PyGAC FDR ATBD

DOI: 10.15770/EUM_SEC_CLM_0060

Doc.No.	:	EUM/USC/DOC/22/1290471
Issue	:	v2 e-signed
Date	:	20 March 2023
WBS	:	

EUMETSAT
Eumetsat-Allee 1, D-64295 Darmstadt, Germany
Tel: +49 6151 807-7
Fax: +49 6151 807 555
http://www.eumetsat.int

Change Record

Issue / Revision	***Date***	***DCN. No***	***Changed Pages / Paragraphs***
V1	09 February 2022		Initial version
V1a	28 December 2022		Updated version
V1b	13 January 2023		Version for the DRR review
V2	20 March 2023		Final version including the changes following the RIDs

Table of Contents

Table of Figures

Table of Tables

1. INTRODUCTION

1.1 Purpose and scope

The purpose of this document is to provide (1) the theoretical basis for the algorithms needed to convert AVHRR GAC L1b data to calibrated Level 1c data, and (2) the basic description of the PyGAC software architecture used to do this conversion.

1.2 Document structure

After an introduction into the purpose and structure of this document, including a list of acronyms (this section), the background of the PyGAC processing framework is presented (section 2). Next, the satellite sensors are briefly described (section 3), followed by the processing framework description (section 4), including an overview (section 4.1), framework input (4.2) and output (4.2), and the algorithm physical basis (4.3). Algorithm validation is presented in section 5, followed by a description of remaining limitations (section 6), and finally acknowledgements (section 7).

1.3 Reference documents

RD 1	Devasthale, A., M. Raspaud, C. Schlundt, T. Hanschmann, S. Finkensieper, A. Dybbroe, S. Hornquist, N. Hakansson, M. Stengel and K-G. Karlsson, (2017), PyGAC: An open-source, community-driven Python interface to preprocess nearly 40-year AVHRR Global Area Coverage (GAC) data record", GSICS Quarterly Newsletter, Vol. 11, No. 2 (Sept. 2017): 3-5. DOI: 10.7289/V5R78CFR.
RD 2	Finkensieper, Stephan and Martin Raspaud, (2022), pytroll/pygac-fdr: Version 0.2.1 (v0.2.1). Zenodo. https://doi.org/10.5281/zenodo.7113857
RD 3	Finkensieper, Stephan & Martin Raspaud, Nina Hakansson, Adam Dybbroe, A. Devasthale, C. Schlundt, Carlos Horn, Antonio Valentino, Daniel Lee, D. Stein, and Oliver Sus, (2022), pytroll/pygac: Version 1.5.0 (2022/01/10) (v1.5.0). Zenodo. https://doi.org/10.5281/zenodo.5832775.
RD 4	Goodrum, G. et al.: NOAA KLM user's guide [electronic resource] / edited by Geoffrey Goodrum, Katherine B. Kidwell, Wayne Winston. Asheville, N.C. : U.S. Dept. of Commerce, National Oceanic and Atmospheric Administration, National Environmental Satellite, Data, and Information Service, National Climatic Data Center, Climate Services Division, Satellite Services Branch, 1999. https://lccn.loc.gov/2009578433.
RD 5	Heidinger, A. K., (2018), Climate Algorithm Theoretical Basis Document (C-ATBD): Fundamental Climate Data Record (CDR) of Reflectance from AVHRR Bands 1, 2 and 3a, CDR Program Document Number CDRP-ATBD-0184, Revision 2.
RD 6	Hollmann, R., Merchant, C., Saunders, R., Downy, C., Buchwitz, M., Cazenave, A., Chuvieco, E., Defourny, P., Leeuw, G. de, Forsberg, R., Holzer-Popp, T., Paul, F., Sandven, S., Sathyendranath, S., Roozendael, M. van, and Wagner W. (2013), The ESA Climate Change Initiative: satellite data records for essential climate variables, Bulletin of the American Meteorological Society, doi: 10.1175/BAMS-D-11-00254.1
RD 7	**Kelso** T.S.**, Orbital Coordinate Systems, Part II. Satellite Times, 2, no. 2 (November/December 1995): 78-79. http://celestrak.org/columns/v02n02/**
RD 8	Kidwell, K. B., (1995), NOAA Polar Orbiter Data Users Guide / edited by Kidwell, K.B. and National Climatic Data Center (U.S.). National Oceanic and Atmospheric Administration, National Environmental Satellite, Data, and Information Service, National Climatic Data Center, Satellite Data Services Division.

RD 9	Raspaud, Martin and David Hoese, Panu Lahtinen, Stephan Finkensieper, Gerrit Holl, Adam Dybbroe, Simon Proud, Andrea Meraner, Joleen Feltz, Xin Zhang, Sauli Joro, William Roberts, Lars Ørum Rasmussen, Jorge Hummberto Bravo Méndez, Yufei Zhu, Rohan Daruwala, Tommy Jasmin, Ron Goodson, (2021), pytroll/satpy: Version 0.30.0 (v0.30.0). Zenodo. https://doi.org/10.5281/zenodo.5514044
RD 10	Raspaud, Martin & Panu Lahtinen, Adam Dybbroe, David Hoese, Antonio Valentino, Lars Ørum Rasmussen, Stephan Finkensieper, Gerrit Holl, Jordan Lui, Sajid Pareeth, Xin Zhang, Ruben Di Battista, AndreaMalgarini, Pablo Canseco, (2021), pytroll/pyorbital: Version 1.7.1 (2021/12/22) (v1.7.1). Zenodo. https://doi.org/10.5281/zenodo.6078954
RD 11	Schlundt, C., Finkensieper, S., Raspaud, M., Karlsson, K.-G., Stengel, M., Hollmann, R., (2017), Technical Report on AVHRR GAC FCDR generation. ESA Cloud_cci.
RD 12	Schulz, J., Albert, P., Behr, H.D., et al., (2009), Operational climate monitoring from space: the EUMETSAT Satellite Application Facility on Climate Monitoring (CM-SAF), Atmos. Chem. Phys., 9, 1687-1709,doi:10.5194/acp-9-1687-2009.
RD 13	**Vallado, D. A., P. Crawford, R. Hujsak, and T.S. Kelso** (2006), **Revisiting Spacetrack Report #3, presented at the AIAA/AAS Astrodynamics Specialist Conference, Keystone, CO, August 21–24.** 2006, **http://celestrak.org/publications/AIAA/2006-6753/**
RD 14	Walton, C. C., Sullivan, J. T., Rao, C. R. N., and Weinreb, M. P. (1998), Corrections for detector nonlinearities and calibration inconsistencies of the infrared channels of the advanced very high resolution radiometer, J. Geophys. Res., 103(C2), 3323– 3337, https://doi:10.1029/97JC02018.

