ESA Cloud_cci

Technical Report on AVHRR GAC FCDR generation

Issue 1 Revision 0

10 May 2017

Deliverable No.:	N.A.
ESRIN/Contract No.:	4000109870/13/I-NB
Project Coordinator:	Dr. Rainer Hollmann Deutscher Wetterdienst rainer.hollmann@dwd.de
Technical Officer:	Dr. Simon Pinnock European Space Agency Simon.Pinnock@esa.int

Document Change Record

Document, Version	Date	Changes	Originator
Draft RAFCDR_v1.0	18/04/2017	First issued version of the document.	C. Schlundt, S. Finkensieper, M. Raspaud, K.-G. Karlsson, M. Stengel
RAFCDR_v1.0	10/05/2017	Final editing and checking	R. Hollmann

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to summarize the AVHRR GAC (Advanced Very High Resolution Radiometer Global Area Coverage) Level 1 processing, including data irregularities and their handling. It describes the technical aspects of generating a multi-decadal AVHRR GAC Level 1c (L1c) dataset (1981 – 2015) based on Level 1b (L1b) GAC raw data provided by NOAA CLASS (Comprehensive Large Array-data Stewardship System). Please note that in this report L1c refers to the calibrated and Earth located product (i.e. reflectance and brightness temperature measurements), in the original pixel location together with needed engineering and auxiliary data (e.g. solar and viewing angles, quality flags).

The AVHRR instrument is a cross-track scanning system with five spectral bands (see Table 1-1) providing data in three different modes:

1. HRPT (High Resolution Picture Transmission): real-time downlink data of 1.1 km resolution. It is confined to areas where the satellite is in range of a ground receiving station.
2. LAC (Local Area Coverage): original 1.1 km resolution data recorded on-board the satellite and transferred to the ground station at a later time.
3. GAC (Global Area Coverage): reduced resolution image data. Four out of every five samples along the scan line are used to compute one average value and the data from only every third scan line are processed, yielding 1.1 km by 4 km resolution at nadir. Only this data format is used here.

The tool used to process the AVHRR GAC L1c data was developed in the framework of the ESA Cloud_cci (Hollmann et al., 2013, Stengel et al., 2013) projects. The AVHRR GAC FCDR created by employing this tool was used for the generation of climate data records of cloud properties by Cloud_cci (AVHRR-AM and AVHRR-PM datasets) and by CM SAF (i.e. for CLARA-A2 which also includes surface albedo, surface shortwave radiation, and surface longwave radiation).

Table 1-1 *AVHRR channel characteristics. AVHRR/3 has 6 channels but only transmits 5, i.e. 3a and 3b are switched (e.g., day/night).*

Band #	AVHRR/1 on-board NOAA-5,6,8,10	AVHRR/2 on-board NOAA-7,9,11,12,14	AVHRR/3 on-board NOAA-15,16,17,18,19, MetOp-1, MetOp-2
1	0.58 - 0.68 µm	0.58 - 0.68 µm	0.58 - 0.68 µm
2	0.725 - 1.10 µm	0.725 - 1.10 µm	0.725 - 1.10 µm
3a			1.58 - 1.64 µm
3b	3.55 - 3.93 µm	3.55 - 3.93 µm	3.55 - 3.93 µm
4	10.50 - 11.50 µm	10.50 - 11.50 µm	10.50 - 11.50 µm
5	Ch4 repeated	11.5 - 12.5 µm	11.5 - 12.5 µm

1.2 Definitions, acronyms and abbreviations

AVHRR	Advanced Very High Resolution Radiometer

CC4CL	Community Cloud retrieval for Climate
CLARA-A2	CM SAF cloud, albedo and radiation dataset from AVHRR – Ed. 2
Cloud_cci	ESA's cloud climate change initiative project
CM SAF	The EUMETSAT Network of Satellite Application Facilities Climate Monitoring
DWD	Deutscher Wetterdienst
ECT	Equatorial crossing time
ECMWF	European Centre for Medium Range Weather Forecast
ESA	European Space Agency
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
FCDR	Fundamental Climate Data Record
GAC	Global Area Coverage
HRPT	High Resolution Picture Transmission
KLM	NOAA-15, 16, 17, 18, 19 satellites carrying AVHRR/3
LAC	Local Area Coverage
MODIS	Moderate Resolution Imaging Spectroradiometer
NASA	National Aeronautics and Space Administration
NOAA	National Oceanic and Atmospheric Administration
POD	NOAA-7, 9, 11, 12, 14 satellites carrying AVHRR/2
pycmsaf	Python tools developed in the framework of CM SAF
PyGAC	A python package to read and calibrate NOAA AVHRR GAC L1 data.
PySTAT	A python tool to compute global daily statistics of inter-calibrated AVHRR GAC L1 data developed in the framework of Cloud_cci.
SMHI	Swedish Meteorological and Hydrological Institute
SST_cci	ESA's sea surface temperature climate change initiative project
TLE	Two-line-element

1.3 Structure of the document

The document first introduces the method and data for generating the AVHRR GAC FCDR (Fundamental Climate Data Record) dataset in Section 2.1. In the subsequent sections (2.2 -2.6) each processing step is explained in detailed. Section 3 describes the application of the FCDR and

Section 4 closes with a summary and describes the accessibility of the produced dataset. Section 5 lists for completeness the observed and fixed bugs in the GAC data record.

1.4 References

Devasthale, A., Raspaud, M., Schlundt, C., et al.: PyGAC: an open-source, community-driven Python interface to preprocess more than 30-year AVHRR Global Area Coverage (GAC) data, in progress.

Heidinger, A. K., Straka, W. C., Molling, C. C., et al.: Deriving an inter-sensor consistent calibration for the AVHRR solar reflectance data record, International Journal of Remote Sensing, vol. 31, no. 24, pp. 6493–6517, 2010.

Hollmann, R., Merchant, C.J., Saunders, R., et al.: The ESA Climate Change Initiative: Satellite Data Records for Essential Climate Variables, Bull. Amer. Meteor. Soc., 94, 1541–1552. doi:10.1175/BAMS-D-11-00254.1, 2013.

Karlsson, K.-G., Riihelä, A., Müller, R., et al.: CLARA-A1: a cloud, albedo, and radiation dataset from 28 yr of global AVHRR data, Atmos. Chem. Phys., 13, 5351-5367, doi:10.5194/acp-13-5351-2013, 2013.

Mittaz, J. P. D., Harris, A. R., and Sullivan, J. T.: A Physical Method for the Calibration of the AVHRR/3 Thermal IR Channels 1: The Prelaunch Calibration Data, Journal of Atmospheric and Oceanic Technology, 26, 5, 996, 2009.

Schulz, J., Albert, P., Behr, H.-D., et al.: 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, 2009.

Stengel, M., Mieruch, S., Jerg, M., et al.: The Clouds Climate Change Initiative: Assessment of state-of-the-art cloud property retrieval schemes applied to AVHRR heritage measurements, Remote Sens. Environ., doi:10.1016/j.rse.2013.10.035, 2013.

2. AVHRR GAC level 1 processing workflow

2.1 Method and data

The main objective of this processing is the generation of an AVHRR GAC FCDR by using upgraded visible (VIS) calibration corrections (based on MODIS collection 6 data) and removing orbits of insufficient data quality. The calibration coefficients for the infrared (IR) channels were not available at the time of processing (and writing). Currently the IR calibration is relying on nominal on-board calibration methodology, which changed significantly over time, i.e. at least four different calibration schemes have been used from TIROS-N to present. Due to this inconsistency across the whole AVHRR record, there is a clear need to redo the thermal IR calibration. Ongoing collaboration of Cloud_cci with SST_cci (at University of Reading, UK), where such inter-calibration is being developed (Mittaz et al., 2009), will allow implementation of the state-of-the-art thermal channel inter-calibration in future. It is planned to reprocess the complete AVHRR GAC L1c dataset as soon as new IR calibration coefficients based on a new physical model become available.

Figure 2-1 shows the equatorial crossing time (ECT) of NOAA and MetOp polar orbiting satellites carrying an AVHRR sensor on-board. These data have been processed covering the time period from 1980-01-01 to 2015-12-31. The last two columns of Table 6-1 show the start and end dates for each platform that has been processed here. Please note that gaps are not indicated in the table but do occur due to various reasons.

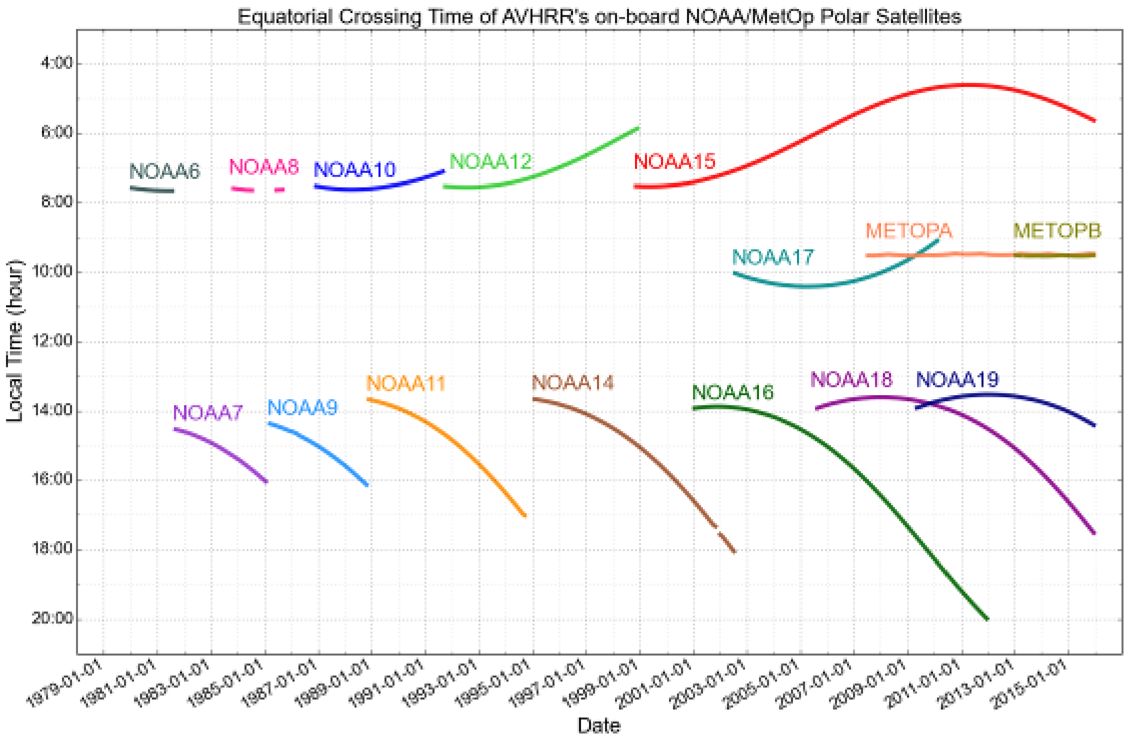


Figure 2-1 *The equatorial crossing time (ECT) of NOAA platforms changes in value over time due to orbit degradation, while the ECT of MetOp satellites remain nearly constant throughout the year because they are in controlled orbits.*

Figure 2-2 displays the processing workflow, which has been carried out to generate this AVHRR GAC L1c dataset. It turned out that several analysis steps combined in an iterative feedback loop are required for achieving the best possible quality of the data. In addition to the individual tasks the SQLite database deployed in this procedure plays a decisive role, because all important Level 1 information (e.g. orbit timestamps, ECTs, L1b file size, L1b filename, blacklist reason for poor data quality, etc.) is stored in one file, which is less than 300 Megabyte. Both CM SAF and Cloud_cci projects are using this database as main source of information on good (i.e. whitelisted) or bad (i.e. blacklisted) orbits as well as start and end scanline of each orbit considering overlapping measurements of two consecutive GAC orbits (see Section 3 for detailed explanation).

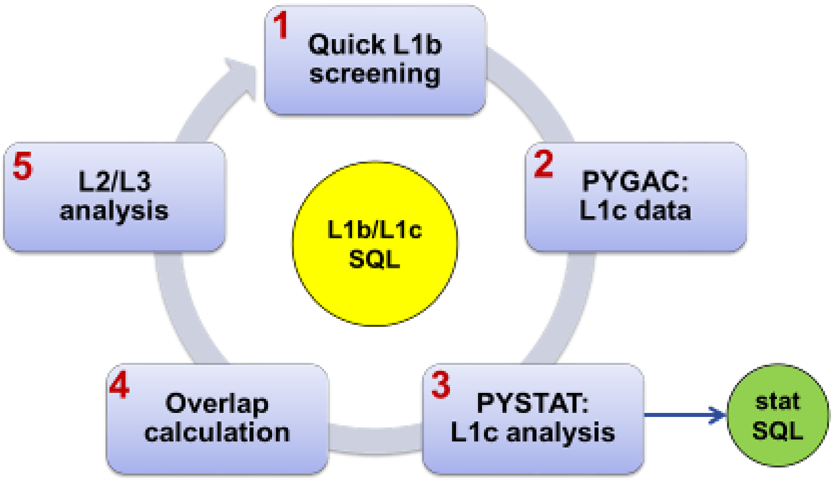


Figure 2-2 *Schematic illustration of the AVHRR GAC processing workflow. In order to ensure the highest quality of the L1c dataset several analysis cycles are required. Valuable Level 1 information is stored in a Sqlite3 (SQL) database (yellow circle). Global daily L1c statistics obtained from PySTAT are stored in a different SQL (green circle).*

The use of a SQL database offers many advantages regarding performance, portability, reliability, and accessibility:

- The application loads only the requested data, rather than reading the entire file.
- The application file is portable across all operating systems.
- Multiple processes can attach to the same database and can read and write without interfering with each other.
- A large number of programming languages provide bindings for SQLite.
- The content can be updated continuously (e.g. adding new columns, modifying entries) and viewed using a wide variety third-party tools.
- Bugs are far less likely in SQLite than in custom-written file I/O code.

The first task of the processing is a quick L1b screening (see Section 2.2) based on the filename and file size providing a whitelist of orbits, whose quality is sufficient for further usage. These L1b orbits are then passed to an open-source community-driven Python program named PyGAC (see Section 2.3), which reads and inter-calibrates the raw GAC data. PyGAC generates AVHRR GAC L1c orbit files, which are read in the third step by another Python tool named PySTAT being responsible for the computation of daily global and zonal statistics (see Section 2.4). The mean and standard deviation of the measurements together with the number of observations are helpful indicators for identifying poor data quality (i.e. additional blacklisting of orbits). In step 4 the start and end timestamps of each orbit are used to calculate the number of scan lines, which are overlapping in two consecutive orbits (see Section 2.5). This information is crucial for the generation of temporal and spatial averages (step 5) because using overlapping measurements (i.e. identical observations received by different ground stations) will lead to an overestimation of grid box mean value.

The obtained AVHRR GAC L1c data record is used by both the CM SAF and Cloud_cci project for deriving Level 2 and 3 products. Those products finally provide important feedback on the quality of the L1c data (see Section 0).

The AVHRR GAC L1b and L1c data is stored in ECMWF's File Storage system (ECFS). Table 2-1 contains the information about the ECFS paths and total size of both datasets.

Table 2-1 *AVHRR GAC Level 1b and 1c datasets stored at ECMWF's ECFS archive.*

ECMWF	AVHRR GAC L1b	AVHRR GAC L1c
ECFS path	ec:/sf3/data/GAC_avhrr_archive	ec:/sf7/data/AVHRR_GAC_L1c_archive_v2
Total size	15,07 Terabyte	22,06 Terabyte

2.2 Quick Level 1b screening

The AVHRR GAC archive at ECMWF comprises roughly 550,000 L1b orbits, which were provided by NOAA (personal communication with Andrew Heidinger). In other words, about 106 years of satellite measurements need to be processed efficiently. Hence it makes sense to screen these orbits in the first instance regarding their fitness for purpose.

As the L1b data record is stored in a tape archive, reading all L1b files would require a tremendous I/O effort. This is where the SQL database comes into play because one can operate on the entries without touching a single tape once the information from all L1b filenames has been inserted into the database. The name of an L1b file has the following format: **Processing-Center.Data-Type.Spacecraft-Unique-ID.Year-Day.Start-Time.Stop-Time.Processing-Block-ID.Source** with each qualifier defined in Table 2-2.

