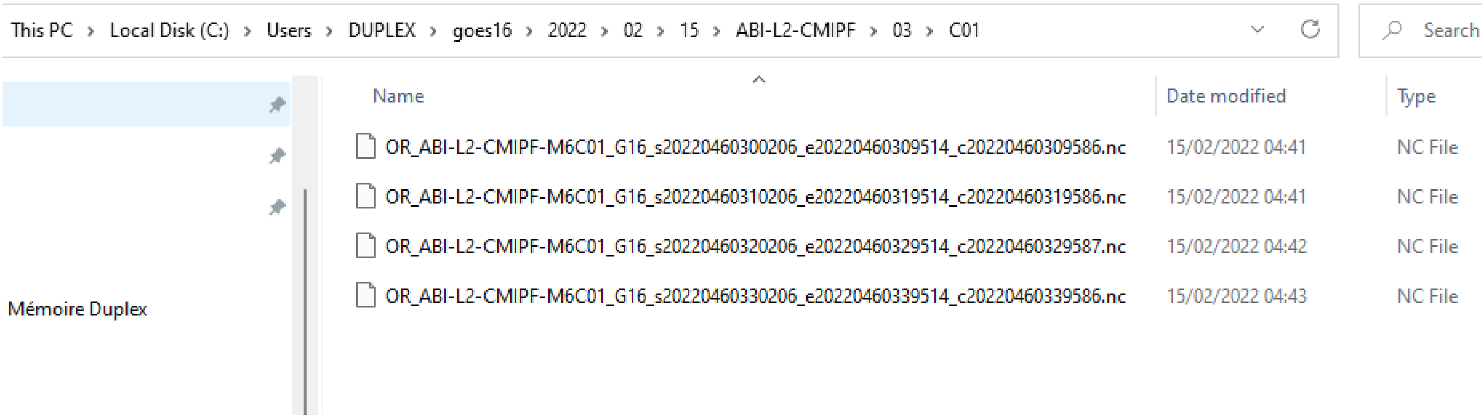


Figure 3.1: L1b-L2-CMIPF product downloaded

3.2 Technique 2: Explore satellite data and metadata

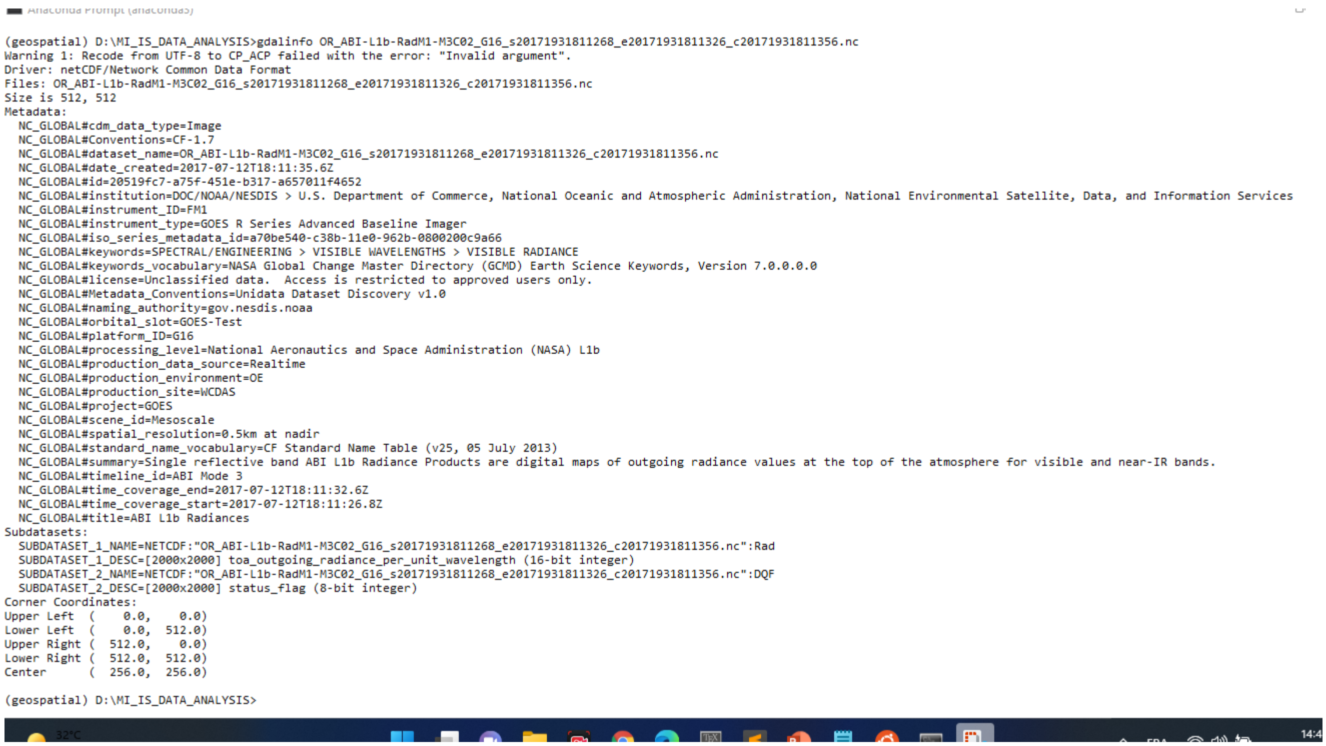
3.2.1 GOES-R Series data format

GOES-R Series product files use the netCDF-4 format. NetCDF (Network Common Data Form) is a set of software libraries and machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data. It is also a community standard for sharing scientific data.

3.2.2 Read metadata

Metadata provides information about the distinct items, such as: means of creation, purpose of the data, time and date of creation, creator or author of data, placement on a network (electronic form) where the data was created, what standards used etc. The main purpose of metadata is to facilitate in the discovery of relevant information, more often classified as resource discovery. Metadata also helps organize electronic resources, provide digital identification, and helps support archiving and preservation of the resource. Before any manipulation of data, it is vitaly important to read metadata. Now let us see how to read the metadata of this satellite data file in command line using gdal tools.

```
## change directory to where you have downloaded the GLDAS data using the below
cd /mnt/d/MI_IS_DATA_ANALYSIS
## extatract metadata of the 'OR_ABI-L1b-RadM1-
   M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc'
gdalinfo OR_ABI-L1b-RadM1-
   M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc
## press enter
```



```
Anaconda Prompt (anaconda3)

(geospatial) D:\MI_IS_DATA_ANALYSIS>gdalinfo OR_ABI-L1b-RadM1-M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc
Warning 1: Recode from UTF-8 to CP_ACP failed with the error: "Invalid argument".
Driver: netCDF/Network Common Data Format
Files: OR_ABI-L1b-RadM1-M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc
Size is 512, 512
Metadata:
  NC_GLOBAL#cdm_data_type=Image
  NC_GLOBAL#Conventions=CF-1.7
  NC_GLOBAL#dataset_name=OR_ABI-L1b-RadM1-M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc
  NC_GLOBAL#date_created=2017-07-12T18:11:35.6Z
  NC_GLOBAL#id=20519fc7-a75f-451e-b317-a657011f4652
  NC_GLOBAL#institution=DOC/NOAA/NESDIS > U.S. Department of Commerce, National Oceanic and Atmospheric Administration, National Environmental Satellite, Data, and Information Services
  NC_GLOBAL#instrument_ID=FM1
  NC_GLOBAL#instrument_type=GOES R Series Advanced Baseline Imager
  NC_GLOBAL#iso_series_metadata_id=a70be540-c38b-11e0-962b-0800200c9a66
  NC_GLOBAL#keywords=SPECTRAL/ENGINEERING > VISIBLE WAVELENGTHS > VISIBLE RADIANCE
  NC_GLOBAL#keywords_vocabulary=NASA Global Change Master Directory (GCMD) Earth Science Keywords, Version 7.0.0.0.0
  NC_GLOBAL#license=Unclassified data.  Access is restricted to approved users only.
  NC_GLOBAL#Metadata_Conventions=Unidata Dataset Discovery v1.0
  NC_GLOBAL#naming_authority=gov.nesdis.noaa
  NC_GLOBAL#orbital_slot=GOES-Test
  NC_GLOBAL#platform_ID=G16
  NC_GLOBAL#processing_level=National Aeronautics and Space Administration (NASA) L1b
  NC_GLOBAL#production_data_source=Realtime
  NC_GLOBAL#production_environment=OE
  NC_GLOBAL#production_site=WCDAS
  NC_GLOBAL#project=GOES
  NC_GLOBAL#scene_id=Mesoscale
  NC_GLOBAL#spatial_resolution=0.5km at nadir
  NC_GLOBAL#standard_name_vocabulary=CF Standard Name Table (v25, 05 July 2013)
  NC_GLOBAL#summary=Single reflective band ABI L1b Radiance Products are digital maps of outgoing radiance values at the top of the atmosphere for visible and near-IR bands.
  NC_GLOBAL#timeline_id=ABI Mode 3
  NC_GLOBAL#time_coverage_end=2017-07-12T18:11:32.6Z
  NC_GLOBAL#time_coverage_start=2017-07-12T18:11:26.8Z
  NC_GLOBAL#title=ABI L1b Radiances
Subdatasets:
  SUBDATASET_1_NAME=NETCDF:"OR_ABI-L1b-RadM1-M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc":Rad
  SUBDATASET_1_DESC=[2000x2000] toa_outgoing_radiance_per_unit_wavelength (16-bit integer)
  SUBDATASET_2_NAME=NETCDF:"OR_ABI-L1b-RadM1-M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc":DQF
  SUBDATASET_2_DESC=[2000x2000] status_flag (8-bit integer)
Corner Coordinates:
Upper Left  (    0.0,    0.0)
Lower Left  (    0.0,  512.0)
Upper Right (  512.0,    0.0)
Lower Right (  512.0,  512.0)
Center      (  256.0,  256.0)

(geospatial) D:\MI_IS_DATA_ANALYSIS>
```