1.4 Acronyms

Acronym	*Meaning*
AVHRR	Advanced Very High Resolution Radiometer
CDR	Climate Data Record
CLASS	Comprehensive Large Array-data Stewardship System
CM-SAF	SAF on Climate Monitiring
ECV	Essential Climate Variables
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
ESA	European Space Agency
FCDR	Fundamental Climate Data Record
FDR	Fundamental Data Record
GAC	Global Area Coverage
GCOS	Global Climate Observing System
ICT	Internal Calibration Target
K	Kelvin
KLM	NOAA K,L,M,N,N' satellites (NOAA-15 to -19 and MetOpA-C)
L1b	Level 1b
L1c	Level 1c
LAC	Local Area Coverage
Metop	Meteorological operational satellite
MODIS	Moderate Resolution Imaging Spectroradiometer
NH	North Hemisphere
NIR	Near InfraRed

NOAA	National Oceanic and Atmospheric Administration
PATMOS-x	avhrr **PA**thfinder at**MOS**pheres – e**X**tended
POD	Polar Orbit Data NOAA TIROS-N and A,C,D,E,F,G,H,I,J satellites (NOAA-6 to -14)
PRT	Platinium Resistance Thermometer
RD	Referenced Document
RPY	Roll-Pitch-Yaw
SAF	Satellite Application Facility
SCOPE-CM	Sustained, Coordinated Processing of Environmental Satellite Data for Climate Monitoring
SNO	Simultaneous Nadir Overpass
TIR	Thermal Infrared
TLE	Two-Line-Element
TIROS-N	Television and Infra-Red Observation Satellite - N
UNFCCC	United Nations Framework Convention on Climate Change
VIS	Visible

2. BACKGROUND

More than four decades (1978 to present) of Global Area Coverage (GAC) data are available from the Advanced Very High Resolution Radiometers (AVHRR) onboard a series of NOAA satellites. These data offer the opportunity to derive Fundamental Data Records (FDRs) of the Essential Climate Variables (ECVs) listed by the Global Climate Observing System (GCOS) to provide support for the United Nations Framework Convention on Climate Change (UNFCCC).

The first important step in deriving an FDR is to read, decode, quality control and (inter)calibrate packed 10-bit Level 1b GAC data. Although this preprocessing step is essential in all applications of GAC data, there is no clear traceability and uniform agreement on preprocessing Level 1b data in an internationally agreed convention. Addressing this issue has been the starting point for the development of PyGAC, a python based open source, community driven interface to preprocess AVHRR GAC data (Devasthale et al., 2017). Moreover, this FDR is the only AVHRR radiance record available in original orbit sampling.

PyGAC was initially developed under the framework of ESA's Cloud Climate Change Initiative Phase II (ESA Cloud_cci, Hollmann et al. 2013) with user feedback from EUMETSAT's Satellite Application Facility on Climate Monitoring (CM SAF, Schulz et al. 2009), and is now maintained by the Pytroll community (http://pytroll.github.io/). PyGAC takes advantage of the modular and object oriented philosophy of Python and its vast cache of public utilities. The AVHRR GAC FDR framework presented here is prepared using this PyGAC software architecture.

3. SATELLITE SENSORS

The NOAA and EUMETSAT satellites circle the Earth 14 times per day at an altitude of about 833 km. The AVHRR instrument takes 2048 Earth views per scan and per channel, having a swath width of 2893 km and a pixel resolution of about 1.1 km at nadir.

The AVHRR instrument includes visible (VIS), near infrared (NIR), and thermal infrared (TIR) channels. The AVHRR passive imager was first a 4-channel radiometer and subsequently improved to 5 and 6 channels at wavelengths between 0.55 and 12.5 µm (Table 1).

Table 1: *Details about the three versions of the AVHRR instrument. Information was derived from the OSCAR WMO website.*

	AVHRR	**AVHRR/2**	**AVHRR/3**
Period	1978 – 2001	1981 – 2007	1998 – 2025
Satellites	TIROS-N NOAA: 6, 8, 10	NOAA: 7, 9, 11, 12, 13, 14	NOAA: 15, 16, 17, 18, 19 Metop: A, B, C
Num of channels	4	5	6
Channels central wavelength (µm) and spectral interval (µm)	**1:** 0.615 (0.55-0.68*) **2:** 0.912 (0.725-1.1) **3:** 3.74 (3.55-3.93) **4:** 11.0 (10.5-11.5) *0.55-0.9 for TIROS-N	**1:** 0.615 (0.55-0.68) **2:** 0.912 (0.725-1.1) **3:** 3.74 (3.55-3.93) **4:** 10.80 (10.3-11.3) **5:** 12.00 (11.5-12.5)	**1:** 0.630 (0.58-0.68) **2:** 0.862 (0.725-1.00) **3A:** 1.61 (1.58-1.64) **3B:** 3.74 (3.55-3.93) **4:** 10.80 (10.3-11.3) **5:** 12.00 (11.5-12.5)
Swath (in pixels)	across track: 2048 (LAC), 409 (GAC) along track: 6		
Resolution at nadir (km) [1]	1.1		