Table 2-2 *Description of AVHRR Level 1b filename format provided by NOAA https://www.class.ngdc.noaa.gov/glossary/AVHRR.htm.*

Qualifier	Details
Processing-Center	Three characters identifying where the data was created. NSS = NOAA/NESDIS - Suitland, Maryland, USA
Data-Type	Four characters identifying the data type and transmission method. GHRR = GAC (recorded reduced resolution AVHRR)
Spacecraft-Unique-ID	MB = METOPB, MA = METOPA, TN = TIROSN, NA = NOAA6, NC = NOAA7, NE = NOAA8, NF = NOAA9, NG = NOAA10, NH = NOAA11, ND = NOAA12, NJ = NOAA14, NK = NOAA15, NL = NOAA16, NM = NOAA17, NN = NOAA18, NP = NOAA19
Year-Day	e.g. D11001, where 'D' identifies this group as a Julian day delimiter, '11' identifies the year in which the spacecraft began recording the data set and '001' identifies the Julian day on which the spacecraft began recording the data set.
Start-Time	e.g. S0000, where 'S' identifies this group as a start time delimiter with '0000' denoting 00 hours 00 minutes UTC (to the nearest minute) and represents the time at which spacecraft recording began
Stop-Time	e.g. E0128, where 'E' identifies this group as a end time delimiter with '0128' denoting 01 hours 28 minutes UTC (to the nearest minute) and represents the time of spacecraft recording of the last usable data in the data set.
Processing-Block	e.g. B0978081, where 'B' identifies this group as a processing block ID delimiter. '0978081' is a seven digit number identifying the spacecraft revolution (orbit) in which recording of this data set began and the revolution in which the recording of the data set ended (the first five digits identifying the beginning revolution and the last two digits being the two least significant digits of the ending revolution).
Source	Two characters identifying data acquisition source, e.g. WI = Wallops Island, Virginia, USA

The database contains four tables, which are shown in Figure 2-3. All information required for quality screening and overlap calculation is stored in table 'orbits' – one row for each orbit. Columns which are not marked by an 'N' can be extracted from the L1b filename (see Table 2-3 for an example). The remaining columns are filled during/after the processing. Table 'orbits' is linked with tables 'satellites' and 'tarfiles' holding all available satellites and ground stations, respectively. The L1b files are packed into 4GB tarfiles, which are registered in table 'tarfiles'.

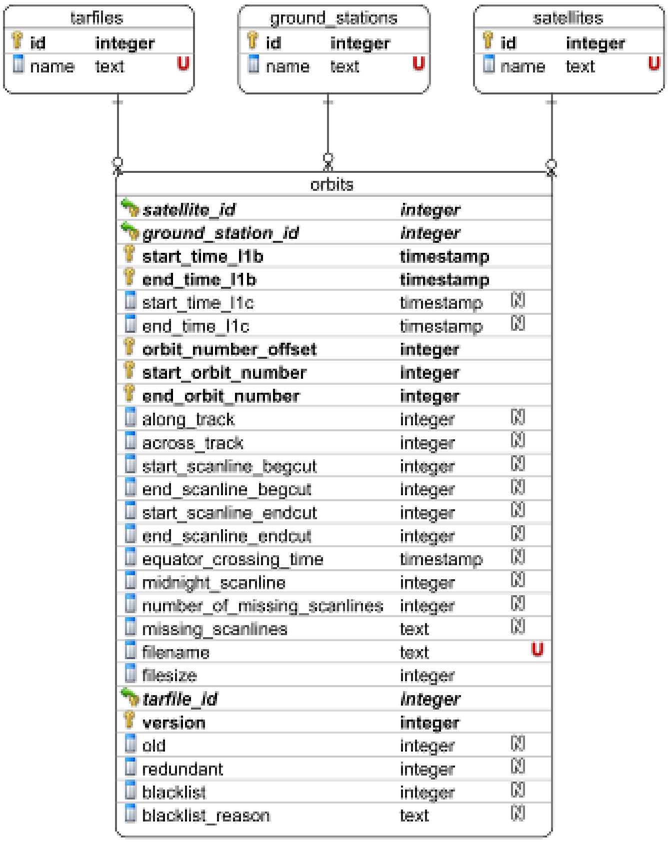


Figure 2-3 *Schematic illustration of the L1b/L1c AVHRR SQLite database storing valuable information collected during the GAC processing and analysis.*

Based on the filename information a quick L1b quality check if performed analysing timestamps, file size, and filename. Five criteria have been defined to exclude a L1b file from the actual processing because of insufficient quality (referred to as pre-processing blacklisting reasons). They are listed and explained in Table 2-4. In case an orbit matches one of these criteria, the corresponding SQL entry is blacklisted, e.g. 'blacklist' value 0 (whitelisted) is replaced by 1. The reason why it was excluded is stored in column 'blacklist_reason'.

Altogether approximately 12,000 orbits (2.1 %) were filtered based on the information from their filenames. The bulk is related to orbits which are either too small (regarding file size) or redundant.

Table 2-3 *Example of useful L1b information extracted from the filename and stored in the SQLite database.*

Example: NSS.GHRR.NP.D11001.S0000.E0128.B0978081.SV.gz	
filename	NSS.GHRR.NP.D11001.S0000.E0128.B0978081.SV.gz
filesize [byte]	30006270
tarfile_name	ec:/sf3/data/GAC_avhrr_archive/n19_2011/n19_2011_001x008.tar
satellite_name	NOAA19
ground_station_name	SV
start_time_l1b	2011-01-01 00:00:00
end_time_l1b	2011-01-01 01:28:00
orbit_number_offset	97
start_orbit_number	80
end_orbit_number	81
blacklist	0
blacklist_reason	None

Table 2-4 *List of reasons for pre-processing blacklisting of L1b orbits.*

Blacklisting reason	Explanation
old	Files which have already been used during CLARA-A1 processing but have been replaced by newer files of bigger size afterwards.
too_small	Files whose size is smaller than a certain threshold: minimum file size = 0.25 * <average file size> = 7.5 MB Inter-calibration tool PyGAC requires a minimum number of scan lines for generating the corresponding L1c file.
too_long	Files whose orbit length is greater than a certain threshold: maximum length = <average orbit length> + <corresponding stddev deviation> = 120 minutes
ground_station_duplicate	There are pairs of files in the archive, which differ in the receiving ground station but have identical start- and end-time. If both files have different sizes, the smaller file is blacklisted. If both files have the same size the first one is blacklisted.
redundant	Some files are completely covered by another file in their vicinity (regarding start- and end-time). Those redundant files can be identified and blacklisted by analysing each orbit's vicinity within a certain window: neighbourhood window size = 50

Figure 2-4 shows the number of L1b orbits as a function of file size and orbit length. One can see that the majority of orbits take about 110 minutes for recording and have an average file size of 30 Megabyte. By applying the filename based filtering outliers in size and are being excluded. The effect of other blacklisting reasons cannot be seen in this plot.

Table 6-2 summarizes the amount of orbits per satellite, which have been blacklisted before, during and after the main processing (i.e. L1c data set generation).

The remaining 98 % of all orbits are then passed to the next processing step, which provides the AVHRR GAC L1c dataset.

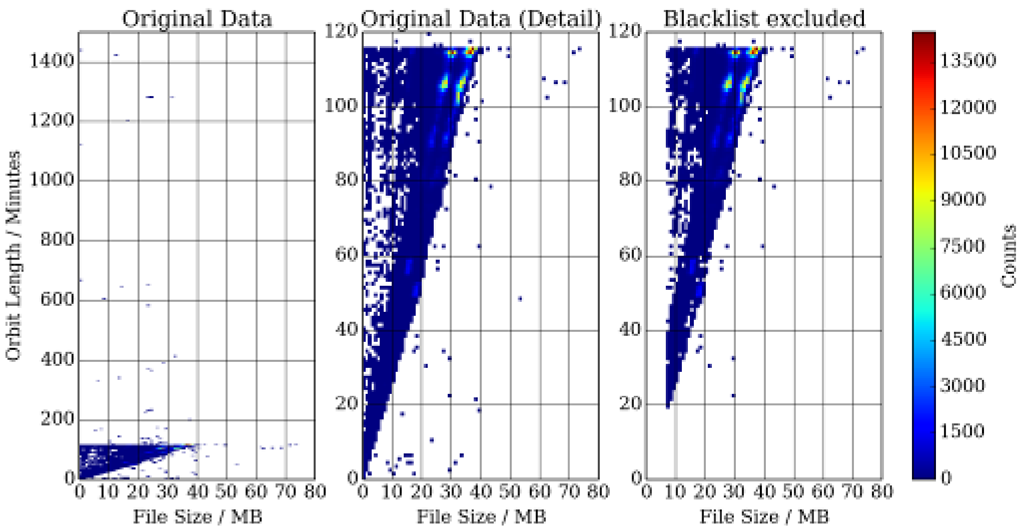


Figure 2-4 *Number of AVHRR GAC level 1b orbits as a function of file size (megabyte) and orbit length (minutes). From left to right the panels show the counts of all available orbits, same counts but with increased y-axis, and the counts of whitelisted orbits, i.e. excluding orbits of insufficient quality.*

2.3 Geo-location and inter-calibration of Level 1b data

For reading and calibrating the raw AVHRR GAC L1b data a new open-source community-driven Python interface (https://github.com/adybbroe/pygac/tree/feature-clock), referred to as PyGAC, has been developed in the framework of Cloud_cci. Here only the main important aspects are summarized. A detailed description of PyGAC is given by Devasthale et al. (2016).

For each GAC L1b orbit, PyGAC provides 3 HDF5 output files (see Figure 2-5) using the following command “python gac_run.py <L1b filename> <start scan line> <end scan line>”:

1. **ECC_GAC_avhrr_satname_99999_yyyymmddThhmmsstZ_yyyymmddThhmmsstZ.h5**

   calibrated reflectances and brightness temperatures together with geolocation (latitude, longitude);

2. **ECC_GAC_sunsatangles_satname_99999_yyyymmddThhmmsstZ_yyyymmddThhmmsstZ.h5**

   solar and viewing angles (zenith, aziumuth);

3. **ECC_GAC_qualflags_satname_99999_yyyymmddThhmmsstZ_yyyymmddThhmmsstZ.h5**

   information about the scan line records (e.g., total number of records, last scan line number) and quality.

For the calibration of the complete L1b orbit, the start and end scan line numbers have to be set equal zero. ‘ECC_GAC’ is a user specific prefix, which has to be defined in the PyGAC configuration file. ‘satname’ is the place holder for the satellite name, for instance ‘noaa18’. The value ‘99999’ is currently used instead of providing the actual orbit number. The timestamps of the first and last valid scan lines (i.e. start and end time of the actual orbit) stored in the file are the last two qualifiers with 'T' and 'Z' letters serving as delimiters. 'yyyymmdd' denotes the year, month and day, while 'hhmmsst' gives the hour, minutes, second and tenth of the second.

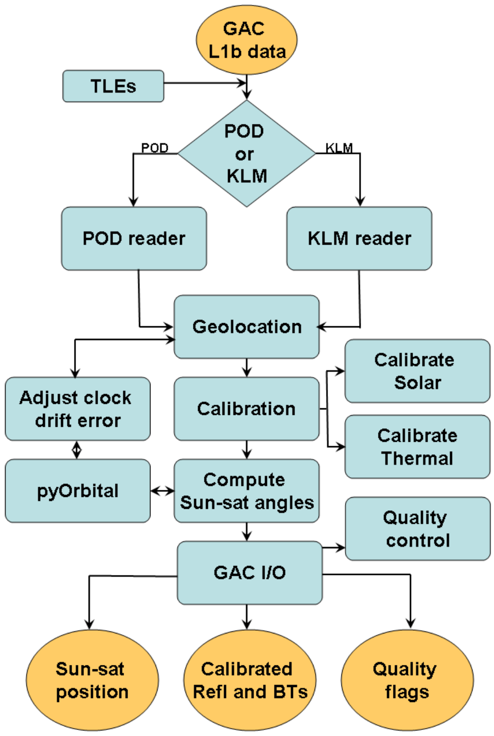


Figure 2-5 *Schematic illustration of the geolocation and calibration tool PyGAC (Devasthale et al., 2016).*

PyGAC requires Two-Line-Elements (TLEs) as auxiliary input, which contains the position and velocity of the associated satellite. POD denotes satellites that carried the first and second revision of AVHRR sensors (i.e. up to and including NOAA-14) while KLM represents those satellites carrying the third generation of the instrument (from NOAA-15 onwards). PyOrbital is a part of PyTroll family of Python interfaces designed to process meteorological satellite data. PyGAC reads the raw L1b GAC data and provides 3 HDF5 files containing the solar and satellite angles, the visible and infrared measurements and quality flags.

Before the interface can read the inserted L1b orbit, it has to determine to which satellite family it belongs, i.e. POD (up to and including NOAA-14) or KLM (NOAA-15 onwards). The L1b data format is different for both families and thus, PyGAC has to call the proper GAC reader to unpack and read 10-bit data. The POD and KLM GAC readers are based on the following NOAA user guides:

- http://webapp1.dlib.indiana.edu/virtual_disk_library/index.cgi/4284724/FID2496/podug/index.htm
- http://webapp1.dlib.indiana.edu/virtual_disk_library/index.cgi/2790181/FID3711/klm/index.htm

PyGAC is a stand-alone tool but it requires Two-Line-Elements (TLEs), which are satellite-dependent files provided by NORAD CELESTRAK (http://celestrak.com/NORAD/archives/request.asp) containing the position and velocity of the satellite.

Furthermore, the geolocation of older NOAA satellites (i.e. POD family, before NOAA-15) needs to be adjusted for onboard clock drift errors. Without this navigation correction spatial misplacement of up to 25 – 30 km can occasionally occur in GAC scenes (see Figure 2-6) influencing retrievals of climate variables, especially along the coastal zones, as the surface information is often one of the key ancillary inputs for an algorithm (e.g., land/sea mask). While some of these clock offsets are available online, not all satellites are covered. In particular, corrections are missing for the morning satellite NOAA-12. In this case, it is needed to estimate the clock drift error, and this is done by coregistration of the L1B data with a reference dataset (in our case, the Blue Marble dataset from NASA, in 500m resolution, mainly derived from MODIS data). The coregistration is performed with the scikit-image library

(http://scikit-image.org/docs/dev/auto_examples/transform/plot_register_translation.html).

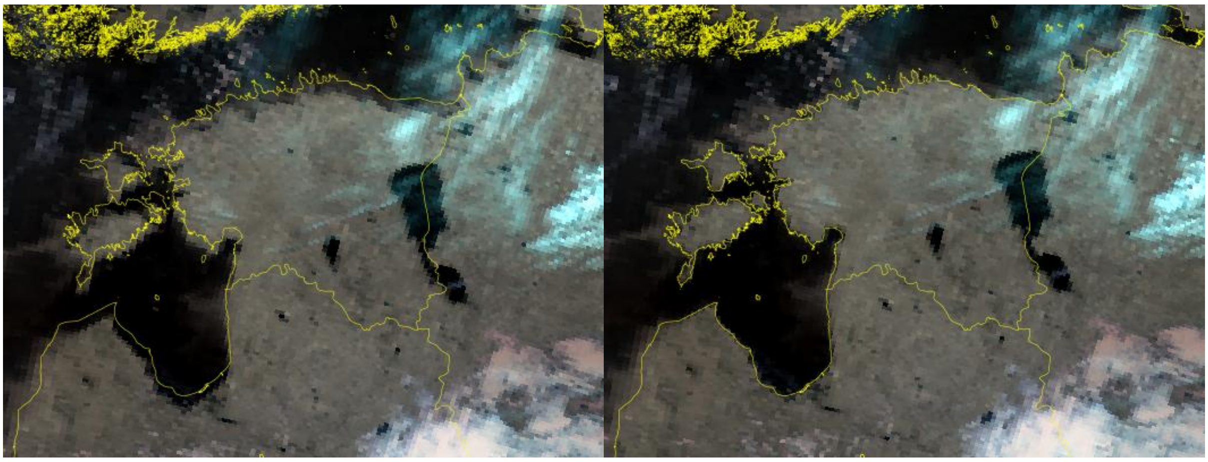


Figure 2-6 *Left: Example of mis-registered NOAA-14 orbit from 15 July 1997 at 00:42 UTC over Estonia. Right: Same case corrected for clock error.*

The orbit-to-orbit detected offsets can be quite different, and using the hypothesis that the clock drift is a slowly varying process, a piecewise linear approximation of the offset over longer periods is performed. Punctual changes in the clock values (as provided in the clock drift files from http://www.ospo.noaa.gov/) are also taken into account. Finally, this offset is used to perform interpolation of the pixels' existing longitude and latitude values where possible (using SLERP https://en.wikipedia.org/wiki/Slerp), and use the PyOrbital package (https://github.com/pytroll/pyorbital) to generate the longitude and latitudes where interpolation is not possible.