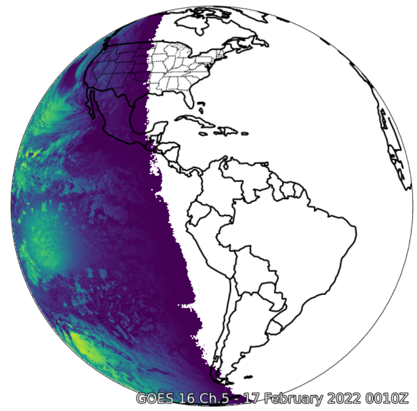
Figure 3.2: metadata

3.3 Technique 3: Plot a single band ABI Channel

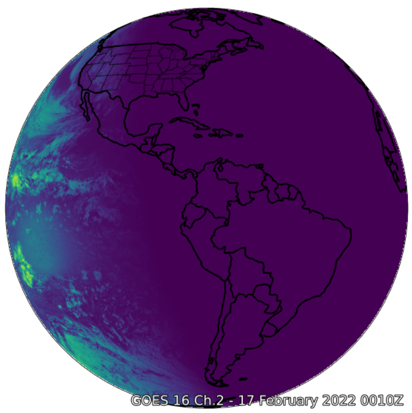
ABI imager has 16 channels, the code below show how to plot a single band ABI channel.

```python
def open_dataset(date, channel, idx, region):
    """
    Open and return a netCDF Dataset object for a given date, channel, and image index
    of GOES-16 data from THREDDS test server.
    """
    cat = TDSCatalog('https://thredds.ucar.edu/thredds/catalog/satellite/goes/east/products/'
                     f'CloudAndMoistureImagery/{region}/Channel{channel:02d}/{date:%Y%m%d}/catalog.xml')
    ds = cat.datasets[idx]
    ds = ds.remote_access(use_xarray=True)
    return ds
def plot_GOES16_channel(date, idx, channel, region):
    """
    Get and plot a GOES 16 data band from the ABI.
    """
    ds = open_dataset(date, channel, idx, region)
    dat = ds.metpy.parse_cf('Sectorized_CMI')
    proj = dat.metpy.cartopy_crs
    x = dat['x']
```

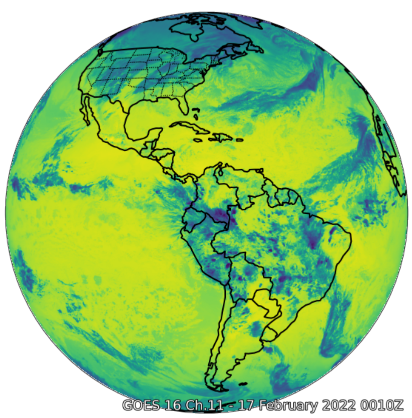
```
    y = dat['y']
    fig = plt.figure(figsize=(10, 10))
    ax = fig.add_subplot(1, 1, 1, projection=proj)
    ax.add_feature(cfeature.COASTLINE, linewidth=2)
    ax.add_feature(cfeature.STATES, linestyle=':', edgecolor='black')
    ax.add_feature(cfeature.BORDERS, linewidth=2, edgecolor='black')
    for im in ax.images:
        im.remove()
    im = ax.imshow(dat, extent=(x.min(), x.max(), y.min(), y.max()), origin='upper')
    timestamp = datetime.strptime(ds.start_date_time, '%Y%j%H%M%S')
    add_timestamp(ax, time=timestamp, high_contrast=True,
                  pretext=f'GOES 16 Ch.{channel} - ',
                  time_format='%d %B %Y %H%MZ', y=0.01,
                  fontsize=18)
    display(fig)
    plt.savefig("goes_fulldisk_C1.png")
    plt.close()
channel_list = {u'1 - Blue Band 0.47 \u03BCm': 1,
                u'2 - Red Band 0.64 \u03BCm': 2,
                u'3 - Veggie Band 0.86 \u03BCm': 3,
                u'4 - Cirrus Band 1.37 \u03BCm': 4,
                u'5 - Snow/Ice Band 1.6 \u03BCm': 5,
                u'6 - Cloud Particle Size Band 2.2 \u03BCm': 6,
                u'7 - Shortwave Window Band 3.9 \u03BCm': 7,
                u'8 - Upper-Level Tropo. WV Band 6.2 \u03BCm': 8,
                u'9 - Mid-Level Tropo. WV Band 6.9 \u03BCm': 9,
                u'10 - Low-Level WV Band 7.3 \u03BCm': 10,
                u'11 - Cloud-Top Phase Band 8.4 \u03BCm': 11,
                u'12 - Ozone Band 9.6 \u03BCm': 12,
                u'13 - Clean IR Longwave Band 10.3 \u03BCm': 13,
                u'14 - IR Longwave Band 11.2 \u03BCm': 14,
                u'15 - Dirty Longwave Band 12.3 \u03BCm': 15,
                u'16 - CO2 Longwave IR 13.3 \u03BCm': 16}
region = Select(
    options=['Mesoscale-1', 'Mesoscale-2', 'CONUS', 'PuertoRico', 'FullDisk'],
    description='Region:',)

channel = Dropdown(options=channel_list,value=16,description='Channel:',)

interact(plot_GOES16_channel, date=fixed(date), idx=fixed(-2),
         channel=channel, region=region)
```



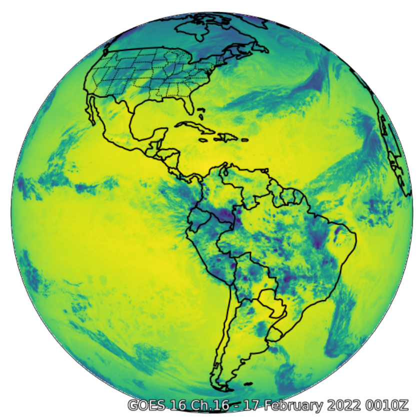
(b) ABI CH05



(a) ABI CH01



(c) ABI CH11



(d) ABI CH16

Figure 3.3: ABI BAND SAMPLE PLOT

3.4 Technique 4: Convert NetCDF files to GeoTIFFs

GeoTIFFS are regarded as the industry standard for georeferenced raster data. The conversion of GOES-R netCDF arrays to single-band GeoTIFF rasters can be accomplished in command line. This conversion an important stop of the analysis, so satellite data when conveted in GeoTIFFS can easily be injected in GIS software to automate workflows.

```
##  Convert radiance from NetCDF files to GeoTIFF - below command
```

```
gdal_translate NETCDF:"OR_ABI-L1b-RadM1-M3C02_G16_s20171931811268_e20171931811326_c20171931811356.nc":OR_ABI-L1b-RadM1-M3C02_G16_s20171931811268_e20171931811326_c20171931811356.tif
```

3.5 Technique 5: Processing satellite data using GIS Software

3.5.1 Objectives

The main objective of this analysis is to process satellite data and compare with data obtained from at aeronautical met stations. Here we will use the three hour data from GLDAS.