The AVHRR data are available at two spatial resolutions. Firstly, the Local Area Coverage (LAC) format, which provides observations at original pixel resolution with 1.1x1.1 km^2 at nadir. Secondly, the GAC format, in which four out of every five samples along the scan line are used to compute one average value. The data from only every third scan line are processed, resulting in a spatial resolution of GAC data at nadir of 1.1x4.4 km^2 with a 2.2 km gap between two scan lines (Figure 1). This spatial resolution changes as a function of the scan angle, and pixels are distorted primarily along but also across the scan (Figure 2). For the instruments on the NOAA satellites, this format was created to reduce the volume of onboard data storage until the data were downlinked to the major ground station. Metop data are available in LAC format and have been reformatted to the GAC format before input to PyGAC. All AVHRR input data used to create this FDR were GAC data downloaded from the NOAA CLASS.

[1] The swath and resolution indicated in the table refer of the full AVHRR LAC resolution. The FDR is derived using AVHRR GAC resolution (see figure 1).

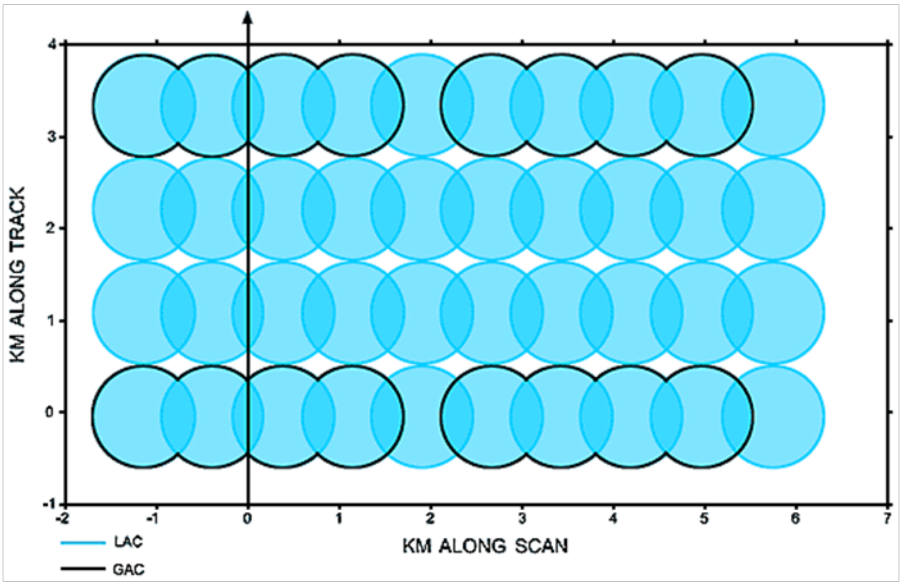

Figure 1: *Idealised view of LAC and GAC spatial sampling.*

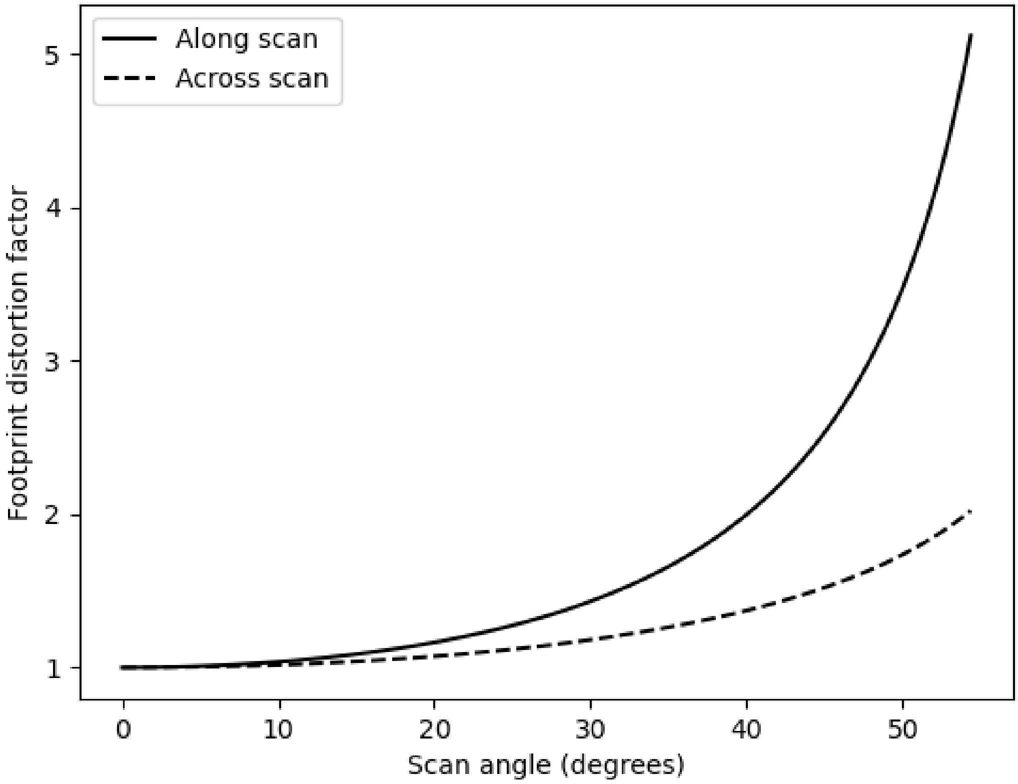

Figure 2: *Footprint distortion of a pixel along the scanline. This was computed using lats/lons of a real NOAA-6 orbit and an approximate altitude of 833km. Reference: Capderou, Michel (2014): Handbook of Satellite Orbits, Springer. Section 12.2 - 12.3.*