Figure 2-7 shows resulting clock errors estimated for the NOAA-12 satellite in the period 1992-1999. It is obvious from the figure that uncertainties are large here as shown by the large spread of results for individual matched orbits (grey columns). Thus, uncertainty is apparently larger for NOAA-12 than for other satellites (e.g. NOAA-14) due to the existence of additional attitude errors (i.e., the satellite pointing towards Earth is not kept sufficiently stable in orbit since the satellite is slowly wobbling). Such errors are also possible to correct for but requires an upgraded methodology.

The interface uses updated calibration coefficients for the AVHRR VIS channels, which have been provided by Andrew Heidinger under SCOPE-CM project. These values are listed in Table 6-4. The solar channel calibration takes into account inter-satellite differences and is based on different calibration references including MODIS Collection 6 data, in-situ targets, and simultaneous nadir observations (Heidinger et al., 2010). The methodology for thermal channel inter-calibration is done from scratch and is described in Devasthale et al. (2016).

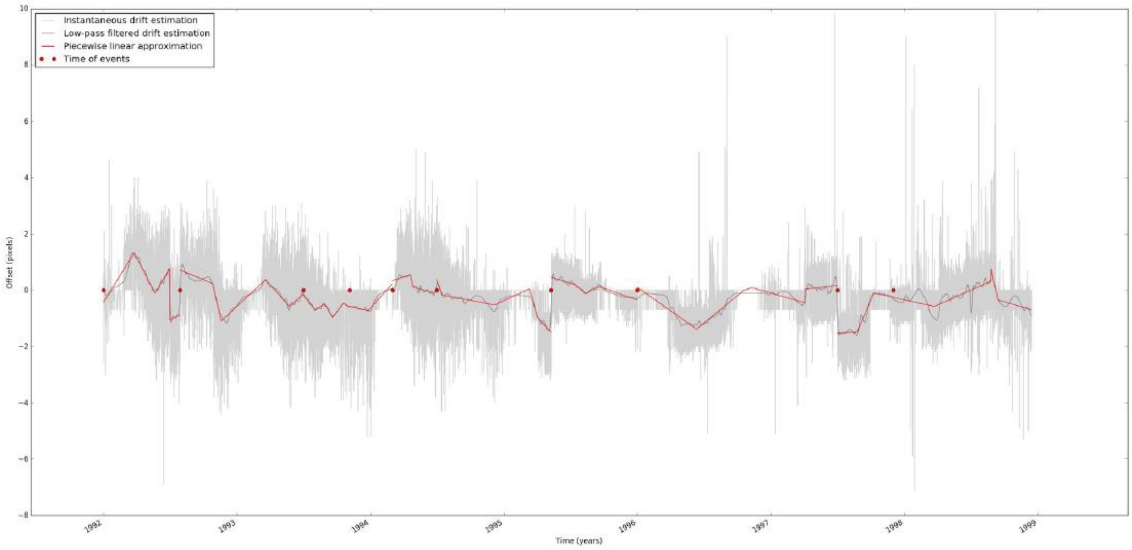


Figure 2-7 *Estimated clock drift errors for NOAA-12 in the period 1992-1999. Red line is the final piece-wise linear approximation of the clock error.*

Table 2-5 *List of SQLite entries added after creation of L1c GAC HDF5 files based on the actual L1b GAC orbit.*

Example: NSS.GHRR.NP.D11001.S0000.E0128.B0978081.SV.gz	
start_time_l1c	2011-01-01 00:00:52.400000
end_time_l1c	2011-01-01 01:27:59.400000
across_track	409
along_track	10455
number_of_missing_scanlines	15
missing_scanlines	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
equator_crossing_time	2011-01-01 13:35:16

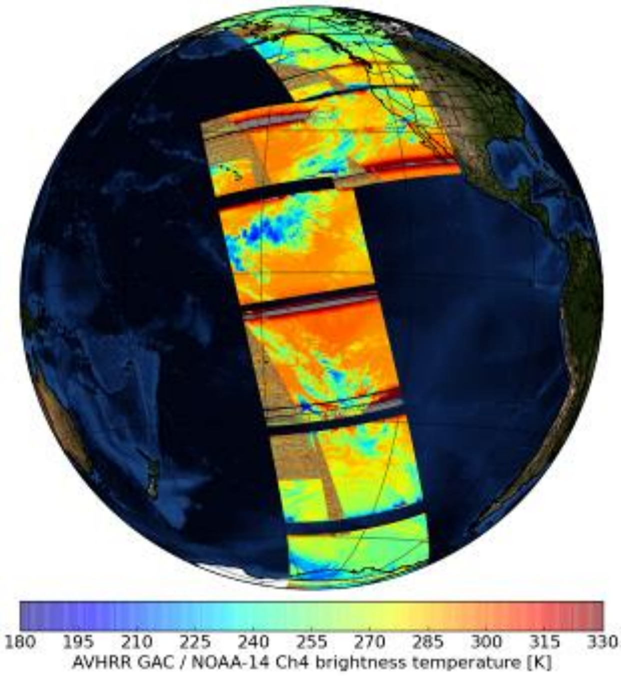


Figure 2-8 *AVHRR GAC / NOAA-14 channel 4 brightness temperature on October 20, 2001. There are two different kinds of missing data. The grey coloured areas represent invalid values in the data, which do occur in the data file as 'missingdata' or 'nodata'. The data gaps correspond to scan lines, which have never been stored in the file.*

After the L1b orbit was successfully processed by PyGAC providing the L1c HDF files, helpful information is added to the SQL database (see Table 2-5). The start and end timestamps of the L1c orbit is taken from the filename by analyzing the string. The across and along track information (i.e. number of pixels and scan lines) of that orbit is obtained by reading the file. The number and position of missing scan lines is derived from the quality flag file. This information is crucial for the GAC overlap computation as will be demonstrated in Section 2.5.

There are two different cases of missing data occurring in the AVHRR GAC data, which are displayed in Figure 2-8. The grey areas represent 'usual' missing data, i.e. the data are labeled as corrupt or missing (see 'missingdata' or 'nodata' attributes). However, in reality there are more missing lines than what is shown in grey, which is demonstrated by the data gaps where obviously several scan lines have never been stored in the GAC file. Such missing scan lines can occur anywhere in the data file.

In order to reveal missing scan lines, one has to analyse the 'qual_flags' variable in the quality file provided by PyGAC. The file has 7 columns and X rows, where X is the along track length of the orbit:

X	Number of data records in the GAC orbit
Column 1	Scan line number
Column 2	Fatal error flag: scan line should not be used for analysis
Column 3	Insufficient data for calibration: scan line should not be used for analysis
Column 4	Insufficient data for navigation: scan line should not be used for analysis
Column 5	Solar contamination of blackbody occurred in channel 3
Column 6	Solar contamination of blackbody occurred in channel 4
Column 7	Solar contamination of blackbody occurred in channel 5

If these flags are not zero, then the data should not be used. There are two important attributes in the file referred to as 'last_scan_line_number' and 'total_number_of_data_records', which are used along with column 1 for determining missing scan lines in the data. If they are equal, no scan lines are missing. Otherwise it is an indication that the data recording is incomplete. In this case the first column of the quality flag file is investigated because it contains the scan line numbering in ascending order. Positions where the scan line number jumps, yield those scan lines, which are missing in the measurement and angle HDF5 files. Figure 6-1 shows the distribution of missing scan lines per orbit for each satellite as function of time along with the total number of scan lines per orbit. The missing scan lines in each orbit are saved into the database as they are required for the overlap calculation.

Table 2-6 *Reasons of blacklisting L1c orbits based on start and end timestamp analysis.*

Blacklisting reason	Explanation
orbit_length_too_long	**Orbit length is larger than 120 minutes.**
negative_orbit_length	**start_time_l1c > end_time_l1c**

Finally, the L1c orbit duration is investigated based on the start and end timestamps (part of the filename string) provided by PyGAC. This is necessary because PyGAC computes new timestamps which are related to the first and last valid scan line of the actual orbit. However it sometimes happens that the L1b time information of the scan line used to calculate the timestamp is corrupt leading to unrealistic values (see Table 2-6)., i.e. either the orbit is longer than 120 minutes ('orbit_lenth_too_long') or the orbit start time lies before the end time ('negative_orbit_length').

The statistics on the blacklisting during the main processing is summarized in Table 6-2 for each satellite. Overall it concerns 1971 L1c orbits, which is 0.35 % of the total volume.

2.4 Level 1c quality analysis

Originally the PySTAT tool has been developed to monitor the stability and homogeneity of the calibrated reflectances and brightness temperatures. However it turned out that the daily global and zonal statistics based on the PyGAC HDF5 files are very useful to identify periods of poor data quality, which will be shown in this section.

The mean and standard deviation for each channel based on all valid satellite pixels obtained from one day are stored in another SQL database. This facilitates the access of the values for plotting purposes. Additionally, the statistics have been separated based on the solar zenith angle (SZA) for providing values for day (SZA < 90), night (SZA ≥ 90), and twilight (80 ≤ SZA < 90), respectively.

In Figure 2-9 the complete AVHRR GAC L1c statistics for channel 1 reflectance (day) is shown as time series. The top panel shows the mean values for each day and satellite, the middle panel shows the standard deviation of the mean and the bottom panel shows the number of pixels, which were used for the statistical computation.

Figure 2-10, Figure 2-11, and Figure 2-12 show the daily global statistics of the brightness temperature at 11 microns (night) for AVHRR/1, AVHRR/2, and AVHRR/3, respectively. It is obvious that outliers in the mean (and standard deviation) are related to days where the number of pixels used for the statistics is very low (e.g., NOAA-10 in Figure 2-10). Moreover, gaps in the data (i.e. no L1c data available) can also be identified from the PySTAT results (e.g., NOAA-8 in Figure 2-10).

Furthermore the PySTAT results shown in Figure 2-9 also reflect the orbital drift of the NOAA satellites, which is visible in the ECT plot in Figure 2-1. This is more pronounced in the results for the visible channels as compared to the infrared statistics. Figure 2-13 shows the PySTAT results for AVHRR channel 2 on-board METOP-A. The mean values are very stable over time due to the controlled orbit of the METOP platforms.

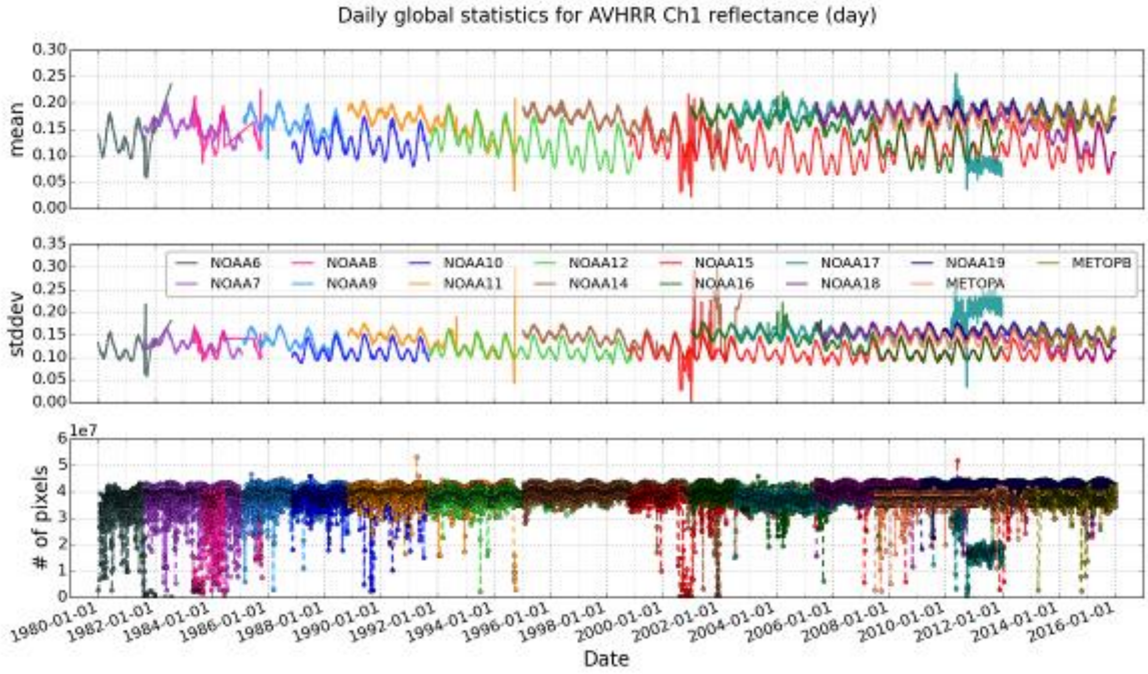


Figure 2-9 *Daily global mean (top), standard deviation (middle) and the number of observations (bottom) of AVHRR channel 1 reflectance for all satellites that have been processed.*

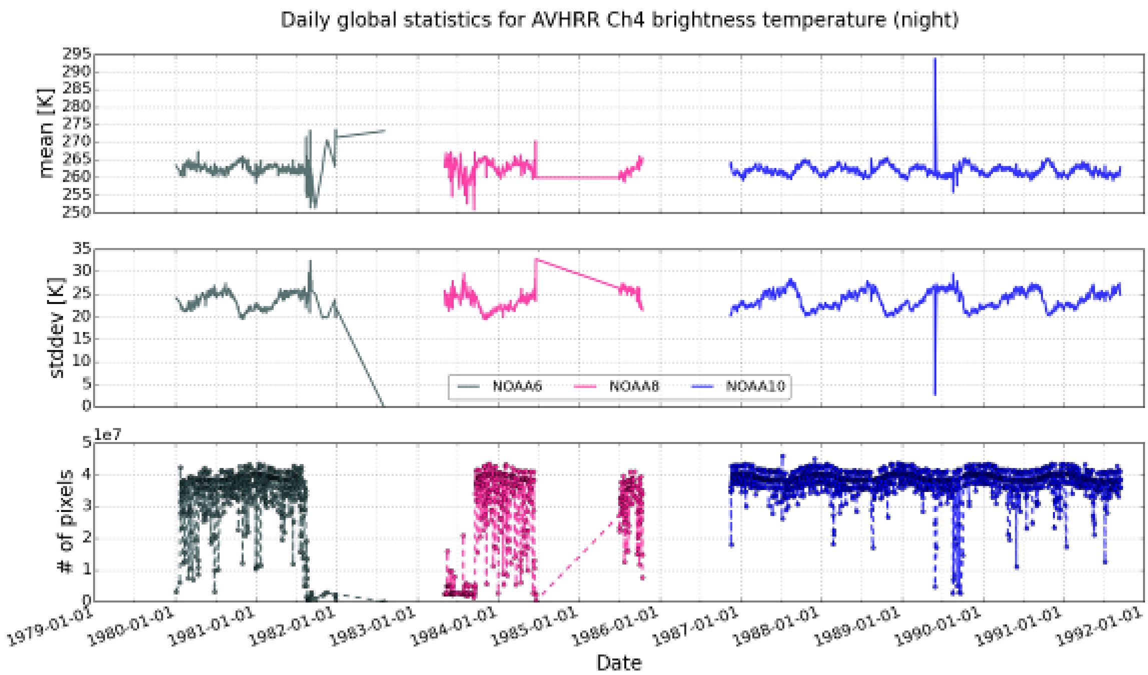


Figure 2-10 *Daily global mean (top), standard deviation (middle) and the number of observations (bottom) of AVHRR channel 4 night-time brightness temperature for NOAA-6, 8, and 10 (AVHRR/1).*