3.5.2 Avition meteorological stations in cameroon

Table 3.1: Aviation meteorological stations

NAME	OMM CODE	ICAO CODE	LATITUDE	LONGITUDE
Yaounde	64950	FKYS	03°50N	011°31E
Douala	64910	FKKD	04°00N	009°44E
Garoua	64860	FKKR	09°20N	013°23E
Maroua	64851	FKKL	10°27'N	14°15'E
Ngaoundéré	64870	FKKN	07°21N	013°34E
Bafoussam	64894	FKKU	05°32'05 N	010°21'15 E
Bertoua	64930	FKKO	04°36N	013°44E
Bamenda	64892	FKKV	06°03N	010°07E

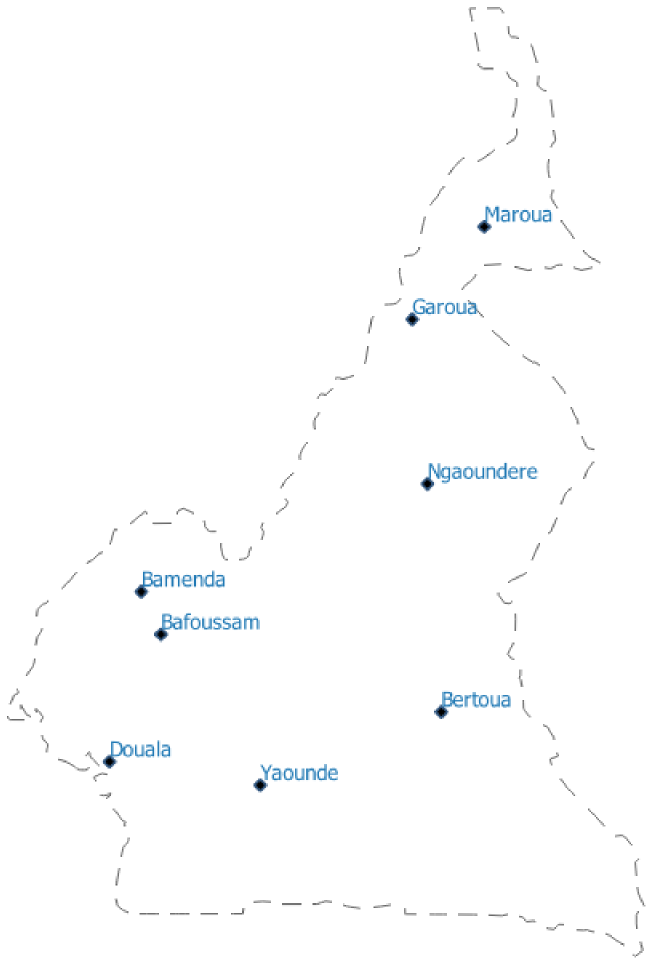


Figure 3.4: aviation meteorological stations

3.5.3 GLDAS data

The goal of the Global Land Data Assimilation System (GLDAS) is to ingest satellite- and ground-based observational data products, using advanced land surface modeling and data assimilation techniques, in order to generate optimal fields of land surface states and fluxes (Rodell et al., 2004a). GLDAS data are available to download from this link. Detailed documentation about GLDAS 2.1 product is available here

3.5.4 Specific requirements

The requirements for the metereorological data at aeronautical station are:

Parameter	Symbols	Unit
Downward shortwave radiation	SWdown	W/m^2
Wind speed	Ws	m/s
Air temperature	Tair	°C
Pressure	P	Mb
Relative humidity	Rh	%

Figure 3.5: meteorological data required

The units at which GLDAS provide air temperature (K) and Pressure(Pa). Further GLDAS provide specific humidity. To match requirements the following conversion steps have to be performed:

Step 1 : Convert GLDAS air temperature from Kelvin to Deg C.

Step 2 : Convert the unit of GLDAS pressure from Pa to Milli bar (Mb).

Step 3 : Convert specific humidity to relative humidity following the description here.

Step 4 : Extract the Ws, Tair, P and Rh corresponding at our aeronautical meteorological stations at observation time : 00:00, O3:00, O6:00, 09:00, 12:00, 15:00, 18:00 and 21:00 UTC.

Step 5 : Compare the results with data collected at the stations.

3.5.5 Download GLDAS data

To download GLDAS data, we can use python script as shown below.

```
## In the Ubuntu session.
## Type in the following command to install the python package "gldas"
pip3 install gldas
## press enter
## Below code will download the GLDAS data between the dates provided in the
    command.
gldas_download -s 2021-01-01 -e 2021-01-31 --product GLDAS_Noah_v21_025 --
    username nina.younkap --password ****** /mnt/path/to/folder
## Note that the last argument in above command is path to a folder where the
    downloaded files will be stored.
```

3.5.6 Processing single data file

Let us now see how to process a single NetCDF file (.nc4) dowloaded from GLDAS using the previous step and perform the required conversions. For example let us consider the GLDAS data representing 06:00 hours on 31 January 2021. The GLDAS file name follows a particular structure. The file name is **GLDAS_NOAH025_3H.A20210131.0600.021.nc4**

- GLDAS_NOAH025_3H represents the name, spatial resolution (025 means 0.25 degree resolution) and temporal resolution (3H means three hour data)
- A20210131 represents the date of acquisition
- 0600 represents the time (in this case 06:00 GMT) the parameters are computed for.
- 021 represents the version of GLDAS data, in this case 2.1.
- nc4 represents the extension and format of the data, in this case netCDF4.

Now let us see how to read the metadata of this file in command line using gdal tools

```
## change directory to where you have downloaded the GLDAS data using the below
cd /mnt/d/mi_is_project_data/2021/31"
## Extract metadata of the 'GLDAS_NOAH025_3H.A20210131.0600.021.nc4' in the
   file "metadata.txt"
gdalinfo GLDAS_NOAH025_3H.A20210131.0600.021.nc4 > metadata.txt
## press enter
## Read metadata using vim editor
vim metadata.txt
```

In the metadata, under Subdatasets: all the parameters provided as subdatasets are listed. These subdataset names are used in the further steps to process individual parameters. For example, NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Tair_f_inst is the name of air temperature grid at 06:00 on 31 January 2021 and we will use this name to extract/process this single grid. The codes of the parameters are listed in the GLDAS user manual (Table 3.1).
So for our project we need the following subdatasets from a single GLDAS file:

NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Qair_f_inst - Specific humidity (Kg/Kg)

NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Psurf_f_inst - Surface Pressure (Pa)

NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Tair_f_inst - Air Temperature (K)

NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Wind_f_inst - Wind speed (m/s)

NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":SWdown_f_inst - Downward shortwave radiation flux (W m-2)

Now let us extract each of the above subdatasets and convert them into tif file. For this we will use a gdal command called gdal_translate which is a global conversion tool for all kind of raster formats. Detailed documentation on gdal_translate is given here.

```
##  Convert specific humidity to tif - below command
gdal_translate NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Qair_f_inst
   GLDAS_NOAH025_3H_A20210131_0600_Qair.tif
```

```
## Convert Surface Pressure to tif - below command
gdal_translate NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Psurf_f_inst
    GLDAS_NOAH025_3H_A20210131_0600_Psurf.tif
## Convert air temperature to tif - below command
gdal_translate NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Tair_f_inst
    GLDAS_NOAH025_3H_A20210131_0600_Tair.tif
## Convert Wind speed to tif - below command
gdal_translate NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":Wind_f_inst
    GLDAS_NOAH025_3H_A20210131_0600_Wind.tif
## Convert Short wave downward radiation to tif - below command
gdal_translate NETCDF:"GLDAS_NOAH025_3H.A20210131.0600.021.nc4":SWdown_f_tavg
    GLDAS_NOAH025_3H_A20210131_0600_SWdown.tif
## Press enter
```

Now that we have the required parameters of 31 January 2021 in tif format, let us open the air temperature map in QGIS and see how it looks!