4. PROCESSING FRAMEWORK DESCRIPTION

4.1. Overview

The overall reprocessing framework is labelled PyGAC. In this section however, we are referring to its individual python modules as *Pygac-fdr* and *Pygac*.
The FDR dataset is based on AVHRR Level 1B data in NOAA-GAC format. The dataset was generated using the open-source software package *Pygac-fdr* (Finkensieper et al., 2022), which is a light wrapper to combine *Satpy* (Raspaud et al. 2021) and *Pygac* (Finkensieper et al. 2022), see Figure 3. *Pygac* is used for decoding the GAC format as well as calibrating raw counts to reflectances or brightness temperatures and navigation. *Pygac* makes use of *Pyorbital* (Raspaud et al., 2021) for geolocation and satellite/solar angle computation. *Pygac-fdr* calls *Pygac* and saves the results to CF-compliant netCDF files. Both steps go through *Satpy*, which provides easy access to *Pygac*'s GAC reader and a general-purpose netCDF-CF writer.

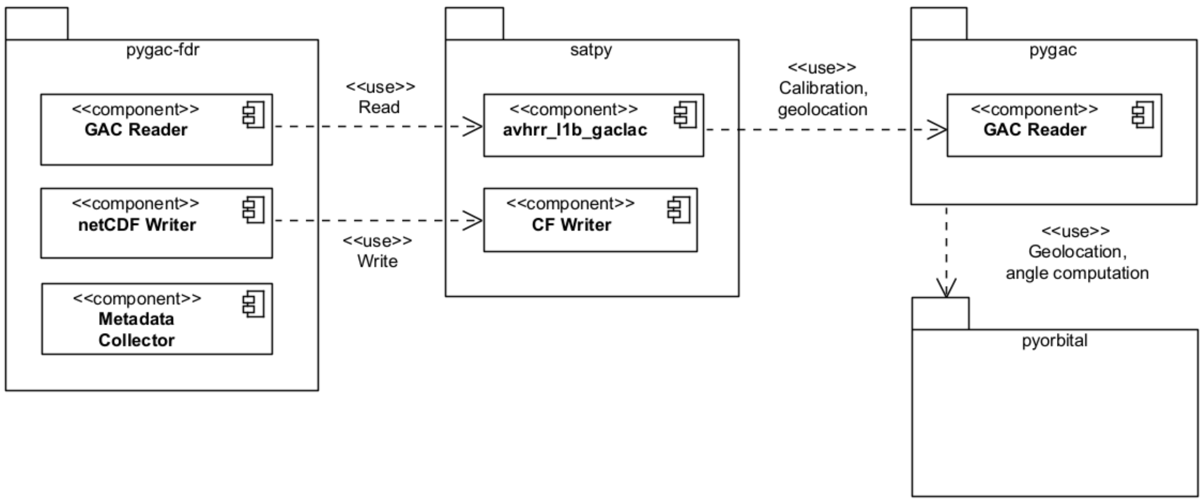

Figure 3: *Open Source software packages used to create the dataset.*

4.2. Framework input

As input, PyGAC requires a single AVHRR GAC level 1b file (see section 3), or a set of such files which will be processed sequentially. All data 1978-present were downloaded from NOAA CLASS (https://class.noaa.gov). The NOAA-GAC format is described in the POD (Kidwell, 1995; TIROS-N through NOAA-14) and KLM (Goodrum, 1999; NOAA-15 onwards) user guides.
Moreover, satellite-specific Two-Line-Elements (TLE) need to be provided that cover the time range of the GAC input data. They were obtained from Space Track (https://www.space-track.org/documentation#/tle).

4.3. Framework output

Calibrated and navigated AVHRR level 1c data are saved in CF-compliant netCDF4 format - one file per orbit. As a byproduct of a processing, an sqlite database file is being created. It contains metadata information for each individual orbit, such as filename, platform, start and end time, number of scanlines, start and end orbit numbers, time and longitude of equator crossing in ascending node, start and end scanline numbers of the overlap free fraction, and a global quality flag.

4.4. L1c dataset generation

The dataset is generated by applying PyGAC to the entire AVHRR GAC level 1b archive and saving the resulting level 1c data to netCDF. These tasks are independent and can be executed in parallel. Based on the level 1c data, additional metadata are computed and added to the netCDF files. This computation requires the analysis of a file's neighborhood (in time), which is why all output files are required for this step.

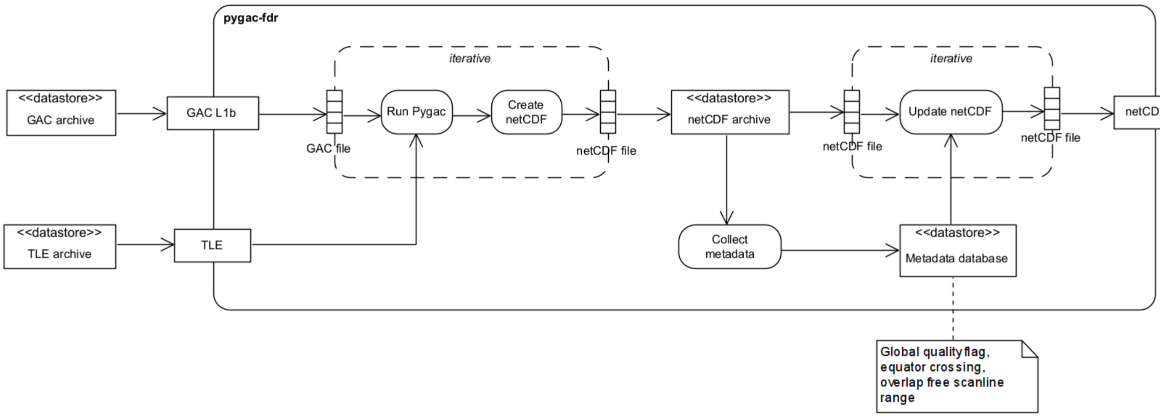

Figure 4: *Reprocessing workflow*

4.5. Algorithm physical basis

4.5.1. Overview

PyGAC starts with decoding the 10-bit GAC format, along with the header and a host of other metadata related to calibration, navigation and quality. PyGAC is able to read both the POD and KLM GAC formats. The POD and KLM data user guides (Kidwell, 1995 and Goodrum, 1999) are taken as the reference source for the design of the corresponding readers and calibrators. The next processing steps involve navigation, thermal and solar calibration as well as the derivation of solar and satellite angles. Finally, a quality control is applied. See Figures 5 and 6 for detailed data flow and control flow diagrams.