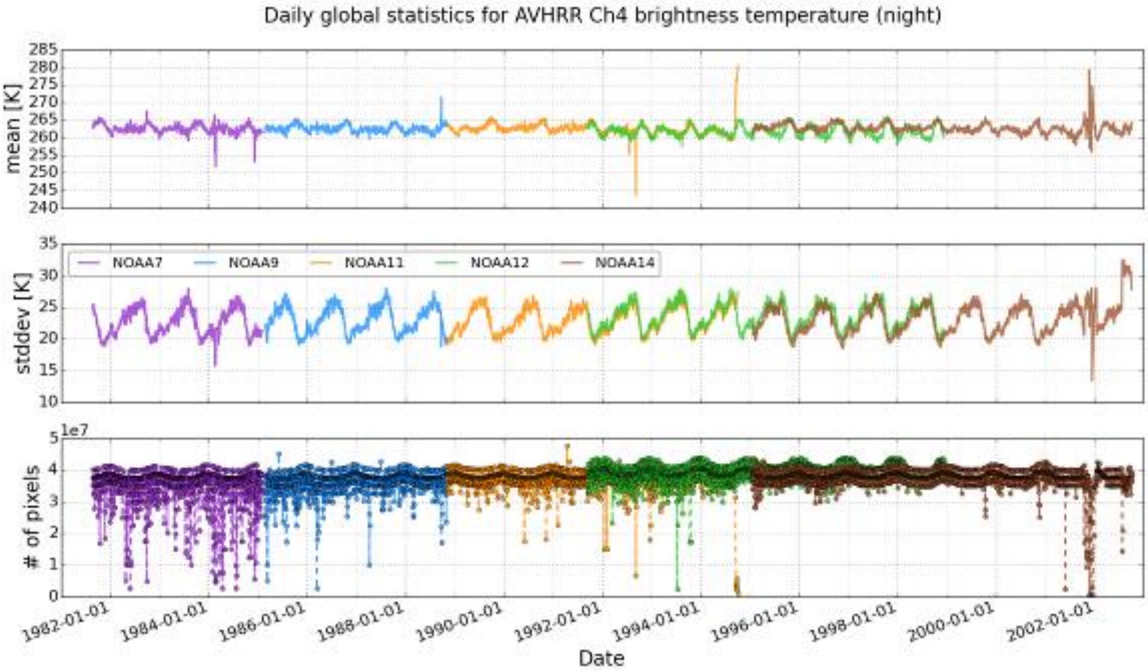


Figure 2-11 *Daily global mean (top), standard deviation (middle) and the number of observations (bottom) of AVHRR channel 4 night-time brightness temperature for NOAA-7, 9,, 11, 12, and 14 (AVHRR/2).*

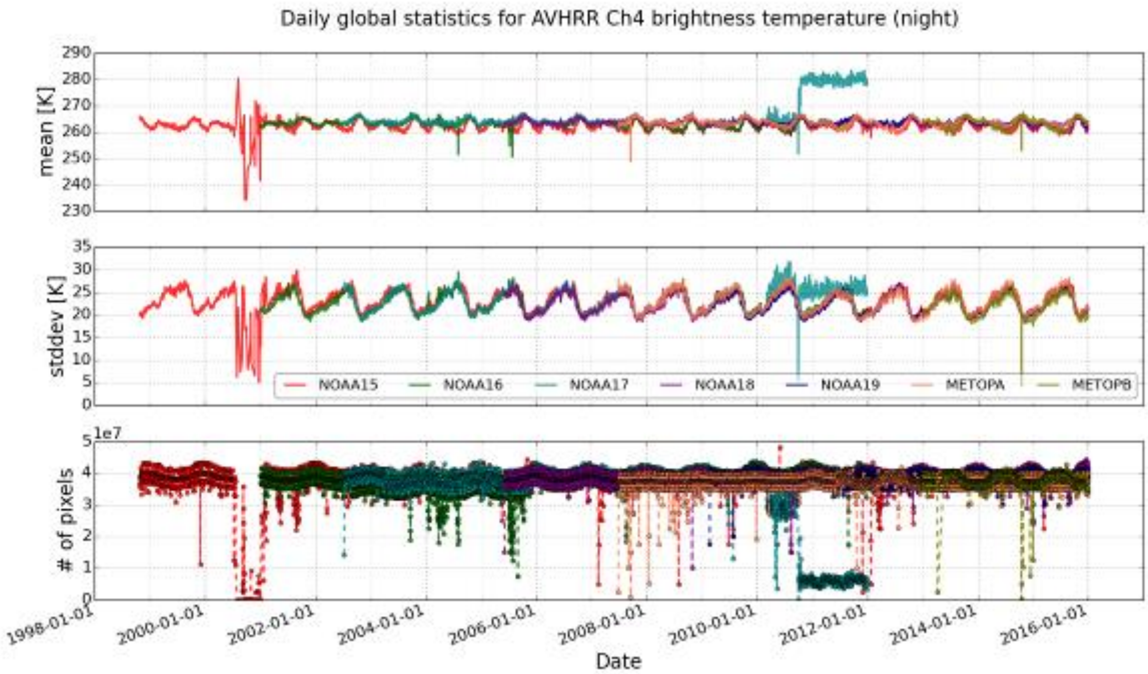


Figure 2-12 *Daily global mean (top), standard deviation (middle) and the number of observations (bottom) of AVHRR channel 4 night-time brightness temperature for NOAA-15, 16, 17, 18, 19, MetOp-A and B (AVHRR/3).*

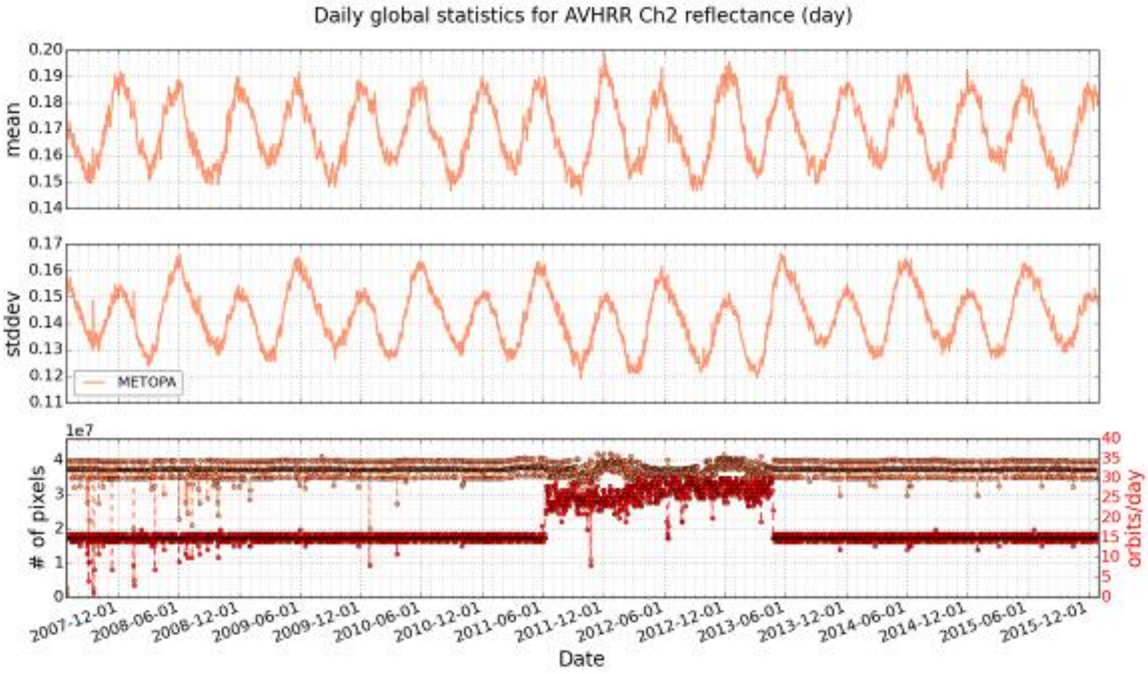


Figure 2-13 *Daily global mean (top), standard deviation (middle) and the number of observations (bottom) of AVHRR channel 2 reflectance for MetOp-A. The number of orbits per day is plotted in the bottom panel on the right y-axis (red). For almost 2 years the number of orbits has been doubled due to splitting the observations in one day and one night orbit.*

Table 2-7 *Post-processing blacklisting reasons.*

Blacklisting reason	Explanation
wrong_l1c_timestamp	Start and end timestamps derived from PyGAC do not match the original L1b timestamps, e.g. for NOAA14: file NSS.GHRR.NJ.D99287.S1459.E1645.B2468081.GC.gz start_time_l1b 1999-10-14 14:59:00 end_time_l1b 1999-10-14 16:45:00 start_time_l1c 2014-02-16 11:10:19 end_time_l1c 2014-02-16 11:23:50
pygac_indexerror	After PyGAC was updated due to a bug fix, the interface failed on 8 L1b orbits (concerning NOAA-7 and 9) and did not provide L1c HDF5 files.
no_valid_l1c_data	PyGAC did not provide valid L1c HDF5 files for the complete day although L1b orbits are available.
bad_l1c_quality	Based on PySTAT results orbits have been identified, which offer an insufficient data quality for further applications. These are: NOAA-6 1981-08-14 to 1982-08-02 NOAA-8 1983-05-04 to 1983-09-19 NOAA-17 2010-03-01 to 2011-12-31

along_track_too_long	During the CLARA-A2 processing 12 L1c orbits were identified, which have much more than 14000 scan lines, i.e. too long.
ch3a_zero_reflectance	Orbits, which offer zero reflectances in channel 3a, while all other channels provide useful data. Since CLARA-A2 cloud retrieval fails in such cases, these orbits have been blacklisted.
temporary_scan_motor_issue	Based on CLARA-A2 products and PySTAT results along with a subsequent visualization of channel 1 and 5, a list of L1b orbits have been blacklisted due to poor l1c quality. Most probably the scan motor was not working properly and thus, such orbits should not be used in further applications.

Table 2-7 summarizes and explains the post-processing blacklisting reasons, which are based on careful analysis of all processing log-files, PySTAT results, and CLARA-A2 Level 2 and 3 analyses. The statistics on the blacklisting after the main processing is summarized in Table 6-2 for each satellite. Overall it concerns 18395 L1c orbits, which is 3.29 % of the total volume. In this section the blacklisting reasons due to log-file analysis and PySTAT results are explained, while those based on L2/L3 products are described in Section 0.

26 L1c orbits have been identified, which got a wrong L1c timestamp from PyGAC, i.e. the start and/or end timestamp of the L1c HDF5 file is/are inconsistent with the L1b start and/or end timestamp. Such orbits are blacklisted as ‘wrong_l1c_timestamp’ in the SQL database and can be accessed by using the following command:

- sqlite> SELECT filename FROM vw_std WHERE blacklist_reason='wrong_l1c_timestamp';

During the GAC processing PyGAC has been improved and further developed. The interface started as a beta-version tool and after processing all the GAC data for the first time, PyGAC had undergone several updates. As a matter of course the final AVHRR GAC L1c data provided in the ECFS archive has been processed with the last and final PyGAC version providing a consistent dataset. Due to a certain bug fix related to the adjustment of the clock drift error, 8 GAC orbits (NOAA-7 and 9) failed being processed by PyGAC due to a python index error, while they have been processed before the bug fix. Therefore, these orbits are blacklisted as ‘pygac_indexerror’ in the database since they are missing in the ECFS archive.

Another blacklisting reason referred to as ‘no_valid_l1c_data’ has been found based on log-file analysis and is related to corrupt L1b data occurring for all orbits of a day. Thus, it makes sense to blacklist all these orbits since no valid L1c information is available for the complete day causing gaps in the GAC L1c dataset.

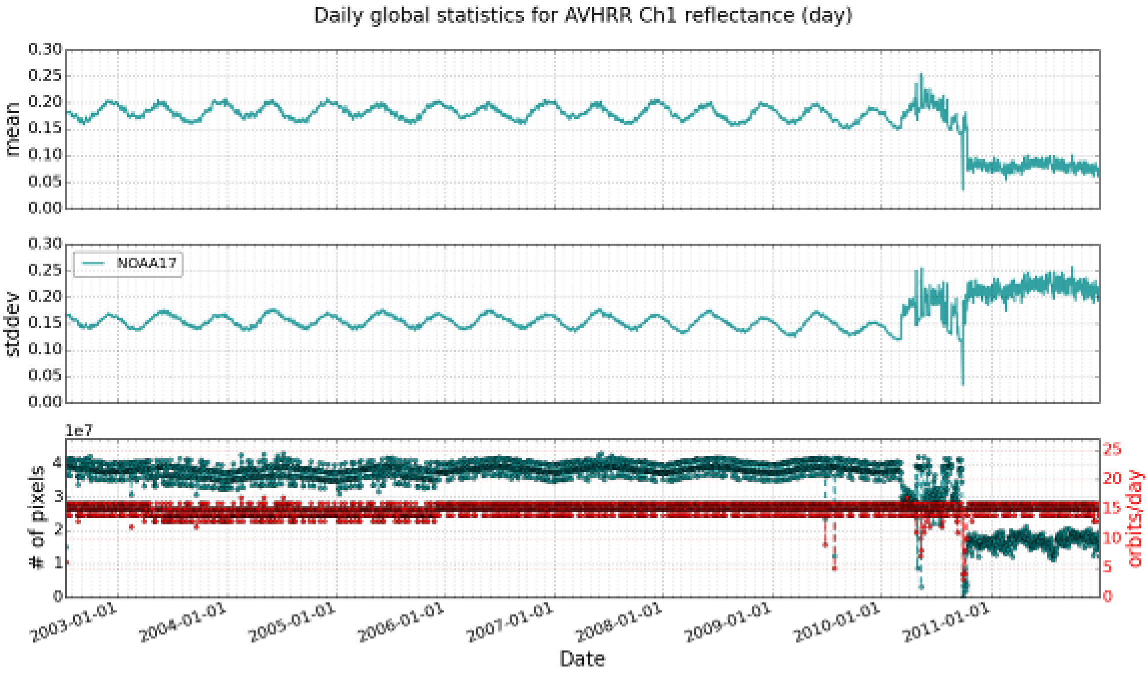


Figure 2-14 *Example for bad L1c data quality occurring from 2010-03-01 onwards for AVHRR/3 on-board NOAA-17 visible in the mean and standard deviation panels as well as in the low number of available satellite pixels for AVHRR channel 1 reflectance (day).*

Figure 2-14 shows an example for another blacklisting reason referred to as 'bad_l1c_quality' determined from PySTAT results. One can see that in spring 2010 the global daily mean is decreasing while its standard deviation is increasing due to significantly decreased number of observations. NOAA states that AVHRR on NOAA-17 stopped working on 15 October 2010 ("Stalled Motor Failure") and transmitting some dummy values. But problems with the scan motor started much earlier as can be seen in the figure. Therefore, all data from 2010-03-01 to 2011-12-31 are blacklisted in the SQL database as 'bad_l1c_quality' because they should not be used in further applications.

2.5 GAC overlap computation

In this section the method for the GAC overlap computation is explained, where missing scan lines are playing an essential role. Overlap between two consecutive orbits refers to a set of scan lines being stored both in the end of the first L1b file and in the beginning of the second L1b file. It is caused by two ground stations simultaneously receiving the same satellite signal. Usually the overlap is about 600 scan lines long (i.e. about 5 minutes). Figure 2-15 shows an example of 7 subsequent GAC orbits of two days, which are partly overlapping (hatched areas). The blue bars represent scan lines before midnight and those in green represent scan lines after 00:00 UTC.