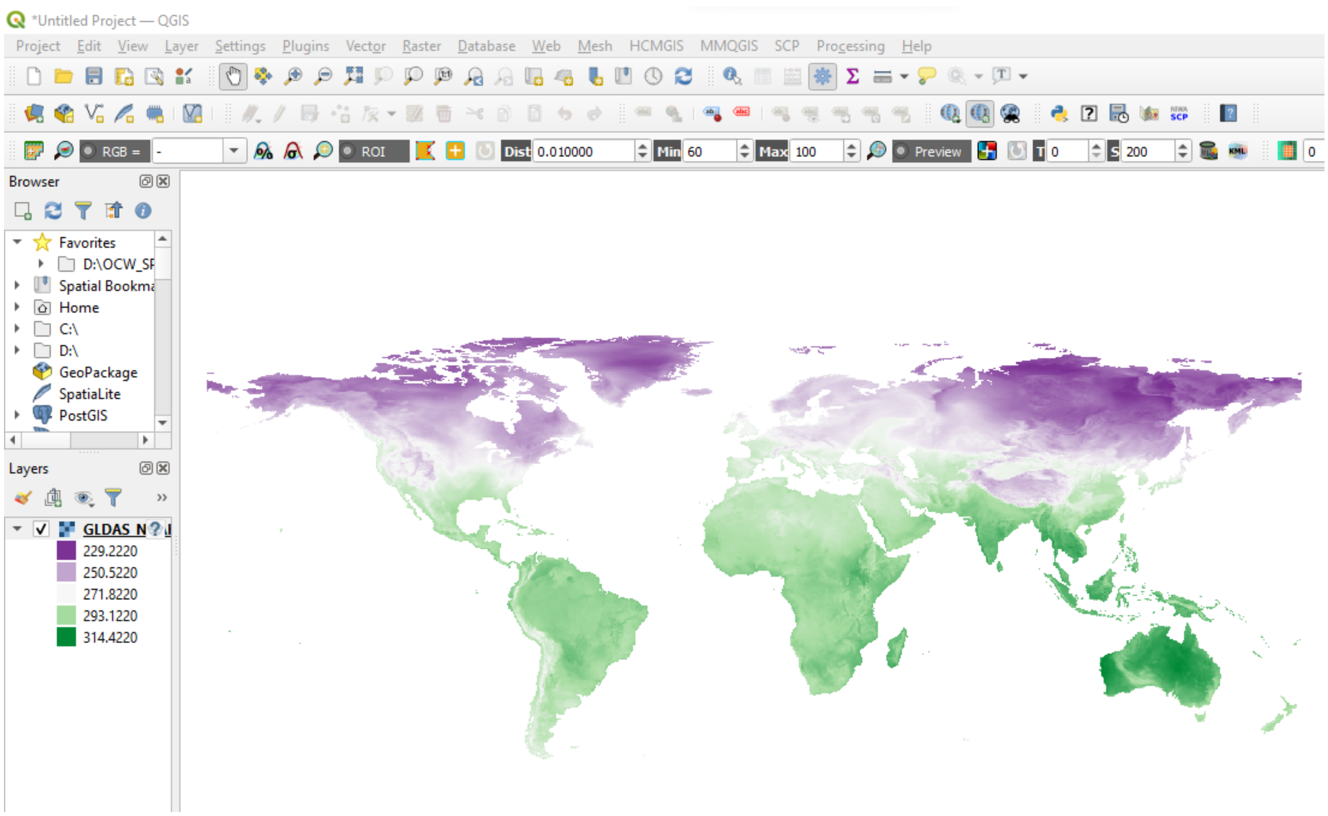


Figure 3.6: GLDAS Tair data displayed in QGIS

Let us also zoom to Cameroon boundaries and have a look at the Tair map. Remember the resolution of data is 0.25 degrees.

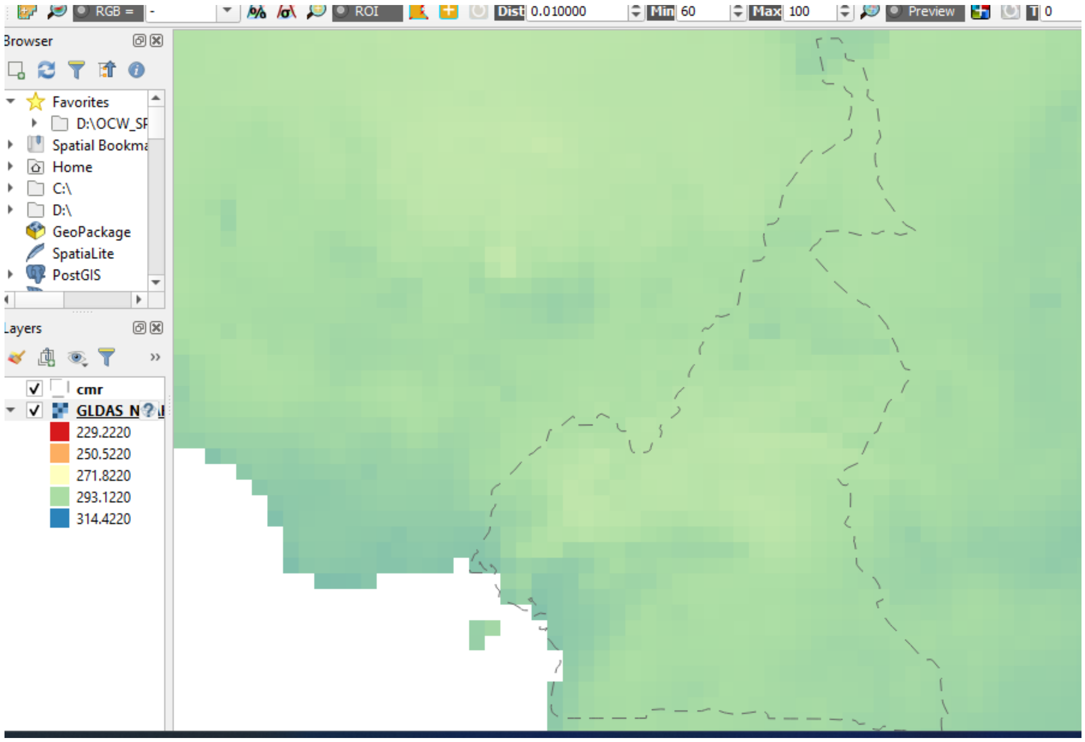


Figure 3.7: GLDAS Tair data displayed in QGIS and zoomed to Cameroon

Now let us see how to do unit conversion of all the five parameters required for aviation. One additional step is to clip the unit converted maps to cameroon boundaries as we are only interested in that region not global.

For further steps let us move to GRASS GIS as it is easier to do spatial and temporal analysis with GRASS library. We will create a new location and mapset for processing GLDAS data for Cameroon.

```
# Create (-c) just the location called "cmr" in epsg:4326 and exit (-e)
grass78 epsg:4326 /mnt/d/grassdata/cmr -c -e
# Create (-c) mapset called "met_data" inside the location "cmr" and open GRASS
    GIS in "cmr/met_data" mapset
grass78 /mnt/d/grassdata/cmr/met_data -c
# "-c" flag in above command is required only one time to create the mapset
   met_data
# Afterwards use below command to just start the existing mapset.
# grass78 /mnt/d/grassdata/cmr/met_data
```

Import the cameroon vector file , aviation meteorological station file , and others file required for our project into grass location.

```
## IMPORT VECTOR DATA: Boundaries of Cameroon an  Aviation meteorological
   station in cameroon
## Navigate (cd) to the folder Cmr_Base_Layers
cd /mnt/d/mi_is_project_data/Cmr_Base_Layers
# Import 'Cameroon' boundary shapefile into a vector in Grass GIS
v.import in=cmr.shp out=cmr
```

```
# Import 'Aviation meteorological staitions' points shapefile into a vector in
    Grass GIS
v.import in=aero_met_station_cmr.shp out=met_station
# Import 'FKKR TMA2' boundary shapefile into a vector in Grass GIS
v.import in=Fkkr_tma2.shp out=fkkr_tma2
# Import 'FKKN TMA' boundary shapefile into a vector in Grass GIS
v.import in=Fkkn_tma.shp out=fkkn_tma
# Import 'FKKL CONTROL ZONE' boundary shapefile into a vector in Grass GIS
v.import in=Fkkl_ctr.shp out=fkkl_ctr
# set the computational region to Cameroon Boundaries and set the computational
     resolution to 0.25 degrees
g.region vector=cmr res=0.25 -a
```