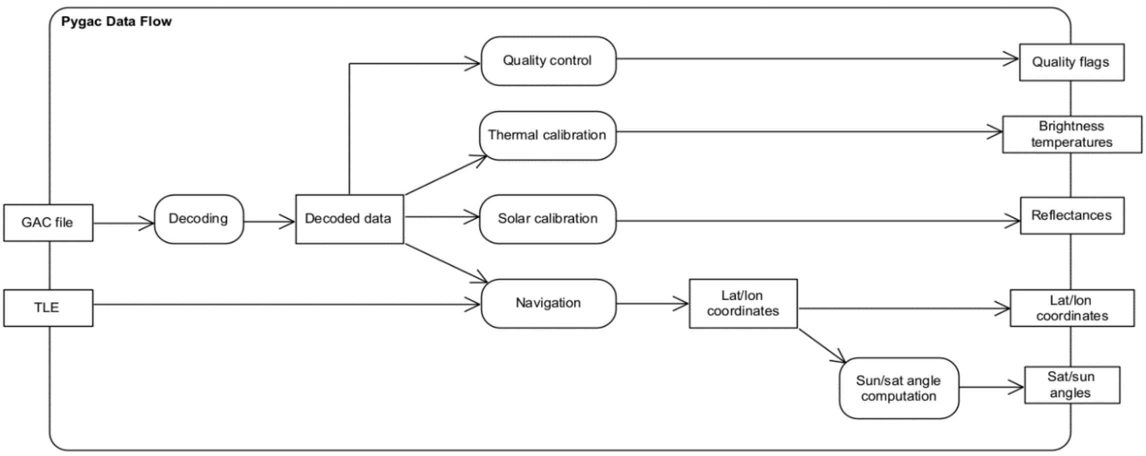

Figure 5*: PyGAC dataflow*

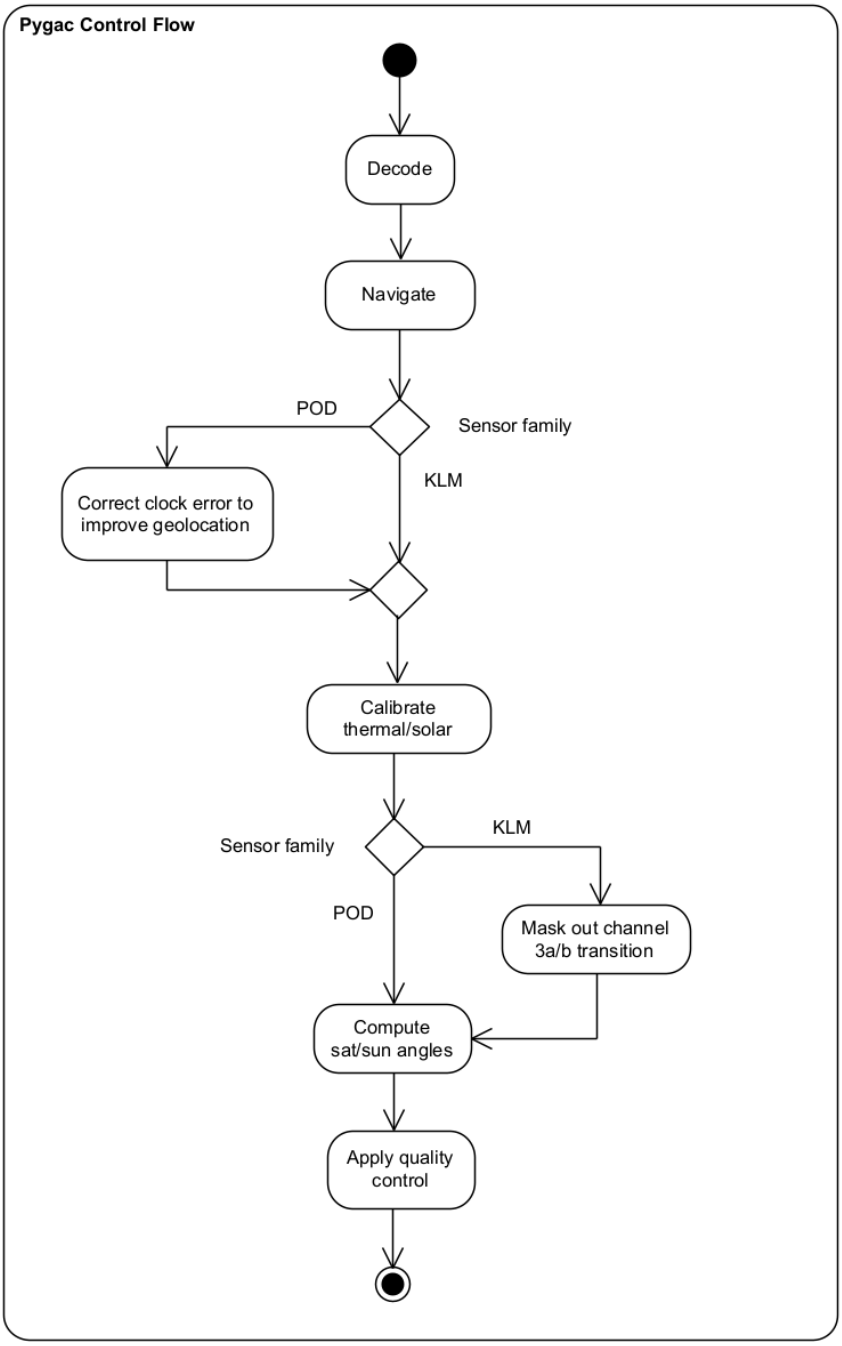

Figure 6: *PyGAC control flow*

4.5.2. Calibration

4.5.2.1. Visible calibration

For all satellites, calibration coefficients and their updates over the years were provided by NOAA (Mike Foster and Andrew Heidinger, personal communication, 2021 + 2022) in the framework of bilateral collaboration between EUMETSAT and NOAA. The version used for this processing is PATMOS-x SOLCAL, v2017r1, including provisional coefficients for MetOp-C (see ATBD here: https://docs.google.com/document/d/1N-Vw5zMNJ6HlR8eNC2s99ue0NxFPxv2I7ezuxdEilv8/edit# and Heidinger, 2018).

The coefficients are included in PyGAC (https://github.com/pytroll/pygac/blob/v1.5.0/pygac/data/calibration.json).
These solar channel calibration coefficients provided by NOAA are the result of a multi-tier process. The major change in their recent methodology is the use of Spectral Band Adjustment Factors (SBAFs) to minimise the spectral differences among the various AVHRR sensors. The SBAFs provide accurate top of the atmosphere spectral adjustments that intrinsically include the impacts of gasses, aerosols and the surface. The primary tier of the calibration process relies heavily on the SNOs between AVHRR and MODIS sensors. The SNO criteria dictate that: a) a time difference be less than 5 minutes, b) a solar zenith angle difference be less than 10 degrees, c) a maximum sensor zenith angle shall be 10 degrees, and finally d) a sensor zenith difference shall be less than 10 degrees. These criteria are applied to SNO events in the North Pole region from days 180 to 220. The AVHRR GAC counts are regressed against SBAF adjusted MODIS reflectances in different solar channels to compute gains.
In the second tier of the calibration procedure, PATMOS-x SOLCAL uses four CEOS recommended physical sites to match the AVHRR observations. Two of these sites are deserts (Libya-1 and Badain-Jaran) and two are ice sheets (Greenland and Dome-C). The desert sites provide observations for the afternoon and mid-morning sensors, while the ice sheets provide data for all sensors. The locations of these sites are given below in Table 2.

Table 2: *Locations of the CEOS Calibrations as used here.*

Location	Latitude South	Latitude North	Longitide West	Longitude East
Badain-Jaran	39.9	40.3	-101.6	-102.0
DOME-C	-74.95	-74.05	122.05	123.95
Greenland	69.05	69.95	-40.95	-39.05
Libya-1	23.7	25.2	12.61	14.09

The MODIS reflectance at these sites are studied as a function of solar zenith angles. The matches with AVHRR observations are then done after adjusting MODIS reflectance using SBAFs.