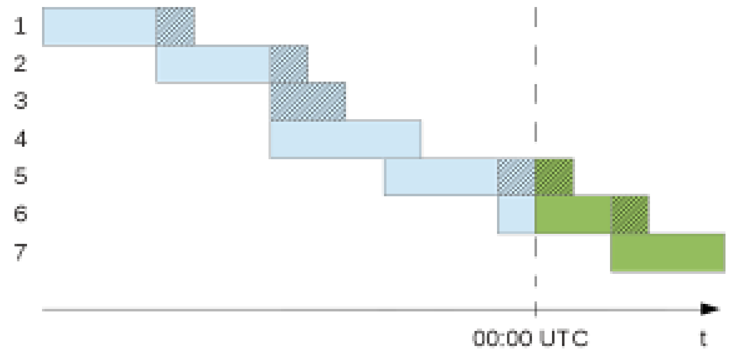


Figure 2-15 *Schematic representation of 7 subsequent AVHRR GAC Level 1 orbits of two days, which are partly overlapping (hatched areas). The bars in blue represent the orbit parts before midnight and the green ones after 00:00 UTC. The third orbit is completely covered by the fourth and thus, the 3rd orbit is blacklisted in the SQL database as 'redundant' because the same observations are included in the next orbit.*

The general idea of the overlap removal is to use the orbits' timestamps. The start and end timestamps of the L1c orbits are unique and thus, they can be used to compute the temporal overlap of two consecutive orbits: start timestamp of the next orbit minus end timestamp of the actual orbit results in the time difference of overlapping lines (in seconds). The AVHRR scanning rate is 2 scan lines per second and hence, the time difference can be translated into the number of overlapping lines.

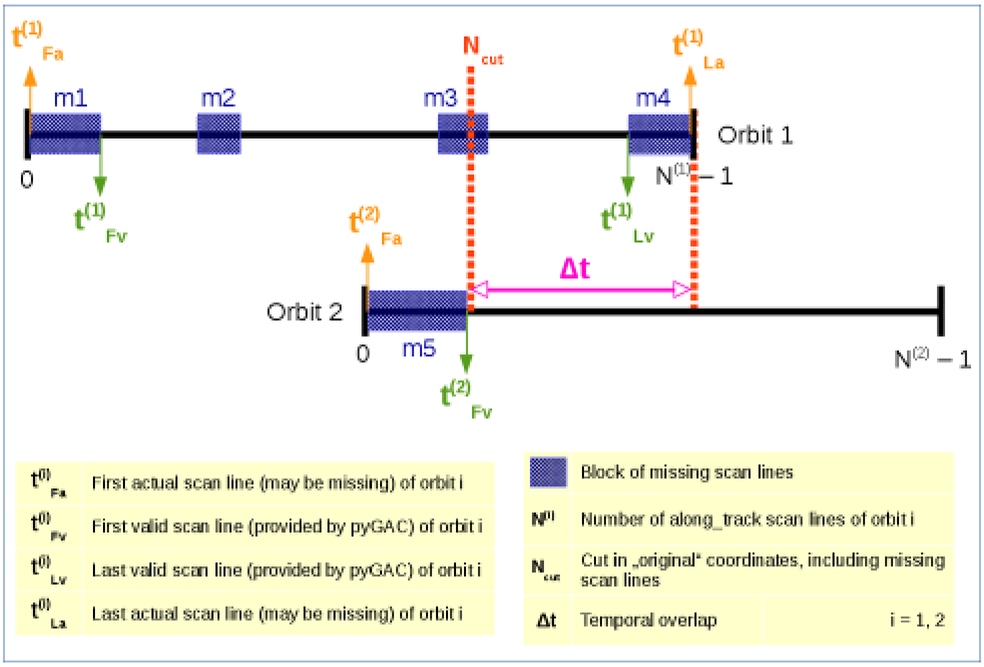


Figure 2-16 *Sketch showing two consecutive AVHRR GAC Level 1 orbit including missing scanlines, which must be taken into account for the calculation of overlapping scan lines.*

However, the assumption of a constant scanning rate (s) at 2 lines per second is only valid, if one takes all scan lines into account, i.e. also those which are missing in the file. Otherwise, the method would result in an overestimation of overlapping lines if the missing lines occur within or very close the overlapping part.

Figure 2-16 illustrates an example of two consecutive GAC orbits, which have incomplete data recordings. Orbit 1 has four blocks of missing scan lines, while the second orbit lacks of some lines at the very beginning of the orbit. The actual total number of scan lines of an orbit, $N^{(1)}$, is the sum of the valid scan lines (i.e. 'along_track') in the file and the number of missing lines not stored in the file:

$$\begin{aligned} N^{(1)} &= \sum valid + \sum missing \\ &= \text{along track} + \sum_i m_i \end{aligned}$$

Then, one has to calculate the "original" coordinates (i.e. L1b start and end timestamp of first and last actual scan line, including the missing lines) of both orbits for deriving the correct temporal overlap (Δt) between them. This is possible because the SQL database contains the number and position of all lacking lines for each orbit (explained in Section 2.3). The cut in "original" coordinates, N_{cut}, is computed as follows:

$$\begin{aligned} N_{cut} &= N^{(1)} - 1 - \Delta t \cdot s \\ &= \text{along track} + \sum_i m_i - s \cdot \left(t_{La}^{(1)} - t_{Fv}^{(2)}\right) - 1 \\ t_{La}^{(1)} &= t_{Lv}^{(1)} + \frac{1}{s} \cdot \sum \text{missing scan lines after } t_{Lv}^{(1)} \\ &= t_{Lv}^{(1)} + \frac{m4}{s} \end{aligned}$$

In the next step the time difference in seconds is translated into the number of overlapping scan lines using the scanning rate of two scan lines per second. This result is based on the assumption that both orbits have complete data recordings. Since this is not generally true, the "original" coordinates have to be transformed back to the "reduced" ones, where missing scan lines are removed again. The true number of overlapping lines, N'_{cut}, is calculated as follows:

$$\begin{aligned} N'_{cut} &= N_{cut} - \left(\sum \text{missing scan lines } \leq N_{cut}\right) \\ &= N_{cut} - \left(m1 + m2 + m3'\right) \\ &= \text{along track} + \sum_i m_i - s \cdot \left(t_{Lv}^{(1)} + \frac{m4}{s} - t_{Fv}^{(2)}\right) - \left(m1 + m2 + m3'\right) - 1 \end{aligned}$$

In this example, m3' denotes those scan lines, which are located and removed before N'_{cut}.

Table 2-8 *The GAC overlap computation uses the information about L1c start and end timestamps, missing scan lines and dimension in the along track for providing the correct start and end scan line. The start scan line is always the first one because the overlapping part is cut off at the end of the actual orbit.*

Example: NSS.GHRR.NP.D11001.S0000.E0128.B0978081.SV.gz	
midnight_scanline	None
start_scanline_endcut	0
end_scanline_endcut	9610
along_track	10455
number_of_missing_scanlines	15
missing_scanlines	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15

In summary, the number of missing scan lines and their exact positions are required for a correct GAC overlap computation. For the sake of convenience the overlapping measurements at the end of each orbit have been calculated, i.e. the SQL columns 'start_scanline_endcut' and 'end_scanline_endcut' are filled with values (see Table 2-8). In other words, the column 'start_scanline_endcut' is always set to zero (i.e. counting at zero is started here), while the column 'end_scanline_endcut' contains the last scan line before the overlap (i.e. along_track minus one and minus number overlapping lines (N'_{cut})). In case there is no overlap occurring (e.g., the next orbit is missing), the SQL column 'end_scanline_endcut' is simply the along track dimension minus one.

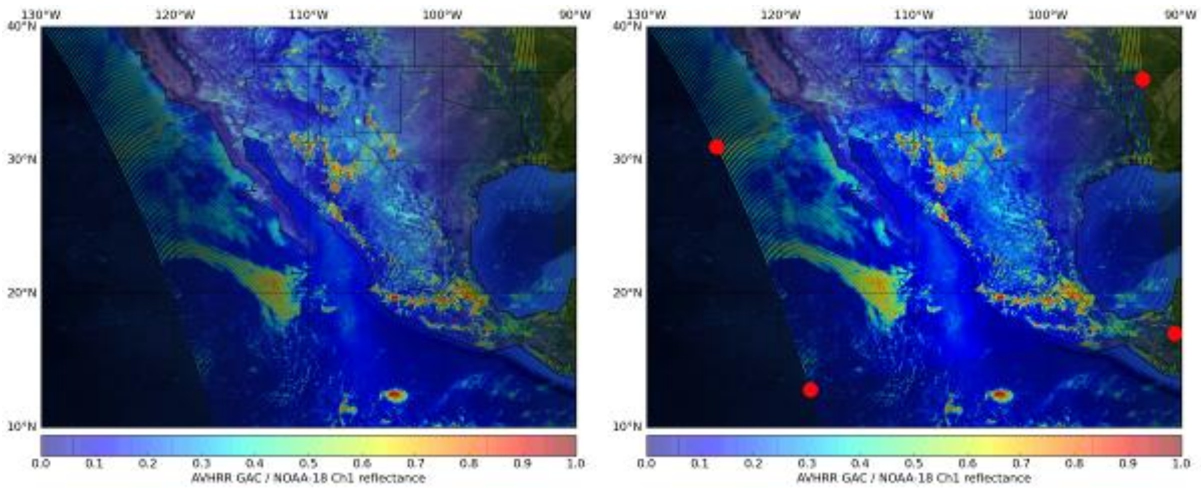


Figure 2-17 *AVHRR GAC / NOAA-18 channel 1 reflectance on June 15th 2008 of two consecutive orbits, i.e. the last part of the first orbit and the first part of the next orbit are shown. The left image shows the measurements where the overlapping scan lines at the end of the first orbit had been removed. In the right image one can see clearly that the overlapping scan lines occurring in both orbits are plotted on top of each other leading to a higher opacity (i.e. area marked with red circles).*

Generally, the last orbit of the day crosses midnight and thus, contains observations of two different days. If this is the case the SQL column 'midnight_scanline' contains the scan line which crosses 00:00 UTC. First, the midnight scan line, N_m, in original coordinates is derived by calculating the time difference in seconds between the first actual scan line, t_{Fa}, and 00:00 UTC, t_m, and multiplied with the scanning rate, s. Since start counting at zero here, 1 scan line has to be subtracted from

N_m. Then, the missing scan lines before midnight have to be removed again leading to the true midnight scan line, N'_m:

$$N_m = s \cdot (t_m - t_{Fa}) - 1$$

$$N'_m = N_m - \left(\sum \text{missing scan lines } \leq N_m\right)$$

When resampling and averaging GAC orbits on a grid, it is important to count each observation only once, i.e. remove the scan line overlap. Figure 2-17 illustrates an example of considering (left image) and neglecting (right image) the identical observations in two consecutive orbits when plotting the channel 1 reflectance of AHVRR GAC / NOAA-18 on June 15th 2008 over Central America.

2.6 Level 2 and Level 3 quality analysis

The AVHRR GAC L1c dataset is used in the CM SAF and ESA Cloud_cci projects for the retrieval of cloud products. The quality analysis of their Level 2 and Level 3 data plays an important role in the processing flow because it has revealed additional time periods of poor data quality. In this section the following three blacklisting reasons are explained: 'along_track_too_long', 'ch3a_zero_reflectance', and 'temporary_scan_motor_issue'.

During the CM SAF CLARA-A2 processing 12 AVHRR GAC L1c orbits failed because they offer much more scan lines in the along track dimension (i.e. > 14000) as the retrieval is expecting. In the case of NOAA-7 orbit (see list below) the along track dimension seems to be in the normal range, however, the 'filesize' is exceeding the average value of about 30 MB indicating that this orbit might be corrupt. Hence, these orbits have been blacklisted as 'along_track_too_long' in the SQL database and can be accessed using the following SLQ command:

- select satellite_name, filename, along_track, across_track, filesize from vw_std where blacklist_reason='along_track_too_long';

satellite name	filename	along track	across track	file size [byte]
NOAA7	NSS.GHRR.NC.D84283.S0744.E0931.B1699899.WI.gz	12739	409	64190844
NOAA15	NSS.GHRR.NK.D09357.S2056.E2242.B6037880.WI.gz	25095	409	68454887
NOAA15	NSS.GHRR.NK.D10075.S0422.E0617.B6155152.WI.gz	15845	409	44949756
NOAA15	NSS.GHRR.NK.D10298.S2008.E2150.B6473637.WI.gz	24565	409	65992491
NOAA15	NSS.GHRR.NK.D12036.S1904.E2022.B7140001.WI.gz	17240	409	45820284
NOAA16	NSS.GHRR.NL.D07100.S0445.E0638.B3375152.WI.gz	18065	409	51423416
NOAA16	NSS.GHRR.NL.D08218.S0916.E1102.B4057273.WI.gz	24993	409	71823929
NOAA16	NSS.GHRR.NL.D11317.S0738.E0932.B5744243.WI.gz	27150	409	75268449
NOAA18	NSS.GHRR.NN.D06032.S0311.E0506.B0361920.GC.gz	18420	409	51869201
NOAA18	NSS.GHRR.NN.D09253.S0123.E0318.B2219293.WI.gz	16275	409	48822364
NOAA18	NSS.GHRR.NN.D12226.S0457.E0652.B3726061.WI.gz	26719	409	77440266
NOAA19	NSS.GHRR.NP.D14087.S0248.E0316.B2645555.SV.gz	16285	409	46762129

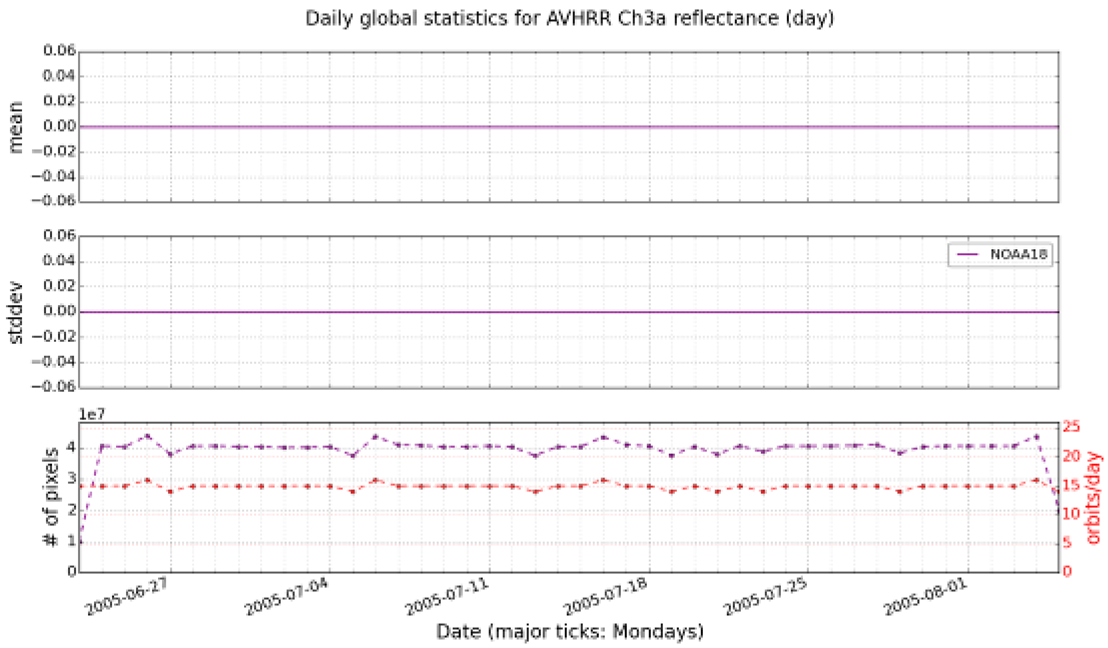


Figure 2-18 *Daily global mean (top), standard deviation (middle) and the number of observations (bottom) of AVHRR channel 3a reflectance for NOAA-18. The number of orbits per day is plotted in the bottom panel on the right y-axis (red). Note that the mean and standard deviation for channel 3a is zero between 2005-06-23 and 2005-08-05.*