To automate the workflows, the following bash script process all the 5 required parameters (Tair, Rh, Wind speed, Psurf and SWDown), clipped on cameroon boundaries and converted to the required units from .nc4 files in a day.

```
#!/bin/bash
## This script process a single day GLDAS data and do all the required
    conversions needed for our project.
## GENERAL ##
if [ -z "$GISBASE" ] ; then
    echo "You must be in GRASS GIS to run this program." >&2
    exit 1
fi
# Set a environment to enable overwrite by default
export GRASS_OVERWRITE=1

# Navigate to the folder containing single day .nc4 files
#e.g INDAT="/mnt/d/mi_is_project_data/Cmr_Base_Layers/2021/01" for 01 January
    2021
#Here we work on the  31 January 2021 data
INDAT="/mnt/d/mi_is_project_data/2021/031"
cd ${INDAT}
# set the computational region to Urmia Lake basin and set the computational
    resolution to 0.25 degrees
g.region vector=cmr res=0.25 -a
# set the mask to cameroon boundaries
#r.mask vect=cmr

# For loop to process all the .nc files in one go
for i in `ls GLDAS*.nc4`; do
    dt=`echo ${i}|cut -d. -f2-3`
    #  Convert specific humidity to tif - below command
    gdal_translate NETCDF:"${i}":Qair_f_inst GLDAS_NOAH025_3H_${dt}_Qair.tif
    # Convert Surface Pressure to tif - below command
    gdal_translate NETCDF:"${i}":Psurf_f_inst GLDAS_NOAH025_3H_${dt}_Psurf.tif
    # Convert air temperature to tif - below command
```

```
  gdal_translate NETCDF:"${i}":Tair_f_inst GLDAS_NOAH025_3H_${dt}_Tair.tif
  # Convert Wind speed to tif - below command
  gdal_translate NETCDF:"${i}":Wind_f_inst GLDAS_NOAH025_3H_${dt}_Wind.tif
  # Convert Short wave downward radiation to tif - below command
  gdal_translate NETCDF:"${i}":SWdown_f_tavg GLDAS_NOAH025_3H_${dt}_SWdown.tif
  # Import to GRASS
  # Import and clip specific humidity
  r.import in=GLDAS_NOAH025_3H_${dt}_Qair.tif out=GLDAS_NOAH025_3H_${dt}_Qair -o
  # Import and clip Surface Pressure
  r.import in=GLDAS_NOAH025_3H_${dt}_Psurf.tif out=GLDAS_NOAH025_3H_${dt}_Psurf -o
  # Import and clip Tair
  r.import in=GLDAS_NOAH025_3H_${dt}_Tair.tif out=GLDAS_NOAH025_3H_${dt}_Tair -o
  # Import and clip Wind speed
  r.import in=GLDAS_NOAH025_3H_${dt}_Wind.tif out=GLDAS_NOAH025_3H_${dt}_Wind -o
  # Import and clip Short wave radiation
  r.import in=GLDAS_NOAH025_3H_${dt}_SWdown.tif out=GLDAS_NOAH025_3H_${dt}_SWdown -o
  # Unit conversion
  # Air temperature from Kelvin to degree celsius
  r.mapcalc "GLDAS_NOAH025_3H_${dt}_Tair_final = GLDAS_NOAH025_3H_${dt}_Tair - 273.15"
  # Short wave radiation (no conversion required)
  r.mapcalc "GLDAS_NOAH025_3H_${dt}_SWdown_final = GLDAS_NOAH025_3H_${dt}_SWdown"
  # Wind speed (no conversion required)
  r.mapcalc "GLDAS_NOAH025_3H_${dt}_Wind_final = GLDAS_NOAH025_3H_${dt}_Wind"
  ## Specific humidity re saved
  r.mapcalc "GLDAS_NOAH025_3H_${dt}_Qair_final = GLDAS_NOAH025_3H_${dt}_Qair"
  ## Pressure convert from pa to mb
  r.mapcalc "GLDAS_NOAH025_3H_${dt}_Psurf_final = GLDAS_NOAH025_3H_${dt}_Psurf / 100"
  ## Humidity according to the url: https://earthscience.stackexchange.com/questions/2360/how-do-i-convert-specific-humidity-to-relative-humidity
  r.mapcalc "es = 6.112 * exp((17.67 * GLDAS_NOAH025_3H_${dt}_Tair_final) / (GLDAS_NOAH025_3H_${dt}_Tair_final + 243.5))"
  r.mapcalc "e = (GLDAS_NOAH025_3H_${dt}_Qair_final * GLDAS_NOAH025_3H_${dt}_Psurf_final) / (0.378 * GLDAS_NOAH025_3H_${dt}_Qair_final + 0.622)"
  r.mapcalc "GLDAS_NOAH025_3H_${dt}_Rh = (e / es) * 100"
  # Final Relative humidity in %
  r.mapcalc "GLDAS_NOAH025_3H_${dt}_Rh_final = float(if(GLDAS_NOAH025_3H_${dt}_Rh > 100, 100, if(GLDAS_NOAH025_3H_${dt}_Rh < 0, 0, GLDAS_NOAH025_3H_${dt}_Rh)))"
```

```

done
# Remove the mask
#r.mask -r
```

We save the script above in the file *myscript.sh*. The following script show how to run this file on Linux Operating System.

```
# Run the following command to install dos2unix
sudo apt-get install dos2unix
# Below command removes the trailing spaces from your script
dos2unix myscript.sh
# Run the following command to run the above saved script file
sh myscript.sh
# press enter
```

3.5.7 Results

3.5.8 Temperature chart

```
#dt="20210131.0300"
dt="20210131.1500"
# Visualize data
r.mask vect=cmr --o
# Open a monitor
d.mon wx0
# Display a raster map
d.rast GLDAS_NOAH025_3H_A${dt}_Tair_final
# Display a vector map
d.vect map=cmr type=boundary
d.text text="31 JAN 2021 0300Z" color=black bgcolor=white size=3
# Add raster legend
d.legend -t -s -b raster=GLDAS_NOAH025_3H_A${dt}_Tair_final title=TEMPERATURE
    title_fontsize=20 font=sans fontsize=18
# Add North arrow
d.northarrow style=1b text_color=black
#Clear the monitor
#d.erase -f
#r.mask -r
# press enter
```