The next important tier in the PATMOS-x SOLCAL process is the generation of AVHRR/AVHRR SNOs. The MODIS record extends back in time only until 2000. Transferring calibration using MODIS/AVHRR SNOs to AVHRR/AVHRR SNOs therefore becomes necessary, also to minimise satellite-to-satellite inconsistencies.
The final step in the SOLCAL process is to merge individual calibration slopes obtained in the various tiers mentioned above in order to derive the slopes as function of the time since the launch, as shown below.

$$S(t) = \frac{S_0(100 + S_1 \times t + S_2 \times t^2)}{100}$$

Where t is the time since launch expressed in years, S(t) is the calibration slope and S_0, S_1 and S_2 are the coefficients of the quadratic fit.
The reflectance derived using these PATMOS-x SOLCAL coefficients are of highest quality and follow the Global Climate Observing System (GCOS) standards for deriving fundamental climate data records. The reflectance are further normalised by a factor (as a function of day of

year) to account for changing Earth-Sun distance, but are not normalised by the cosine of the solar zenith angle.

4.5.2.2. IR calibration

The thermal channel calibration is done from scratch, starting from obtaining Platinum Resistance Thermometer (PRT), space and Internal Calibration Target (ICT, blackbody) counts. ICT temperatures are obtained from PRT counts based on coefficients provided in the POD and KLM Data User Guides (Kidwell, 1995). For each thermal channel, a smoothing window of 51 successive PRT, ICT and space counts is used to obtain robust gain values and to dampen undue high frequency fluctuations in the count data (Trishchenko, 2002). Section 7.1.2.5 of the KLM User Guide presents the summary of equations implemented in PyGAC to calibrate thermal channels, including the non-linearity correction (Walton et al. 1998).
The PRT readings are supposed to be present in a specific order, for example, the reset value followed by four readings from the PRTs. However, this may not always be the case for orbits that contain data gaps or due to any other unexplained reason. If not taken into account properly, this irregularity could result in the underestimation of brightness temperatures, when calibration information is smoothed over many scanlines. In PyGAC, this inconsistency is handled properly while calibrating thermal channels. In some cases it was found that, apart from the reset values, even the readings from any one of the four PRTs could also have very low suspicious values. This could also seriously affect the computation of brightness temperatures. PyGAC detects such anomalies and corrects them using interpolation of nearby valid PRT readings.