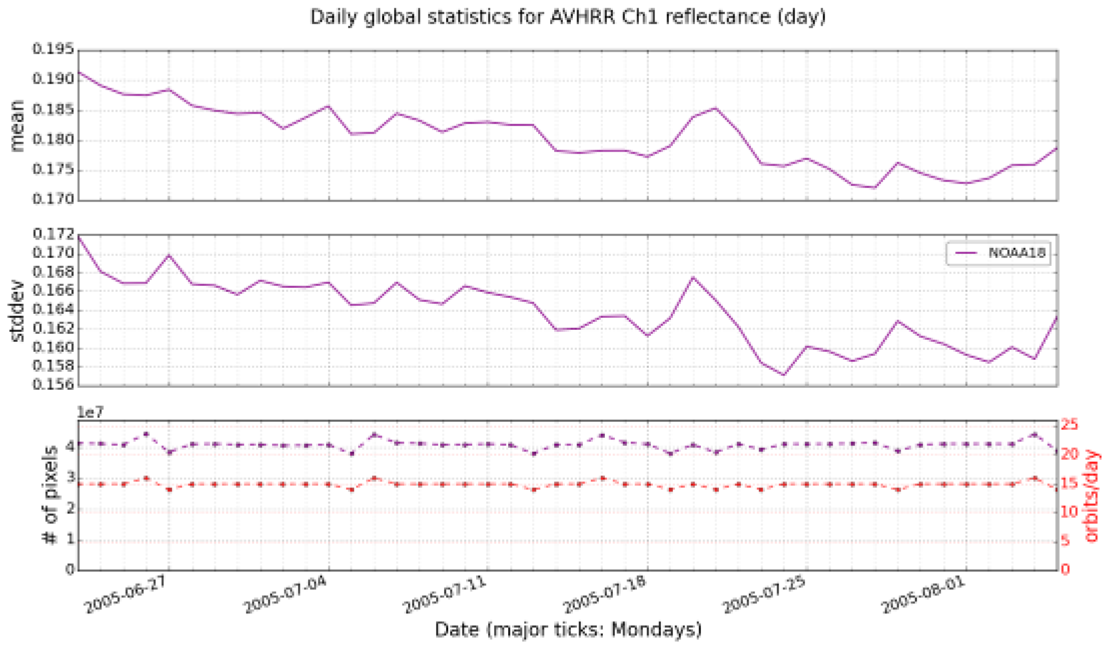


Figure 2-19 *Daily global mean (top), standard deviation (middle) and the number of observations (bottom) of AVHRR channel 1 reflectance for NOAA-18. The number of orbits per day is plotted in the bottom panel on the right y-axis (red).*

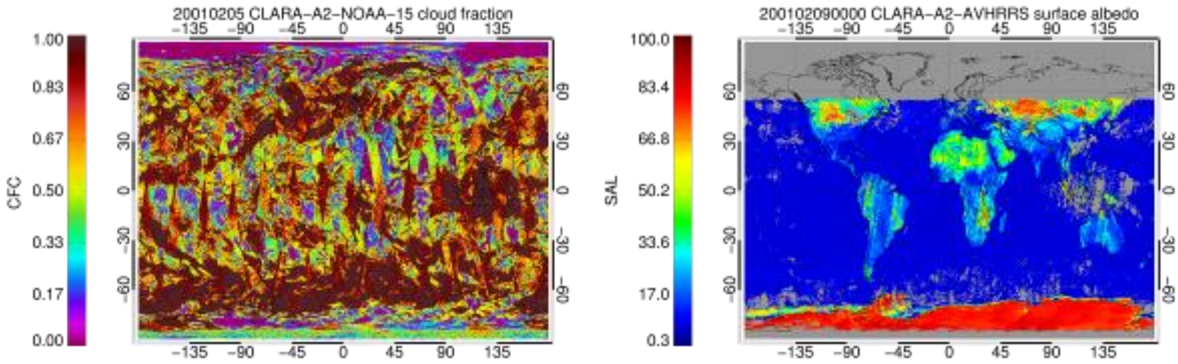


Figure 2-20 *The left image shows the cloud fraction derived from CLARA-A2 based on NOAA-15 observations on 2001-02-05, while the right image shows the CLARA-A2 surface albedo product. In both products one can see artificial stripes, which are due to insufficient AVHRR GAC L1c data used by the retrieval.*

Moreover the evaluation of the CLARA-A2 products has shown that there are some orbits, where channel 3a is activated but offer only zero reflectances, while all other channels provide useful data. In such cases the cloud mask retrieval using channel 3a leads to significant errors, which in turn impact on the subsequent derived surface albedo products. Therefore, another blacklisting reason referred to as 'ch3a_zero_reflectance' has been introduced, especially for CLARA-A2 processing. Please note that these orbits can be utilized in further applications if channel 3a is not used.

Figure 2-18 displays PySTAT daily global mean (top panel), standard deviation (middle panel) and the number of observations (bottom panel) of AVHRR channel 3a reflectance for NOAA-18 from 2005-06-23 to 2005-08-05. During this time period 618 L1b orbits have been blacklisted as 'ch3a_zero_reflectance' due to useless data provided by this channel. Figure 2-19 shows the same time period of NOAA18 but these PySTAT results are based on AVHRR channel 1 reflectances and provide evidence that other channels are not affected.

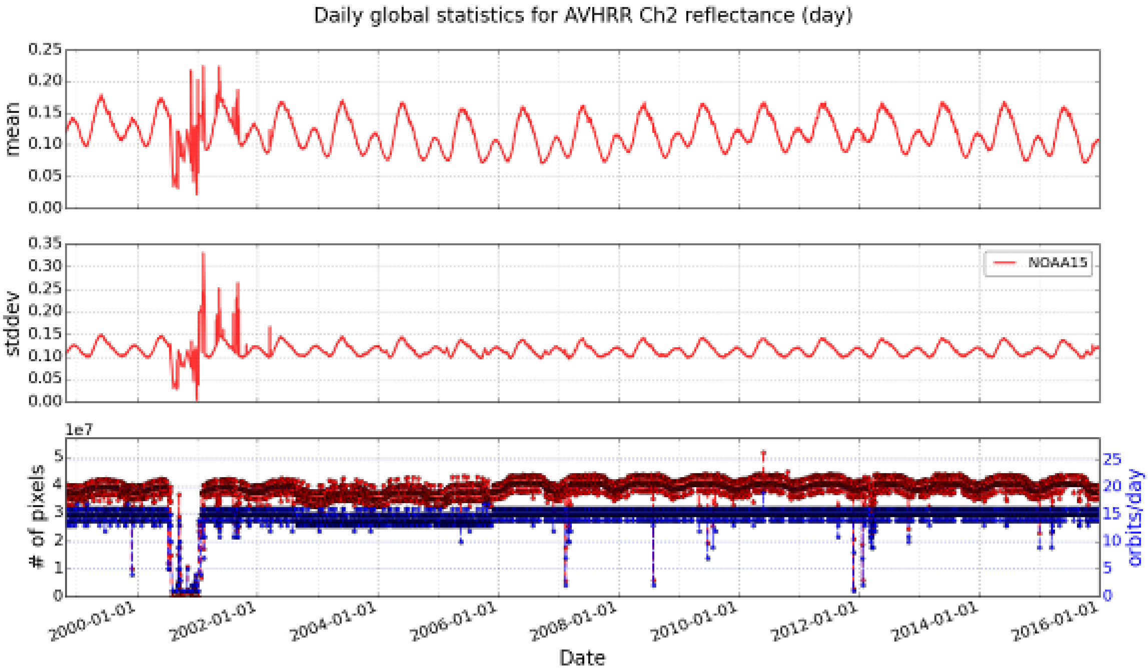
Beside GAC orbits from NOAA-18, additional 646 NOAA-15 and 804 NOAA-19 orbits are exhibiting zero reflectances in channel 3a (see Table 6-2).

Another irregularity detected through quality analysis of CLARA-A2 products is shown in Figure 2-20. The left image shows the daily mean cloud fraction based on NOAA-15 observations on 2001-02-05, while the right figure shows the surface albedo product. In both images artificial stripes are visible, which are caused by poor GAC L1c data quality used in CLARA-A2 processing.

The PySTAT results based on AVHRR channel 2 reflectances for NOAA-15 are shown in Figure 2-21. The daily global mean values and its standard deviation also indicate that the data quality around the year 2001 might be insufficient. The lower two maps in Figure 2-21 show AVHRR GAC / NOAA-15 channel 1 reflectance and channel 5 brightness temperature measurements over Africa on the 5th of February 2001 demonstrating that these orbits contain corrupt data.

Based on these findings, AVHRR channel 1 and 5 observations from NOAA-14 (2001, 2002), NOAA-15 (2000, 2001, 2002), and NOAA-16 (2004) have been plotted and analyzed by visual inspection. Most probably the AVHRR scan motor was not working properly during this time causing insufficient data recordings, which is clearly visible when plotting the data. Another example is given in Figure 2-22 for NOAA-14.

Overall 2957 AVHRR GAC L1b orbits have been selected manually and blacklisted as 'temporary_scan_motor_issue' (see Table 6-2) because they should not be used in further applications.



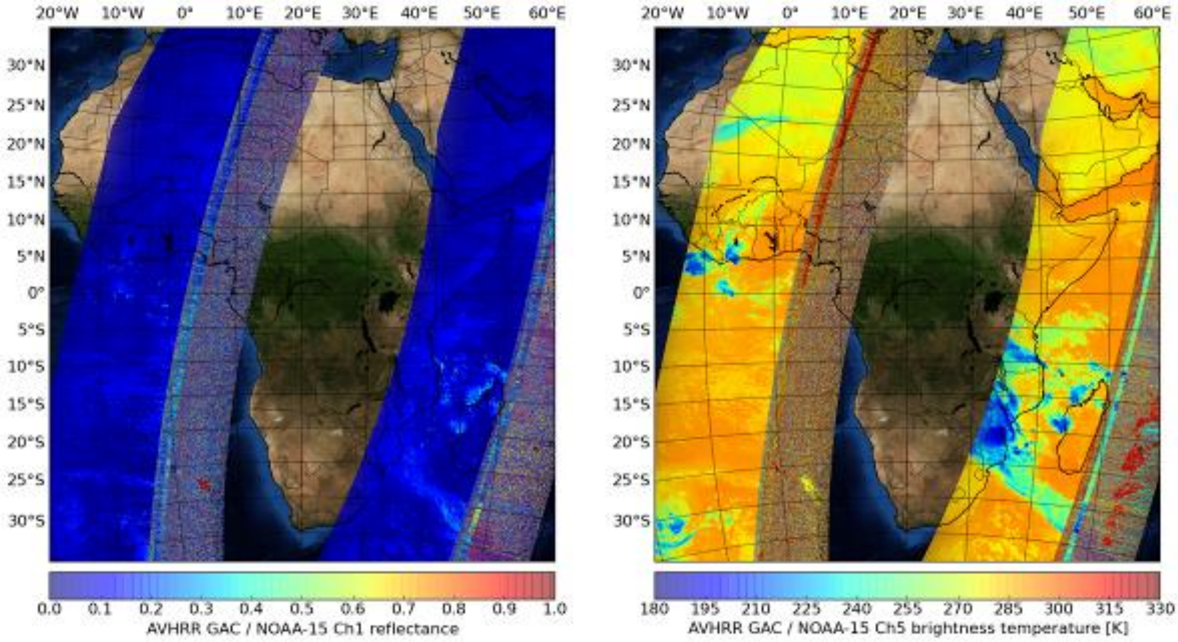


Figure 2-21 *PySTAT results based on AVHRR GAC L1c channel 2 reflectances on-board NOAA-15. During 2001-01-09 and 2002-03-19 592 orbits have been identified, which offer poor data quality, most probably due to temporary problems with the AVHRR scan motor (see blacklisting reason = 'temporary_scan_motor_issue'). Example for such orbits are shown in the lower two maps presenting AVHRR GAC / NOAA15 channel 1 reflectance and 5 brightness temperature measurements over Africa on 2001-10-20.*

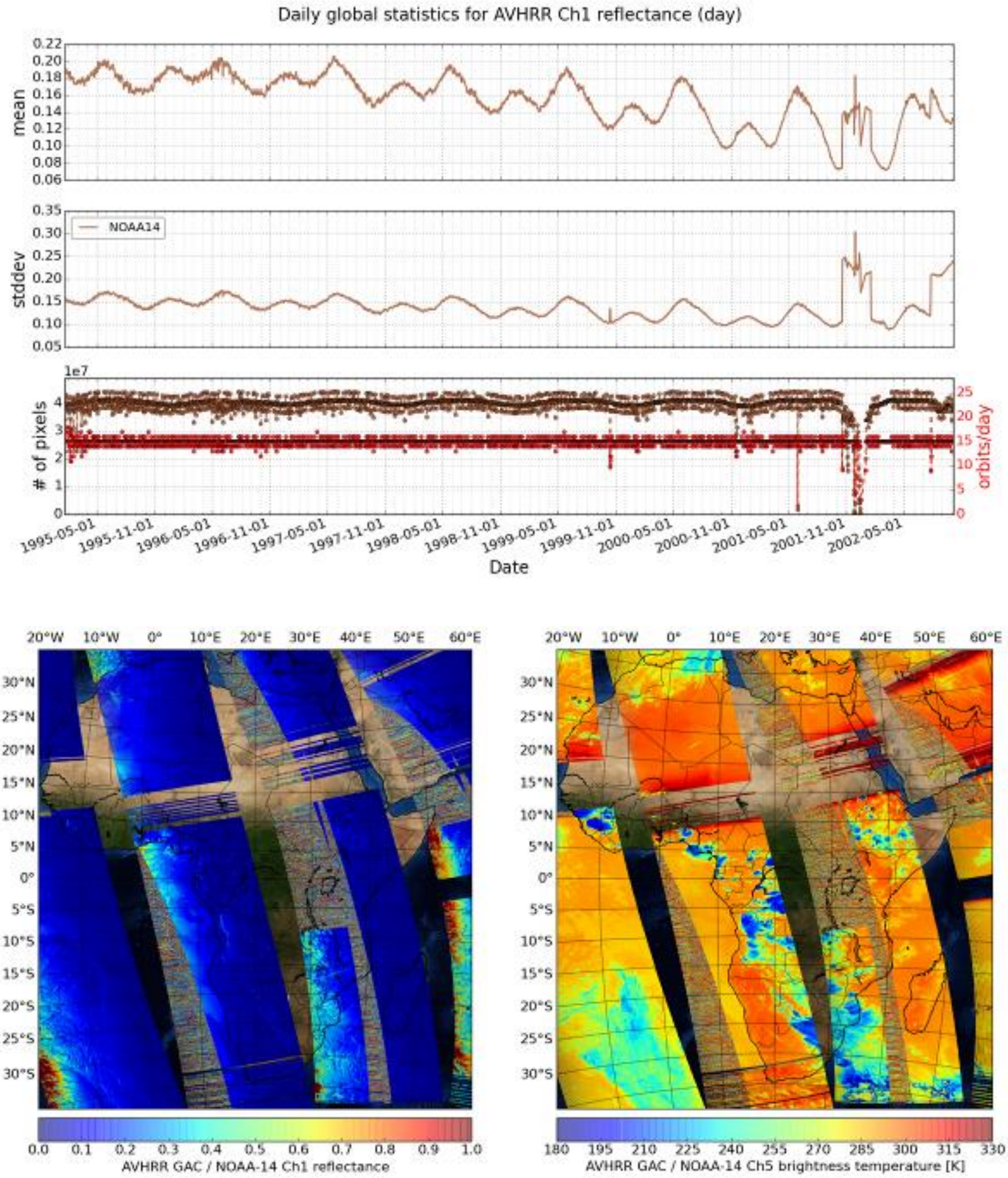


Figure 2-22 *PySTAT results based on AVHRR GAC L1c channel 1 reflectances on-board NOAA-14 (top three panels). During 2001-10-19 and 2002-10-17 1914 orbits have been identified, which offer poor data quality, most probably due to temporary problems with the AVHRR scan motor. Such orbits have been selected manually through plotting and visual investigation and have been blacklisted as 'temporary_scan_motor_issue'. Example for such orbits are shown in the lower two maps presenting AVHRR GAC / NOAA14 channel 1 reflectance and 5 brightness temperature measurements over Africa on 2001-10-20.*

3. Using the AVHRR GAC L1c data

The CM SAF and ESA Cloud_cci projects are using the AVHRR GAC L1c dataset as input for the retrieval of cloud products. However, both processing systems are not starting from the same data level due to technical and historical reasons. The CLARA-A2 algorithm starts from GAC L1b data, while CC4CL reads directly the generated L1c dataset. In both projects the inter-comparison of Level 2 and 3 data with each other is planned and hence, differences due to inconsistency in the dataset used or overlap information should be avoided. At this point the SQL database was introduced for serving as reference source regarding blacklisted orbits and overlap information.