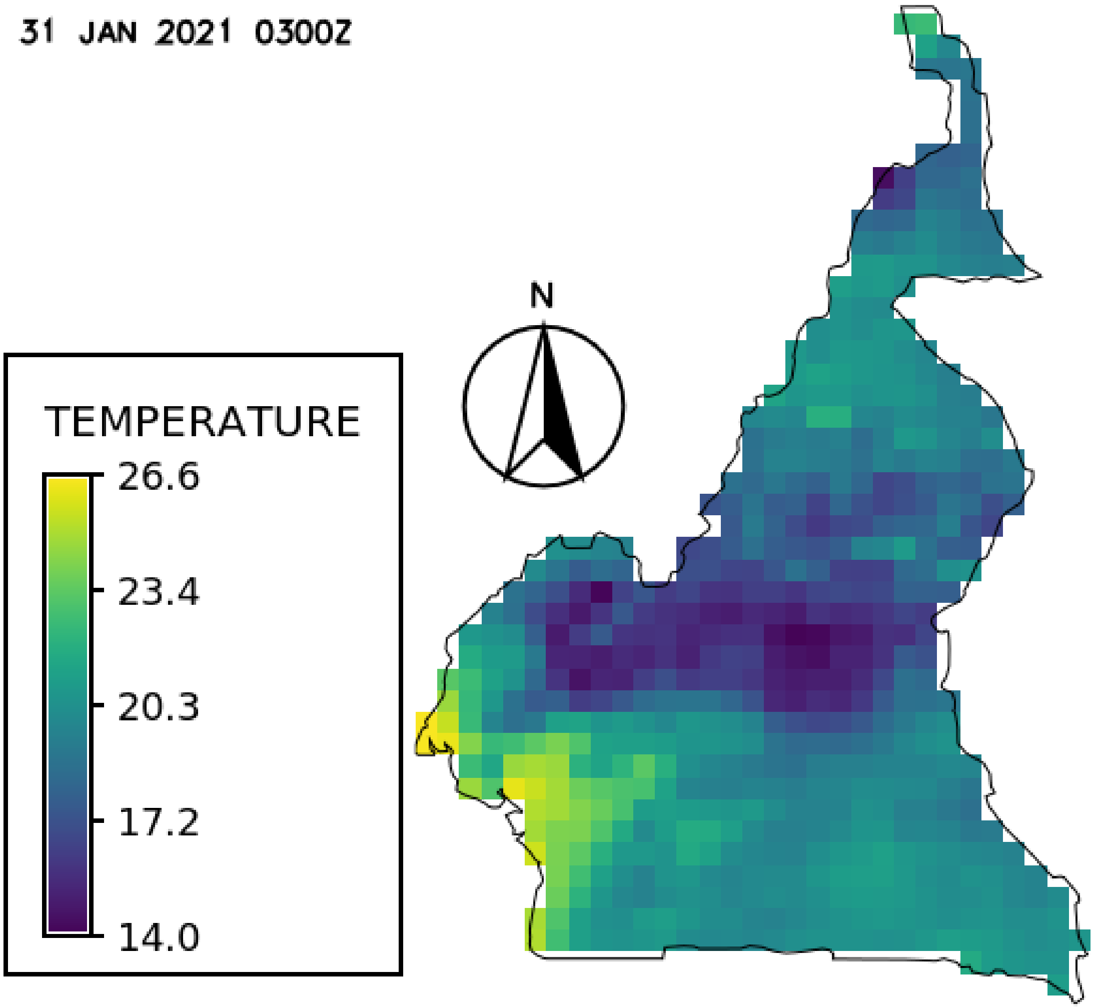


Figure 3.8: Temperature at 03:00 UTC

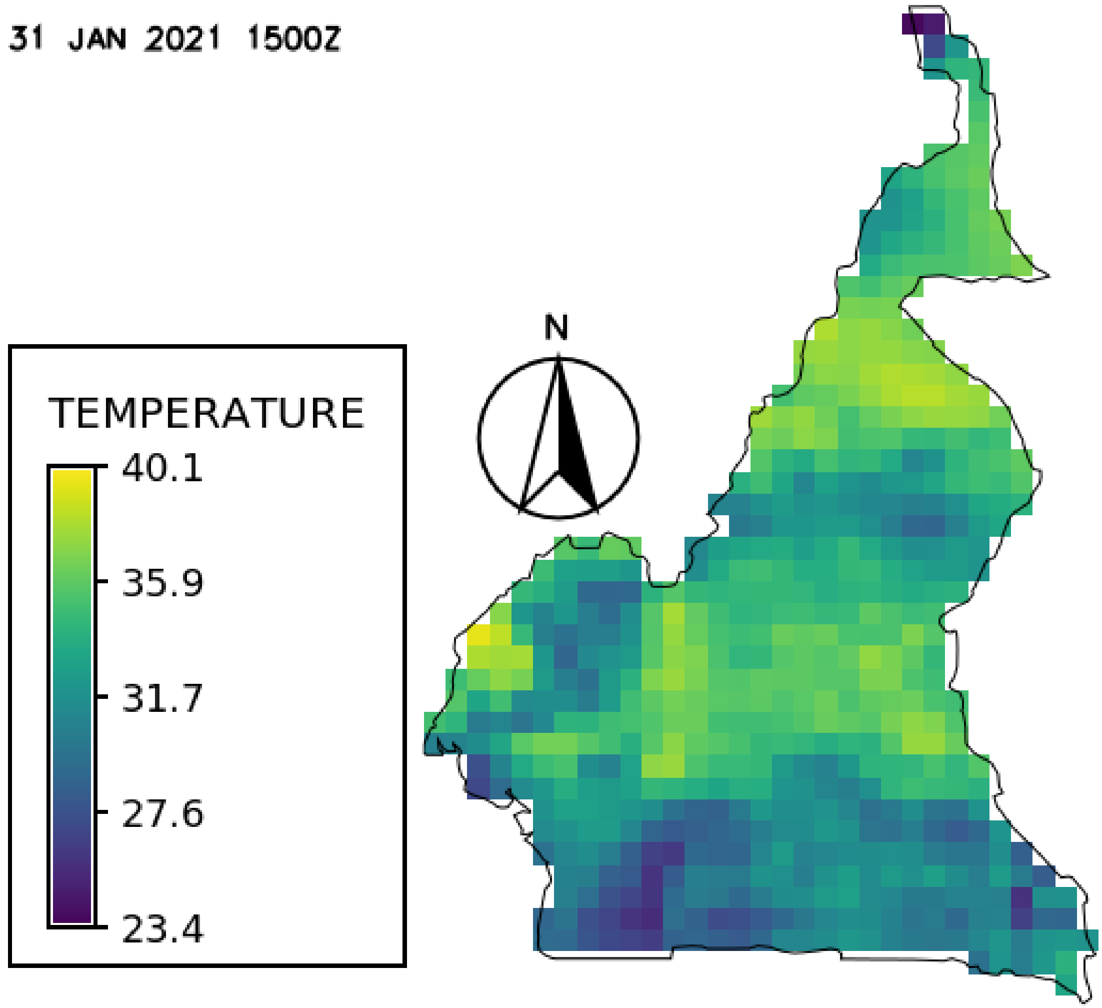


Figure 3.9: Temperature at 15:00 UTC

3.5.9 Relative Humidity Chart

```
dt="20210131.0300"
#dt="20210131.1500"
# Visualize data
r.mask vect=cmr --o
# Open a monitor
d.mon wx0
# Display a raster map
d.rast GLDAS_NOAH025_3H_A${dt}_Rh_final
# Display a vector map
d.vect map=cmr type=boundary
d.text text="31 JAN 2021 0300Z" color=black bgcolor=white size=3
# Add raster legend
d.legend -t -s -b raster=GLDAS_NOAH025_3H_A${dt}_Rh_final title=RELATIVE
    HUMIDITY title_fontsize=20 font=sans fontsize=18
# Add North arrow
```

```
d.northarrow style=1b text_color=black
#Clear the monitor
#d.erase -f
#r.mask -r
```