4.5.3. Geolocation

Each GAC scan line contains 409 pixels. However, latitude-longitude values are provided for every eighth pixel starting from pixel 5 and ending at pixel 405. Using Numpy, Scipy and Pyresample packages, inter- and extrapolation is carried out to obtain a geolocation for each pixel along the scanline.
If the GAC data belong to the POD family, then clock drift errors are used to adjust existing latitude-longitude information. Here, PyGAC makes use of the PyOrbital package (https://pyorbital.readthedocs.io/en/latest/). PyGAC interpolates the clock offset and adjusts the nominal scan times to the actual scan times. Since the geolocation was computed using the nominal scan times, PyGAC interpolates the latitudes and longitudes to the actual scan times using spherical linear interpolation, aka `slerp`. However, in the case of a clock drift error greater than the scan rate of the dataset, the latitude and longitude for each pixel of the scan lines that cannot have an interpolated geolocation (typically at the start or end of the dataset) are recomputed. This is done using pyorbital, which in turn does two things:1) compute the position of the satellite at each scan line by TLE data, 2) compute the longitude and latitude of each pixel using instrument geometry.

4.5.4. Computation of angles

The azimuth angles are calculated using `pyorbital` (see also Vallado et al. 2006 and Kelso 1995). The azimuth is measured clockwise from North instead of counter-clockwise from South. Counter-clockwise from South would be the standard for a right-handed orthogonal coordinate system. All angles are converted to degrees and azimuth angles are converted to the open interval]-180, 180].

4.5.5. Correction of satellite location

PyGAC uses roll-pitch-yaw (RPY) corrections along with other orbital parameters to compute accurate satellite location (e.g. instead of assuming constant altitude). RPY values varied considerably throughout the lifetime of KLM satellites (Figure 7). RPY data are not available for POD satellites, for which these corrections are set to zero.

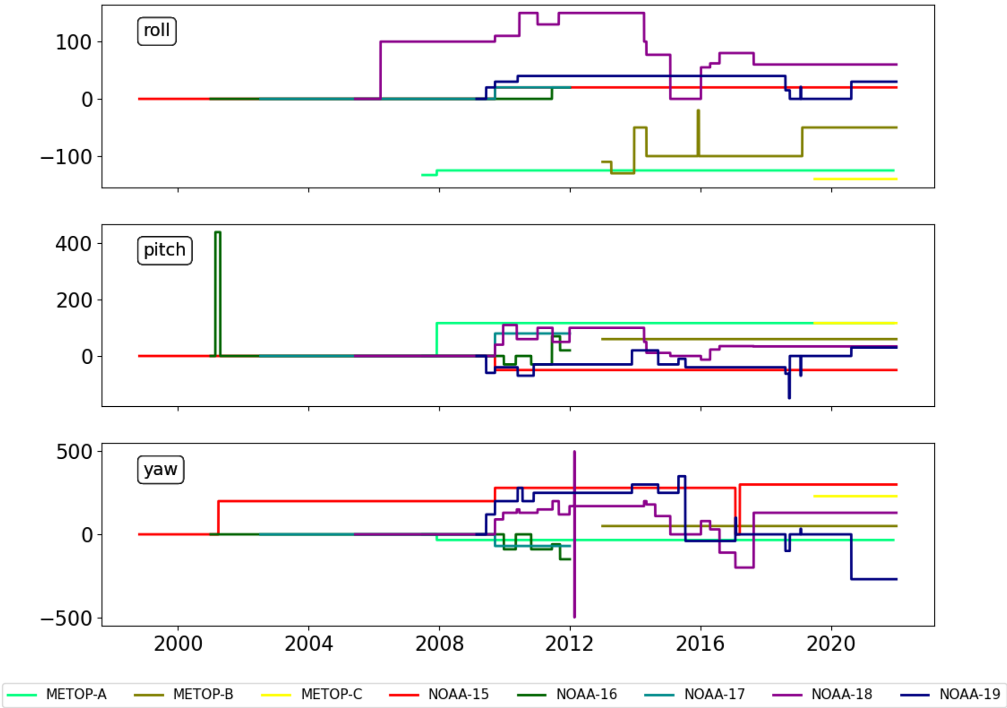

Figure 7: *Roll, pitch, and yaw attitude data for all KLM satellites. Values are 0 for all POD satellites.*

4.5.6. Correction of scanline timestamps

The geolocation in PyGAC depends on accurate scanline timestamps. However, these may be corrupt, especially for older sensors. Assuming a constant scanning rate, PyGAC attempts to fix them using extrapolation based on the scan line number and a reference time. Finding the right reference time is difficult due to the multitude of possible timestamp corruptions. But the combination of the following three options proved to be a robust reference in many situations: timestamp of the first scanline, median time offset of all scanlines and header timestamp. Finally, not only timestamps but also scanline numbers may be corrupt. Therefore lines with erroneous scanline numbers are removed from the dataset, because it is impossible to reconstruct their original position in the dataset.

4.5.7. Scan-motor issue

Between 2001 and 2004 GAC data from NOAA-14, NOAA-15, and NOAA-16 frequently contain a significant amount of noise towards an edge of the swath, which appears as a bar-code feature within a visualised orbit. As reported by Schlundt et al (2017), section 5.2, this is probably caused by a temporary scan-motor issue. PyGAC contains an algorithm to identify and mask affected pixels within a 3x3 pixel grid box. A pixel is flagged as noisy if both of these two criteria are fulfilled at the same time:

- the standard deviation of the absolute difference between channel 1 (0.67 µm) and channel 2 (0.87 µm) is larger than 0.02
- the standard deviation of the relative difference between channel 4 (11 µm) and channel 5 (12 µm) is larger than 2.00

4.5.8. Treatment of missing TLE

By default, PyGAC uses TLE data to determine satellite angles. As those TLE data have gaps (in particular for NOAA-7 and -9), PyGAC contains a backup approach to calculate angles for affected orbits. A threshold value needs to be set in the run configuration file (pygac-fdr.yaml). This value determines the number of days within which PyGAC will use the closest TLE data available in time, and was set to 7 days for this processing. If no TLE data are available within the threshold, PyGAC uses nadir geolocation data to approximate the satellite's position. The satellite altitude is set to the value provided by OSCAR for TIROS-N, 850 km (see https://space.oscar.wmo.int/satellites/view/tiros_n).
If TLE data were available within that user-defined time window, PyGAC will add a variable to the NetCDF metadata, containing all TLE information. The user can grep for the availability of such a variable by using the unix command:

```
ncdump -h [your_pygac_orbit.nc] | grep "orbital_parameters_tle"
```

If no TLE data were available, that variable is absent. No additional flag is set in the metadata to indicate such cases.