Table 3-1 summaries the main important python front end tools, which have been developed in the course of AVHRR GAC L1c processing and are mostly accessible via GitHub. The PyGAC repository is freely available. The latest PyCMSAF release can be obtained from CM SAF (mailto: contact.cmsaf@dwd.de). The pytAVHRRGACl1c repository contains the PySTAT tool and several other python scripts, for instance, a tool for visualizing L1c GAC data or the GAC overlap calculation script. Please note that the pytAVHRRGACl1c repository has dependencies on PYCMSAF and thus, its scripts can not be used without the installation of PYCMSAF.

Table 3-1 *GitHub repositories containing python tools, which have been developed in the course of AVHRR GAC L1c processing. The 'gac_run.py' is the stand alone PyGAC tool and is freely available. The 'gacdb_client.py' is located under the PYCMSAF repository, which is password protected. The PySTAT tool 'run_pystat_add2sqlite.py' is freely available under the pytAVHRRGACl1c repository, which has dependencies on PYCMSAF.*

Python tool	GitHub repository
gac_run.py (PyGAC)	https://github.com/adybbroe/pygac/tree/feature-clock
run_pystat_add2sqlite.py (PySTAT)	https://github.com/cschlund/pytAVHRRGACl1c.git
gacdb_client.py (pycmsaf)	None yet

For accessing the required information stored in the SQL database, a python front end tool referred to as 'gacdb_client.py' (see Table 3-1) has been developed and is used by CLARA-A2 and CC4CL algorithms for obtaining the start and end scan lines for each whitelisted orbit, for example:

- python gacdb_client.py --dbfile=AVHRR_GAC_archive_v2_201603_post_overlap.sqlite3 get_scanlines --date=20110102 --sat=NOAA18 --mode=l1c --header

Table 3-2: *resulting table based on the application of gacdb_client.py*

Start-Time l1c	End-Time l1c	Start-Scanline	End-Scanline
20110101T2235282Z	20110102T0030102Z	10144	13109
20110102T0024432Z	20110102T0208177Z	0	11769
20110102T0202482Z	20110102T0357327Z	0	13109
20110102T0352032Z	20110102T0546477Z	0	13109
20110102T0541182Z	20110102T0734527Z	0	12979
20110102T0729282Z	20110102T0924102Z	0	13109
20110102T0918432Z	20110102T1113277Z	0	13109
20110102T1107582Z	20110102T1231277Z	0	9359
20110102T1225582Z	20110102T1420402Z	0	13109

Start-Time l1c	End-Time l1c	Start-Scanline	End-Scanline
20110102T1415132Z	20110102T1552577Z	0	11069
20110102T1547282Z	20110102T1742127Z	0	13109
20110102T1736432Z	20110102T1857577Z	0	9089
20110102T1852282Z	20110102T2047102Z	0	13109
20110102T2041432Z	20110102T2230127Z	0	12369
20110102T2224482Z	20110103T0019327Z	0	11423

In this example, the front end tool provides the start and end scan line numbers of 15 whitelisted orbits for NOAA-18 on 2011-01-02. The first and the last orbit are crossing the date line and thus, the provided scan line values correspond to those observations in the files, which only belong to 2011-01-02.

For the sake of convenience the overlapping measurements at the end of each orbit is taken into account, i.e. using the provided the start and end scan lines one ignores always the overlap at the end of the orbit.

The CLARA-A2 algorithm uses the start and end scan line numbers as input parameters to call PyGAC along with the associated whitelisted L1b orbit providing the corresponding L1c file without overlapping measurements regarding the subsequent orbit. For orbits crossing midnight, the L1b file is passed to PyGAC twice, i.e. considering the part of the orbit before and after midnight separately resulting in two different L1c files.

The CC4CL processing system applies the start and end scan lines for selecting only this part of the L1c orbit, which does not contain the same observations included at the beginning of the next orbit. In case the data recording of the orbit crosses the date line, the midnight scan line value gives the proper start and end line of the file (see first and last entry in Table 3-2).

4. Summary and accessibility of AVHRR GAC FCDR

This document summaries the processing flow of the AVHRR GAC L1c generation carried out by a GAC L1c processing tool developed in the framework of the ESA Cloud_cci projects. The processing system involves multiple steps for ensuring high data quality:

1. **Quick L1b screening**: investigation of filename and file size of each L1b orbit providing a whitelisted L1b dataset
2. **PyGAC**: generation of L1c dataset based on whitelisted L1b files
3. **PySTAT**: daily global statistics of L1c data for monitoring the data quality
4. **GAC overlap computation**: two consecutive GAC orbits generally overlap by about 600 scan lines, i.e. identical measurements stored in two different orbit files, and thus, the start and end scan line considering this overlap are calculated
5. **Level 2 and Level 3 quality analysis**: investigation of cloud and surface products based on AVHRR GAC L1c dataset provided by CLARA-A2 (CM SAF) and CC4CL (Cloud_cci) retrievals for identifying poor L1c data quality impacting their retrieval results.

A detailed description of each task is given in Section 2. High quality of the generated dataset has been achieved trough several processing events, i.e. step 1 to 5 being part of an iterative procedure.

Emphasis has been placed on excluding orbits, which offer poor quality and thus, should not be used in further applications. Different criteria have been defined for blacklisting L1b orbits before, during and after the generation of the L1c data set (Figure 4-1, Figure 4-2). Overall, 6% of the orbits have been blacklisted based on results from logfile analysis, daily global statistics, and retrieval products using this dataset.

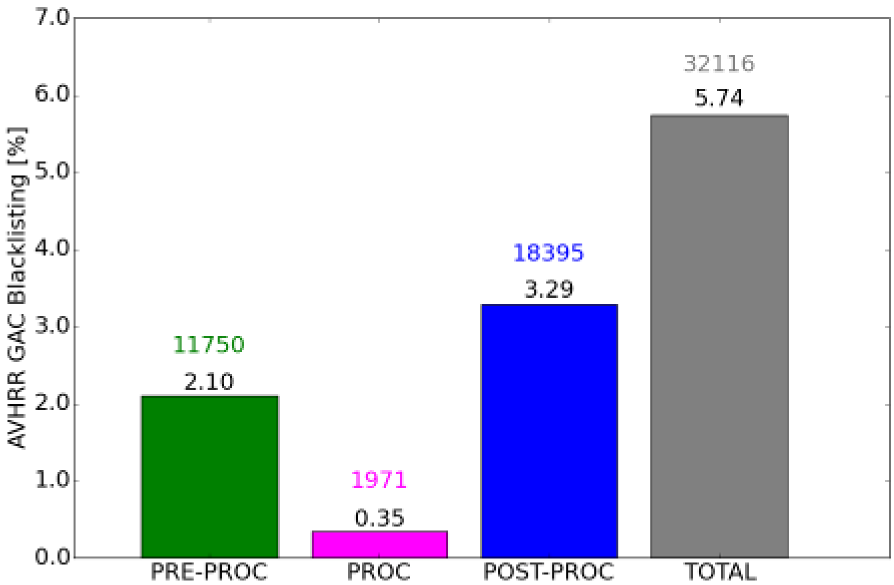


Figure 4-1 *Statistics on blacklisted GAC L1b orbits considering all AVHRR's. The values in colour are the number of blacklisted orbits, while those in black represent their percentage compared to the complete dataset. About 6 % of the orbits have been identified offering insufficient data quality and thus, have been excluded from the AVHRR GAC FCDR. Detailed statistics, i.e. how many orbits are blacklisted due to a specific criterion, is shown in Figure 4-2.*

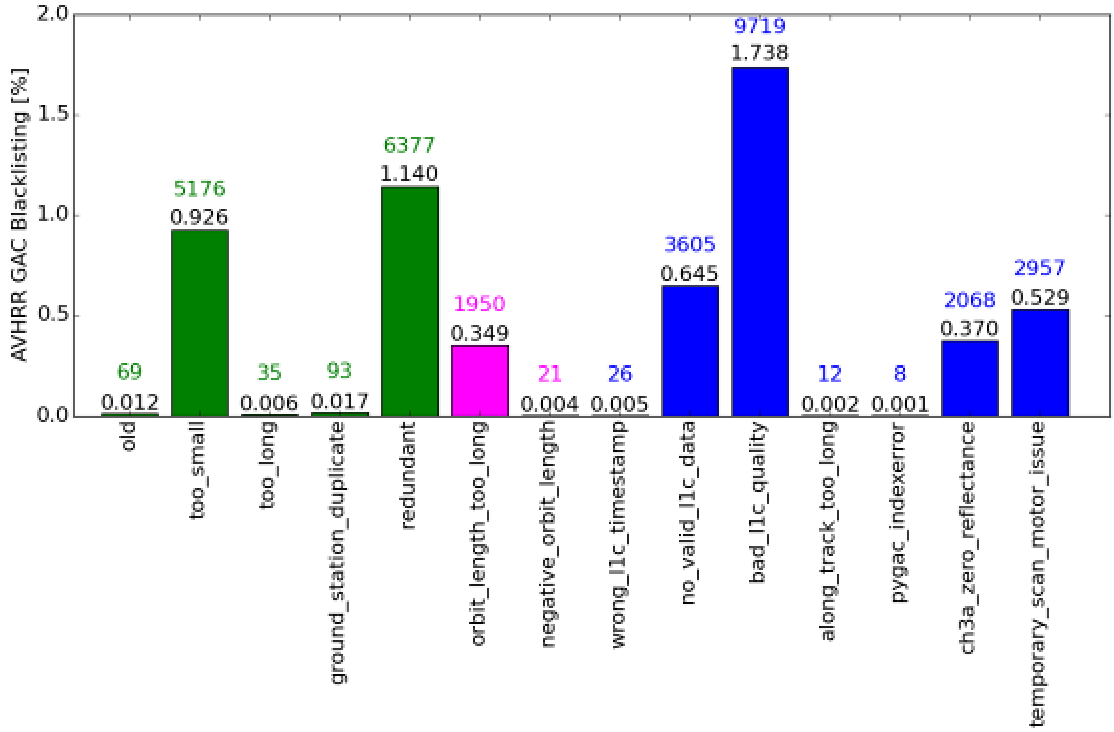


Figure 4-2 *Statistics on blacklisting criteria selected before (green), during (magenta) and after (blue) the main L1c processing. The values in colour are the number of blacklisted orbits, while those in black represent their percentage compared to the complete dataset.*

A SQLite database (Table 4-1) containing AVHRR GAC L1b and L1c information (e.g. start and end timestamps of the orbit, start and end scanline regarding overlap, filename, file size, blacklist reason due to corrupt data, along and across track, etc.) has been chosen as reference source for both projects in order to ensure the identical data used when inter-comparing the Level 2 and Level 3 cloud products with each other. Another advantage of the SQL file is that the overlap computation of consecutive orbits is carried out offline using SQL entries, which takes a few minutes for the complete AVHRR GAC dataset because there is no data I/O.

Table 4-1 *AVHRR GAC L1b and L1c datasets and the associated SQLite database stored at the ECFS tape archive at ECMWF.*

AVHRR GAC Data	ECFS path
L1b dataset	ec:/sf3/data/GAC_avhrr_archive
L1c dataset	ec:/sf7/data/AVHRR_GAC_L1c_archive_v2
Latest SQLite database	ec:/sf7/data/AVHRR_GAC_L1c_sqls/v2_201603/ AVHRR_GAC_archive_v2_201603_post_overlap.sqlite3

The AVHRR GAC L1c processing has been carried out on the cray machines at ECMWF using the ecFLOW work flow package that enables users to run a large number of programs (with dependencies on each other and on time) in a controlled environment. First, the L1b files are dearchived from the ECFS tape archive. Then PyGAC produces the corresponding L1c data. Before archiving the L1c files into ECFS, the daily global statistics is computed and stored in a particular SQL database containing only the PySTAT results.

The AVHRR GAC L1b dataset (1981 – 2015) has been downloaded from NOAA CLASS (Comprehensive Large Array-data Stewardship System). Please note that in this report L1b refers to the raw GAC data (i.e. photon counts), while L1c refers to the calibrated and Earth located product (i.e. reflectance and brightness temperature measurements), in the original pixel location together with needed engineering and auxiliary data (e.g. solar and viewing angles, quality flags).

The GAC L1b data in the ECFS archive are available as tarballs containing all orbits of a satellite covering about 8 days, e.g. ec:/sf3/data/GAC_avhrr_archive/m01_2013/m01_2013_001x008.tar

The GAC L1c data in the ECFS archive are available as monthly tarballs per satellite, e.g. ec:/sf7/data/AVHRR_GAC_L1c_archive_v2/2008/06/AVHRR_GAC_L1C_NOAA18_200806.tar, which contain daily tarballs, e.g. AVHRR_GAC_L1C_NOAA18_20080601.tar.bz2, in order to achieve high level of compression.

5. Bugfixes

Having the next release in mind, some major problems discovered during CLARA-A2 and Cloud_CCI were addressed by the following PyGAC bug fixes.

5.1 Scanline Timestamps

As described in section 2.3 and 2.4, scanline timestamps are often erroneous, which affects the output filenames and the overlap computation. Most of such corruptions could be fixed using the concept of 'ideal' scanline numbers:

Assuming a constant scanning rate *s* (2 scanlines per second), one can calculate an 'ideal' scanline timestamp $tn = t0 + n/s$ based on the scanline number $n=1, 2, \ldots, N$ which is also stored in the GAC data. Scanline timestamps deviating more than a certain threshold from the corresponding 'ideal' timestamp are replaced by the ideal value. The offset t0 is estimated using a statistical approach with boundary conditions defined by the header timestamp. The boundary conditions are required because there are cases where the majority of timestamps is corrupted.

Unfortunately, cases have been discovered where even the scanline number is corrupted. That is why some simple scanline number corrections were added before the actual timestamp correction. They make use of the strictly monotonic properties as well as the limited valid range of the scanline number. Figure 5-1 shows an example of erroneous timestamps and the applied correction.

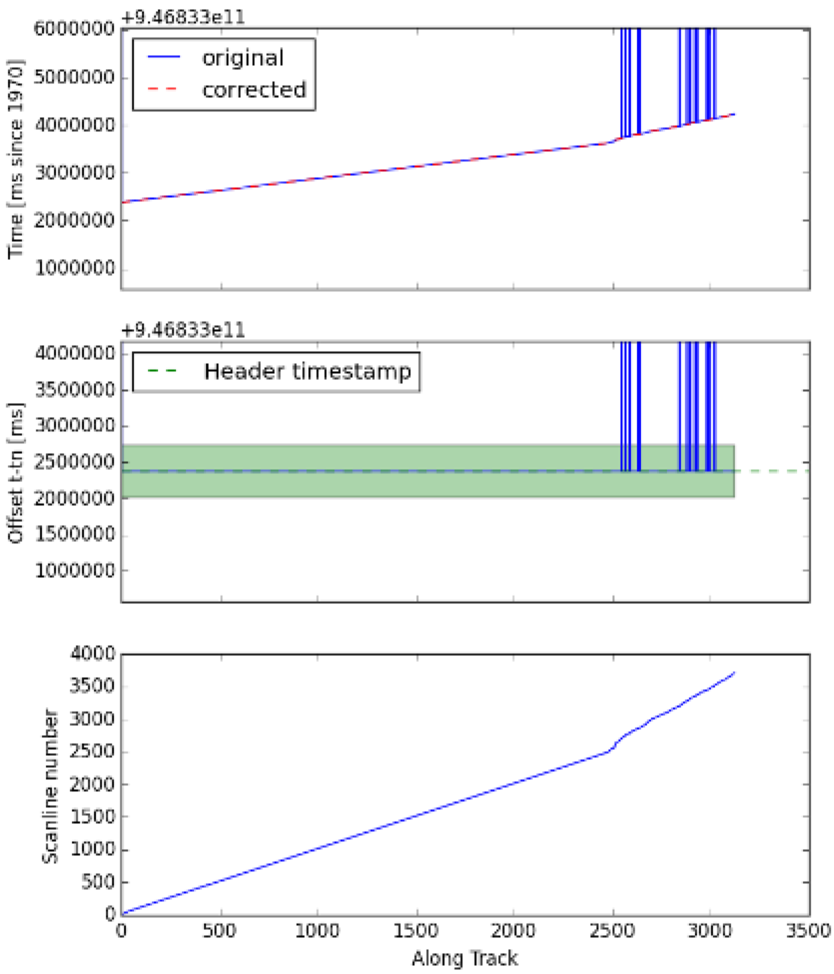


Figure 5-1 *Example of erroneous timestamps and the applied correction. Offset t0 is obtained by computing the median of all values within the green area around the header timestamp*

5.2 Temporary Scan Motor Issue

The quality analysis of the AVHRR GAC Level 2 and Level 3 products revealed several issues regarding the usability of some orbits (see Section 2.6). One of these problems affecting NOAA-14, NOAA-15, and NOAA-16 data is caused (most probably) by a temporary scan motor (TSM) issue (see examples shown in Figure 2-21 and Figure 2-22). Those orbits contain a significant amount of corrupt data and hence, have been excluded from the AVHRR GAC FCDR.