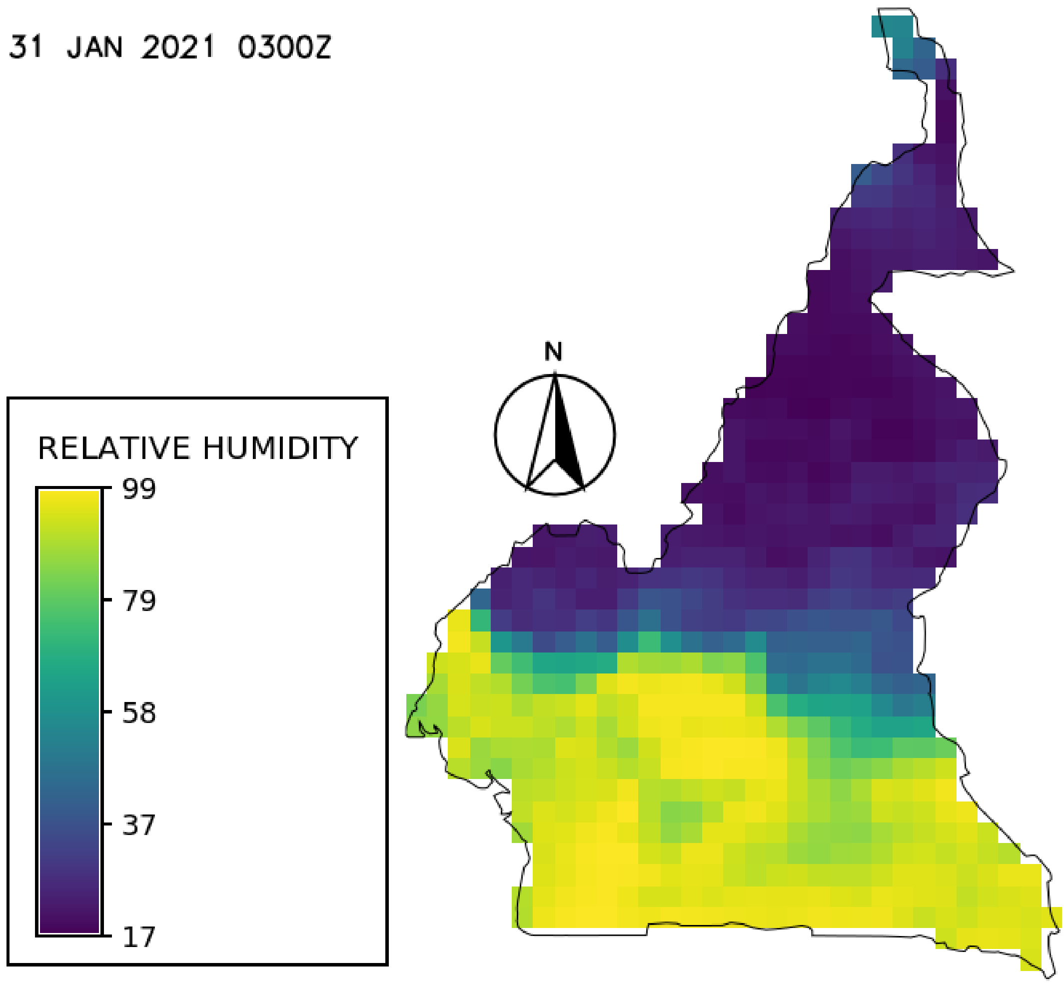


Figure 3.10: Relative Humidity at 03:00 UTC

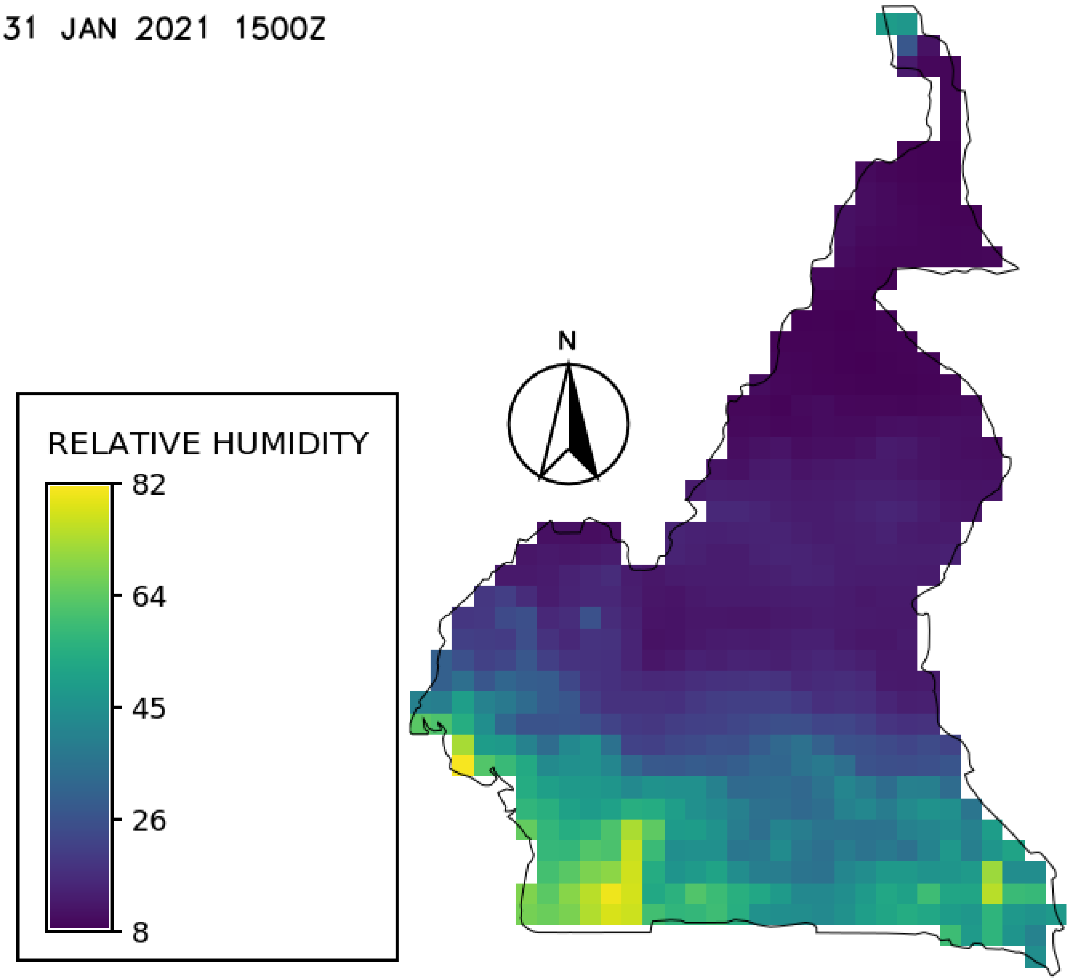


Figure 3.11: Relative Humidity at 15:00 UTC

3.5.10 Surface Pressure Chart

```
#dt="20210131.0300"
dt="20210131.1500"
# Visualize data
r.mask vect=cmr --o
# Open a monitor
d.mon wx0
# Display a raster map
d.rast GLDAS_NOAH025_3H_A${dt}_Psurf_final
# Display a vector map
d.vect map=cmr type=boundary
d.text text="31 JAN 2021 1500Z" color=black bgcolor=white size=3
# Add raster legend
d.legend -t -s -b raster=GLDAS_NOAH025_3H_A${dt}_Psurf_final title="SURFACE
    PRESSURE (Pa)" title_fontsize=20 font=sans fontsize=18
# Add North arrow
d.northarrow style=1b text_color=black
```

```
#d.erase -f
#r.mask -r
```