4.5.9. Quality control and information

4.5.9.1. Per scanline quality flag

Three scanline quality flags are applied to the output files to mask out potentially corrupt data. These flags, that are common for POD and KLM families and available in the GAC data stream, are set to true if a) fatal error occurs in the orbit pre-processing, or b) calibration information is missing, or c) navigation information is missing. If any of the three flags is true, then those scan lines are masked out as missing data. These quality flags are also included in the FDR dataset so that it is transparent to the user why a scanline has been masked. Furthermore, solar blackbody contamination flags for channel 3 to 5 are included.

4.5.9.2. Global quality flag

In addition to scanline-based quality flags, a global quality flag per orbit is included in the FDR. It is raised if high-level checks fail (e.g. inconsistent timestamps or very small files) or if the file contains redundant information because it is overlapped entirely by another file (this occurs when the same data were received by different ground stations for example).

5. ALGORITHM VALIDATION

A validation report is available for the entire PyGAC FDR dataset. For further details, please see the FDR AVHRR Radiance validation report.

6. LIMITATIONS AND RECOMMENDATIONS

This is a list of known limitations, and it is very likely that further unknown limitations exist. We strongly advise that the user consults the validation report, and in particular data availability and hovmoeller plots in that report. If in doubt please also use the EUMETSAT User Service Helpdesk for questions by email: *ops@eumetsat.int*.

- Periods of data outages covering at least several days:
    - NOAA-14: 08-1988, 06-1989, 04-1990, 01/02-1991, 12-2001
    - NOAA-12: 10-1993
    - NOAA-15: 07-2000 to 02-2001
    - Metop-B: 04/05-2013
- Periods when channel 3a was turned on at daytime (3b off) and channel 3b at nighttime (3a off):
    - NOAA-15: 1999-03-09 to 1999-04-20, and during NH summer over Alaska from 2016 onwards
    - NOAA-16: until 2003-05-01, and channel 3b always on afterwards
    - NOAA-17: throughout
    - NOAA-18: 2005-06-23 to 2005-08-05, and channel 3b always on otherwise
    - NOAA-19: 2009-03-19 to 2009-05-14, and during NH summer over Alaska from 2016 onwards
    - Metop-A/-B/-C: throughout
- In the absence of Two Line Element data, angles are computed based on a fixed satellite height of 850 km. The following number of orbits are affected:
    - NOAA-6: 307 orbits
    - NOAA-8: 592
    - NOAA-9: 1385
    - NOAA-10: 3
    - TIROS-N: 94

  Without TLE data, satellite angles will probably be less accurate. We did not assess whether a switch from TLE usage to the non-TLE fallback approach introduces discontinuities into satellite angle time series. If interested, the user can identify such orbits (metadata variable "orbital_parameters_tle" does not exist in NetCDF file) and compare satellite angles to adjacent orbits.

- The orbits of most satellites are not stabilised and thus drift regarding their equator crossing time. This drift introduces artefacts into derived products such as time series and trend analyses. We thus recommend that this PyGAC FDR should be used with extreme caution for climatological applications. The PyGAC FDR itself is not a climate data record.
- There is no on-board calibration of solar channels. Instead, SNOs with MODIS data are used as an independent calibration source. For years before MODIS started operation (i.e. in and before 2000), AVHRR data are intercalibrated through AVHRR-to-AVHRR SNOs, starting with the earliest AVHRR record that was cross-calibrated to MODIS as the initial reference.

- There is no cross-calibration of thermal channels, which are calibrated only based on on-board instrumentation.
- There is no homogeneous treatment of night time observations in VIS channels. For some satellites, these are always set to 0, while for others these are close to 0. We recommend that users flag night-time VIS data as a function of solar azimuth angle.
- There is no blacklisting/filtering of irregular or potentially corrupt orbits, so the user needs to apply own quality control/monitoring metrics (see section 4.5.9). For a previous internal release of PyGAC, we found that ~6 % of orbits were affected by at least one of the issues in the following list. Some issues are probably fixed with this release, in particular those related to timestamp corruption:
    - file overly long due to corrupt l1b timestamps
    - file contains data of 2 or more orbits
    - file containing only redundant orbit fragments
    - negative orbit length, i.e. end date and time before start
    - wrong l1c timestamp
- For this release, the following periods are associated with bad l1c quality:
    - NOAA-17 between 2010-03-01 and 2011-12-31
    - NOAA-6 between 1981-08-14 and 1982-08-02
    - NOAA-8 between 1983-05-04 and 1983-09-19
    - Metop-C channel 3a from 2021 (gradually increasing reflectance); all of these time periods also appear as anomalous in the data availability plots.

We recommend that users should treat data from such orbits with care. The channel 3a trend for Metop-C is a gradual phenomenon, and affected data might still be useful for qualitative applications. However, the data listed for NOAA-6/8/17 above should be blacklisted completely.

- For this release, the FDR geocoding quality has not been assessed and the geolocation should be taken with care when using the data to derive certain level 2 land products such as snow, vegetation dynamics, lake surface water temperature etc. It is planned to include a geolocation assessment in the next release.

7. ACKNOWLEDGEMENTS

PyGAC was initially developed under the framework of ESA's Cloud Climate Change Initiative Phase II with user feedback from EUMETSAT's Satellite Application Facility on Climate Monitoring (CM SAF). PyGAC is now maintained by the Pytroll community (http://pytroll.github.io/). For this release, we used visible calibration coefficients kindly provided by NOAA (Mike Foster and Andrew Heidinger) through the SCOPE-CM collaboration between EUMETSAT and NOAA.