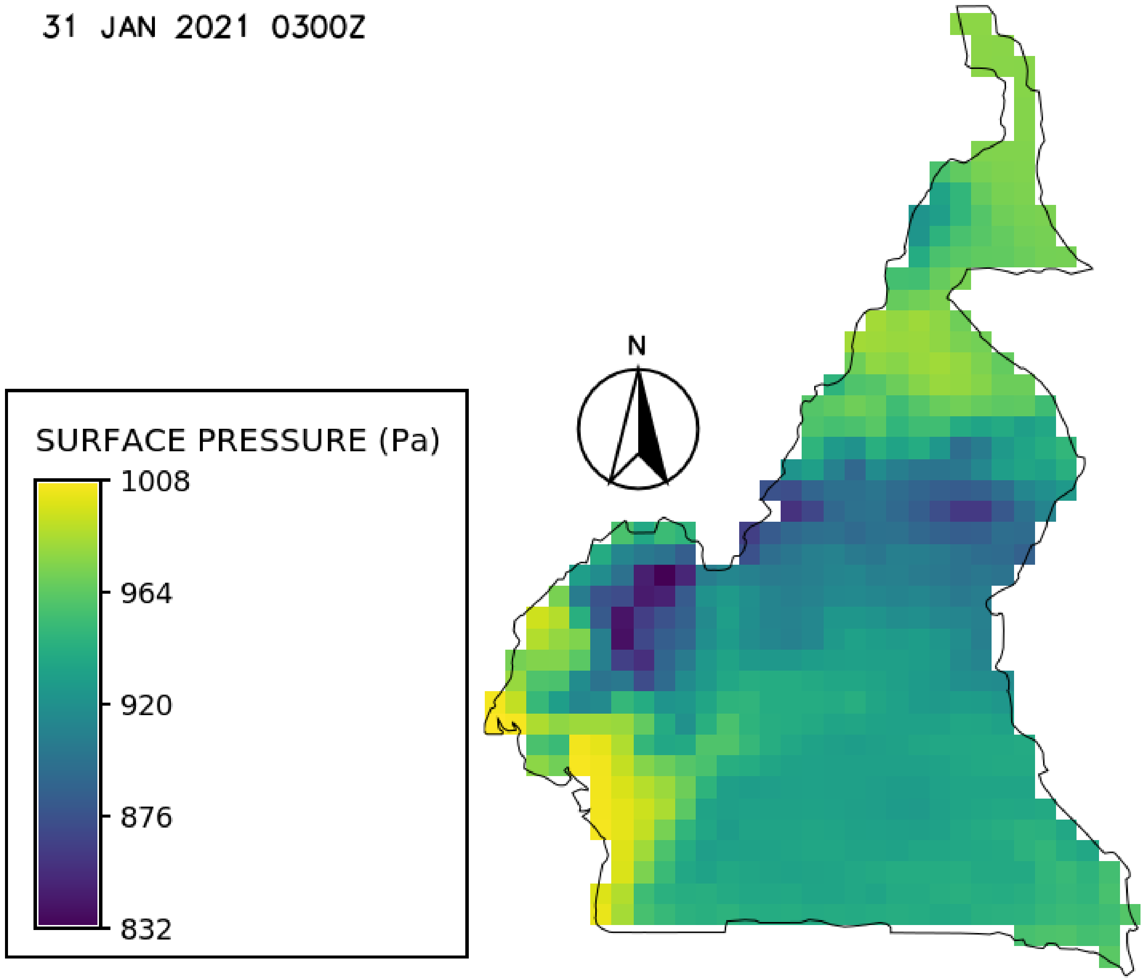


Figure 3.12: Surface Pressure at 03:00 UTC

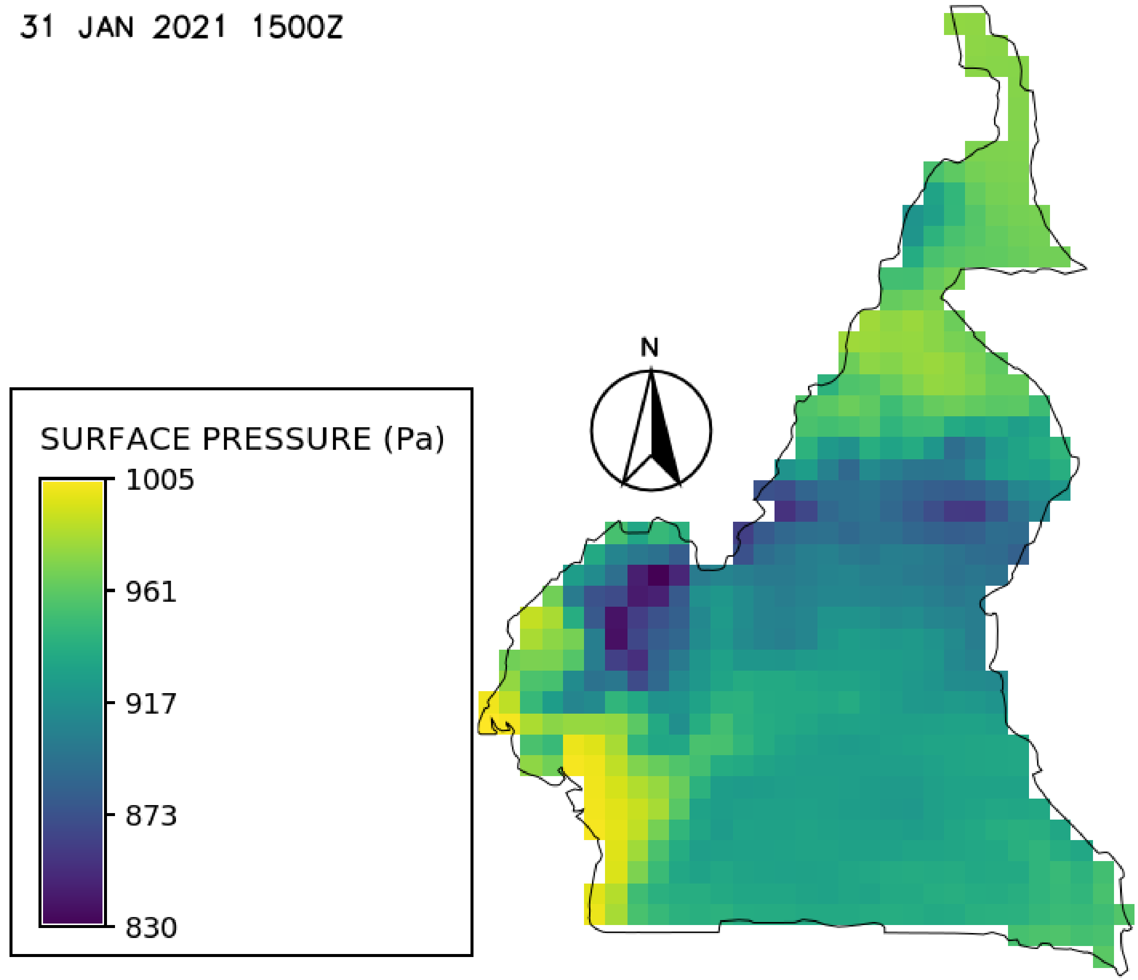


Figure 3.13: Surface Pressure at 15:00 UTC

3.5.11 Wind Speed Chart

```
#dt="20210131.0300"
dt="20210131.1500"
# Visualize data
r.mask vect=cmr --o
# Open a monitor
d.mon wx0
# Display a raster map
d.rast GLDAS_NOAH025_3H_A${dt}_Wind_final
# Display a vector map
d.vect map=cmr type=boundary
d.text text="31 JAN 2021 1500Z" color=black bgcolor=white size=3
# Add raster legend
d.legend -t -s -b raster=GLDAS_NOAH025_3H_A${dt}_Wind_final title="WIND SPEED m/s" title_fontsize=20 font=sans fontsize=18
#Add North arrowd.northarrow style=1b text_color=black
d.northarrow style=1b text_color=black
```

```
#d.erase -f
#r.mask -r
```

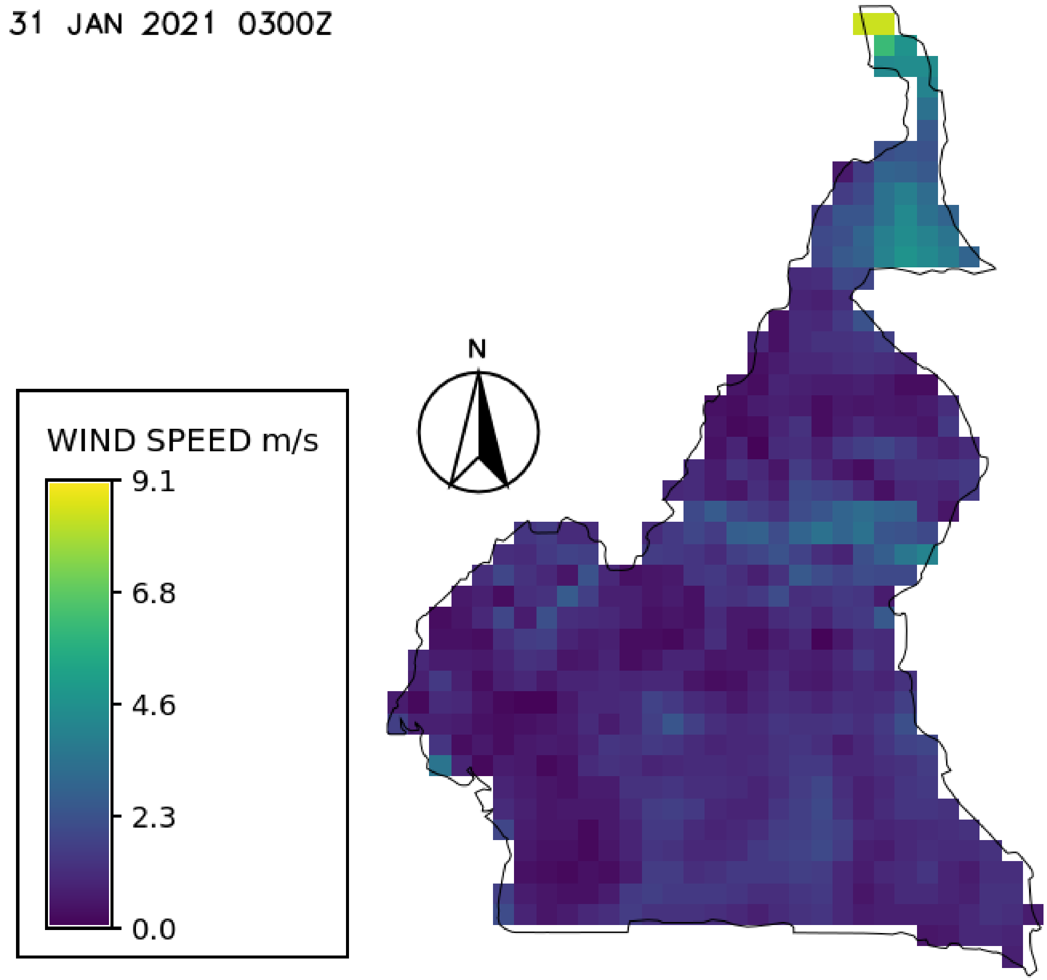


Figure 3.14: Wind Speed at 03:00 UTC

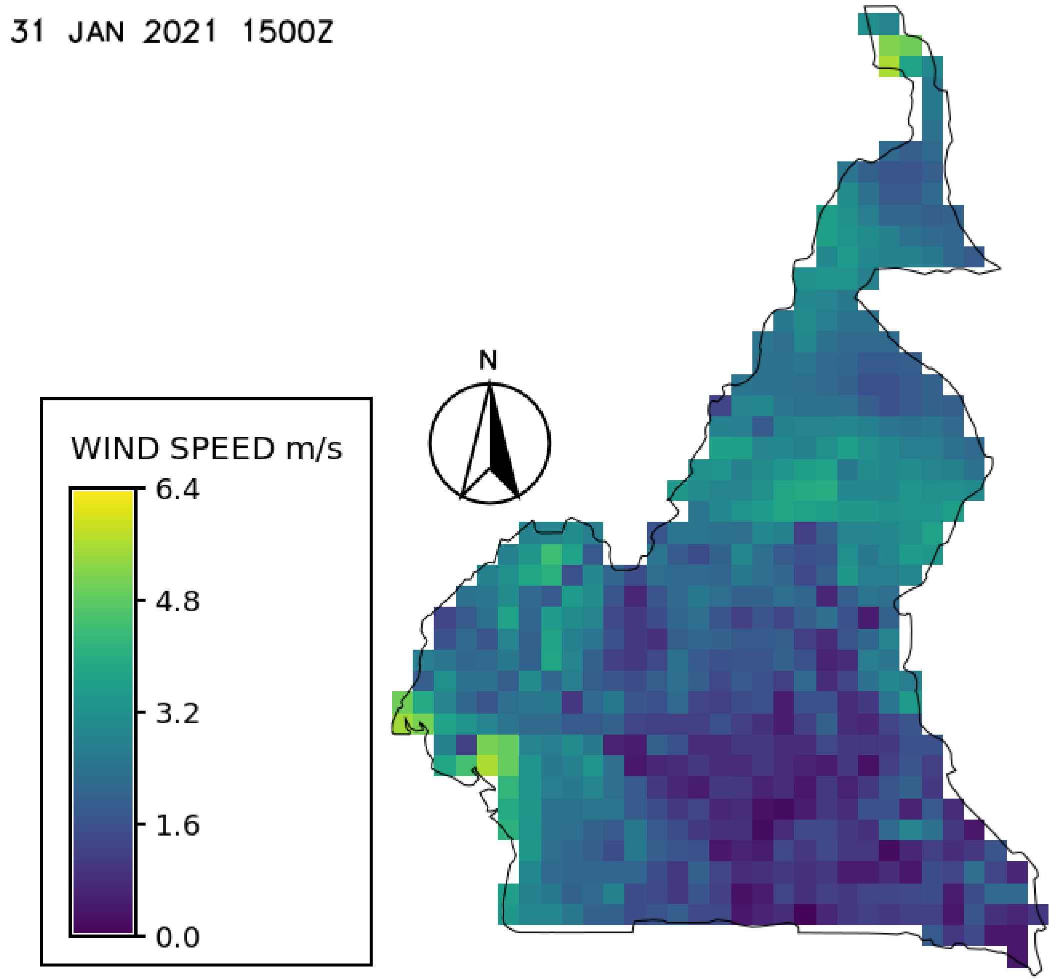


Figure 3.15: Wind Speed at 15:00 UTC