Only 0,529 % of all orbits are blacklisted due to the reason of 'temporary_scan_motor_issue'. This very small number might be suggestive of simply neglecting such orbits. However, they appear within a rather short time period leading to gaps in the time series or downgraded geophysical parameters used for time series generation.

Figure 2-21 and Figure 2-22 demonstrate that a good portion of the TSM affected orbits could actually be used for climatological applications if the corrupt measurements are masked out. Luckily the problem with the scan motor occurs in every channel for the same satellite pixels and the areas have a very noisy structure. Therefore, a correction method using simple statistics can be applied. The following two standard deviations using a 3x3 grid box can be used for the identification of deteriorated regions:

- $\boldsymbol{\sigma_{12}}$ is the standard deviation of the absolute difference between channel 1 (0.67 µm) and channel 2 (0.87 µm) and
- $\boldsymbol{\sigma_{45}}$ is the standard deviation of the relative difference between channel 4 (11 µm) and channel 5 (12 µm).

Based on a threshold approach a pixel is set to fill value (-32001) if σ_{12} is larger than 0.02 AND σ_{45} is larger than 2.00 (see Figure 5-2). Very low threshold values are necessary because the unhealthy pixels exhibit "reasonable" values. However, the combination of both, σ_{12} and σ_{45}, ensures that healthy pixels are not masked out by mistake. This correction method leads to the recovery of 2957 orbits.

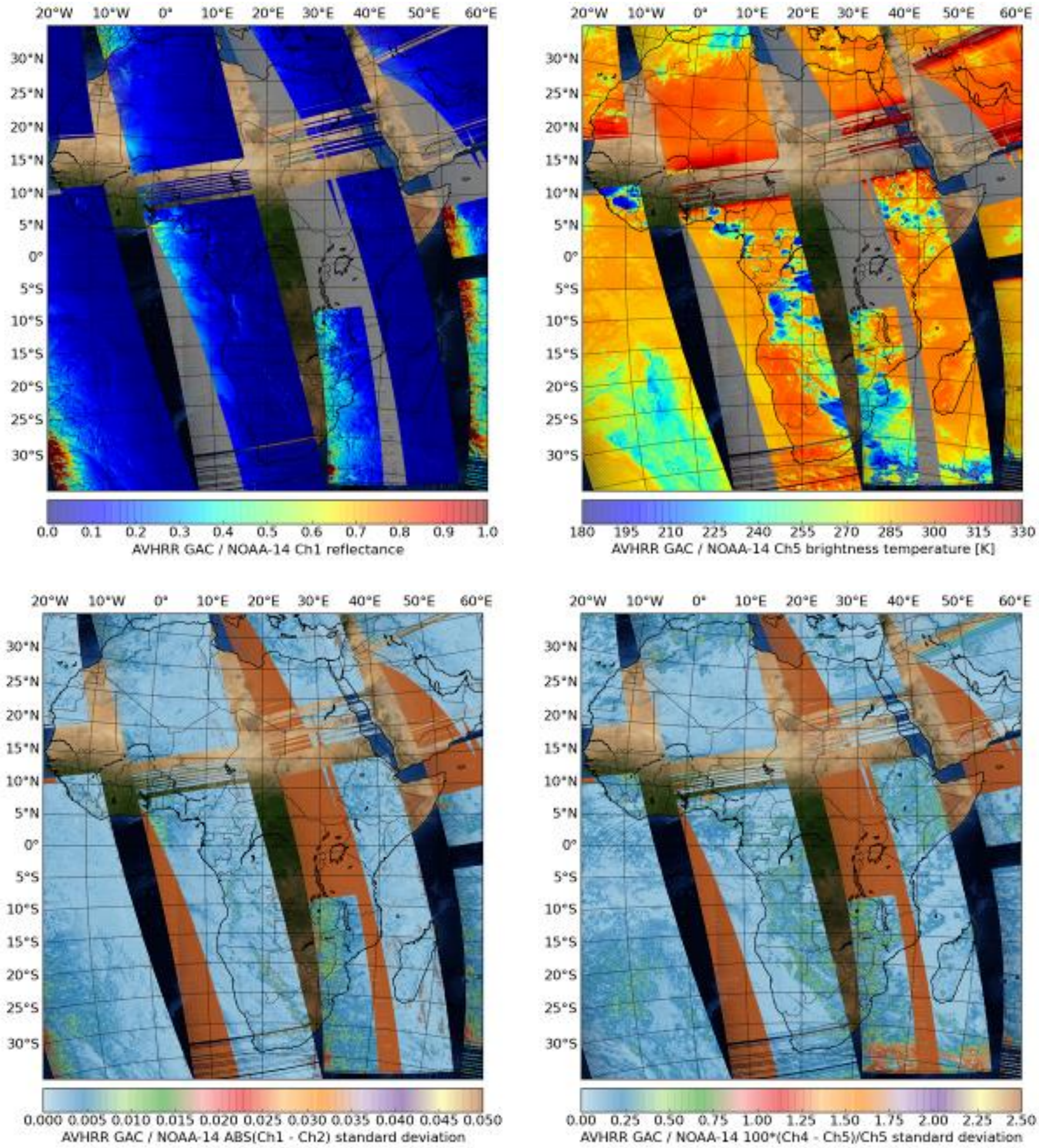


Figure 5-2 *Upper left and right maps show the temporary scan motor (TSM) issue corrected AVHRR GAC / NOAA-14 channel 1 reflectance and 5 brightness temperature measurements over Africa on 2001-10-20 (uncorrected version shown in Figure 2-22). The lower left and right maps demonstrate the standard deviation of the absolute difference between channel 1 and 2, and the standard deviation of the relative difference between channel 4 and 5. These channel differences can be used to identify the noisy areas, so that corrupt satellite pixels can be set to fill value (see dark grey in upper maps). The standard deviation is calculated using a grid box size of 3 satellite pixels.*

6. Appendix

Table 6-1 *AVHRR version 1, 2, and 3 instrument embarked on NOAA and MetOp platforms.*

AVHRR revision	Platform	Spacecraft Letter	Launch Date	Operational Start Date	Operational End Date	Operational Status (active / deactivated)	Active Channels	ECT [descending / ascending]	Sunsynch. Orbit	International Designator	NORAD Catalog Number	GAC Processing Start Date	GAC Processing End Date
1	NOAA-5	TN	1978-10-13	1978-10-19	1980-01-30	1981-02-27	4	15:00 asc	PM	1978-096A	11060	1978-11-05	1980-01-30
	NOAA-6	A	1979-06-27	1979-07-17	1986-07-09	1987-03-31	4	07:30 desc	AM	1979-057A	11416	1980-01-01	1982-08-03
	NOAA-8	E	1983-03-28	1983-06-20	1985-10-17	1985-12-29	4	07:30 desc	AM	1983-022A	13923	1983-05-04	1985-10-14
	NOAA-10	G	1986-09-17	1986-11-17	1991-09-16	2001-08-30	4	07:30 desc	AM	1986-073A	16969	1986-11-17	1991-09-16
2	NOAA-7	C	1981-06-23	1981-08-24	1985-02-01	1986-06-07	5	14:30 asc	PM	1981-059A	12553	1981-08-24	1985-02-01
	NOAA-9	F	1984-12-12	1985-02-25	1988-11-07	1998-02-13	5	14:30 asc	PM	1984-123A	15427	1985-02-25	1988-11-07
	NOAA-11	H	1988-09-24	1988-11-08	1994-10-16	2004-06-16	5	14:10 asc	PM	1988-089A	19531	1988-11-08	1994-10-16
	NOAA-12	D	1991-05-14	1991-09-17	1998-12-14	2007-08-10	5	05:10 desc	AM	1991-032A	21263	1991-09-16	1998-12-14
	NOAA-14	J	1994-12-30	1995-04-10	2002-10-07	2007-05-23	5	09:30 desc	PM	1994-089A	23455	1995-01-20	2002-10-07
3	NOAA-15	K	1998-05-13	1998-12-15	2015-12-31	AM Secondary	5	16:51 asc	AM	1998-030A	25338	1998-10-26	2015-12-31
	NOAA-16	L	2000-09-21	2001-03-20	2011-12-31	2014-06-09	5	09:01 desc	PM	2000-055A	26536	2001-01-01	2011-12-31
	NOAA-17	M	2002-06-24	2002-10-15	2011-12-31	2013-04-10	6	07:03 desc	AM	2002-032A	27453	2002-06-25	2011-12-31
	NOAA-18	N	2005-05-20	2005-08-30	2015-12-31	PM Secondary	5	15:23 asc	PM	2005-018A	28654	2005-05-20	2015-12-31
	NOAA-19	N-PRIME	2009-02-06	2009-06-02	2015-12-31	PM Primary	5	13:39 asc	PM	2009-005A	33591	2009-02-06	2015-12-31
	MetOp-1	B	2012-09-17	2013-04-24	2015-12-31	AM Primary	5	09:30 desc	AM	2012-049A	38771	2013-01-01	2015-12-31
	MetOp-2	A	2006-10-19	2007-05-21	2015-12-31	AM Backup	6	09:30 desc	AM	2006-044A	29499	2007-06-28	2015-12-31

Table 6-2 *Number of AVHRR GAC level 1 orbits per satellite, which have been blacklisted due to a specific reason.*

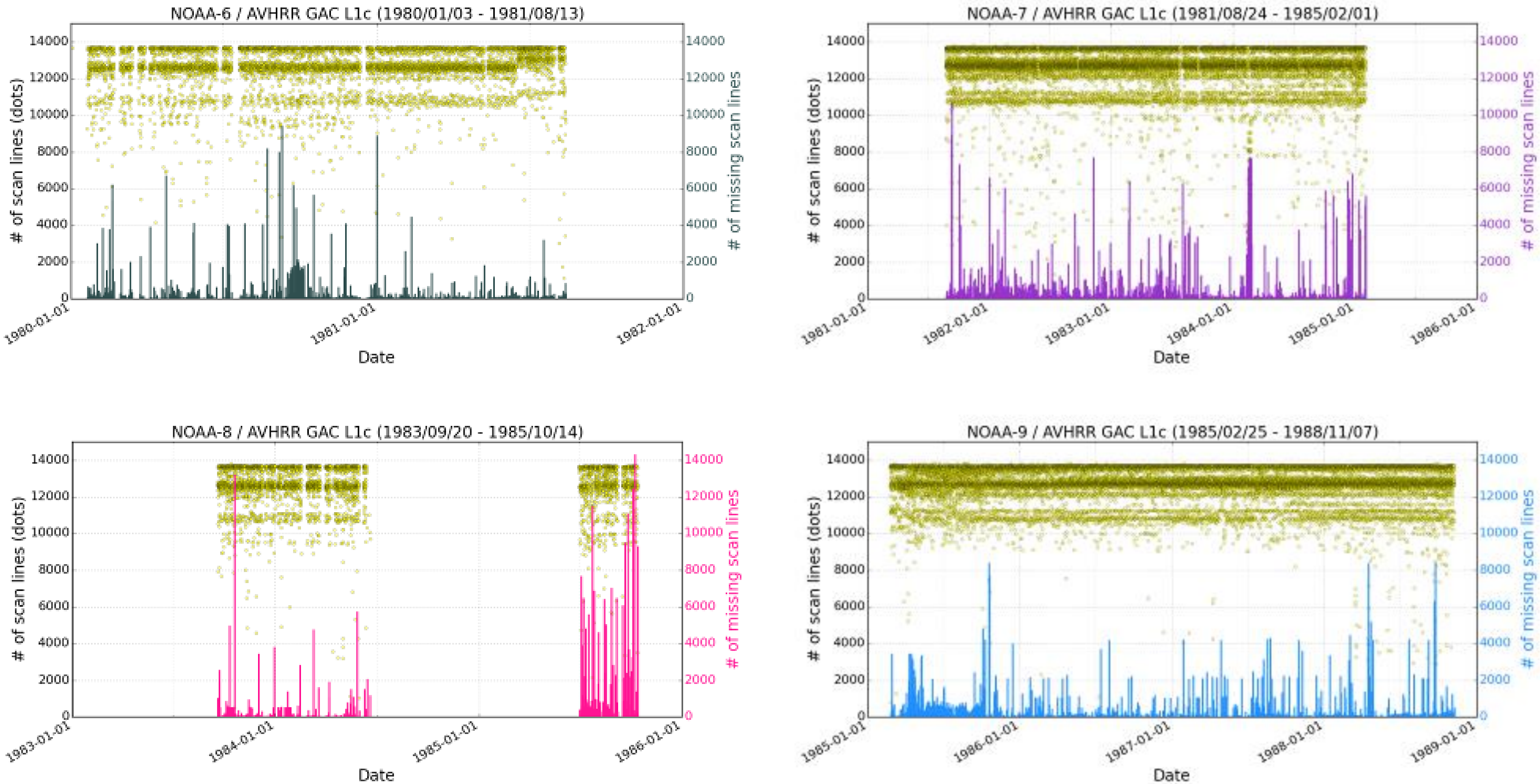
	Blacklisting reason	TIROSN	NOAA6	NOAA7	NOAA8	NOAA9	NOAA10	NOAA11	NOAA12	NOAA14	NOAA15	NOAA16	NOAA17	NOAA18	NOAA19	METOPA	METOPB	TOTAL
PRE-PROC	old	1	0	2	0	1	1	16	10	3	9	1	1	6	0	18	0	**69**
	too_small	199	56	156	53	20	484	580	687	623	416	135	163	156	121	1036	291	**5176**
	too_long	0	1	1	0	13	3	4	3	4	0	0	0	0	0	6	0	**35**
	ground_station_duplicate	1	0	3	0	0	0	0	1	5	0	0	5	12	32	25	9	**93**
	redundant	23	42	133	38	241	528	1050	1560	771	564	265	302	417	124	118	201	**6377**
PROC	orbit_length_too_long	0	3	9	3	14	5	8	24	47	776	682	361	7	8	3	0	**1950**
	negative_orbit_length	0	1	1	0	0	0	1	0	9	3	2	0	2	0	2	0	**21**
POST-PROC	wrong_l1c_timestamp	0	0	2	0	0	0	1	2	19	2	0	0	0	0	0	0	**26**
	no_valid_l1c_data	0	0	191	75	0	367	26	95	223	1928	0	239	217	215	5	24	**3605**
	bad_l1c_quality	0	90	0	117	0	0	0	0	0	0	0	9512	0	0	0	0	**9719**
	along_track_too_long	0	0	1	0	0	0	0	0	0	4	3	0	3	1	0	0	**12**
	pygac_indexerror	0	0	3	0	5	0	0	0	0	0	0	0	0	0	0	0	**8**
	ch3a_zero_reflectance	0	0	0	0	0	0	0	0	0	646	0	0	618	804	0	0	**2068**
	temporary_scan_motor_issue	0	0	0	0	0	0	0	0	1914	592	451	0	0	0	0	0	**2957**
	TOTAL	**224**	**193**	**502**	**286**	**294**	**1388**	**1686**	**2382**	**3618**	**4940**	**1539**	**10583**	**1438**	**1305**	**1213**	**525**	**32116**

Table 6-3 *This table summarizes the number of AVHRR GAC level 1b and l1c orbits along with the latest processing date.*

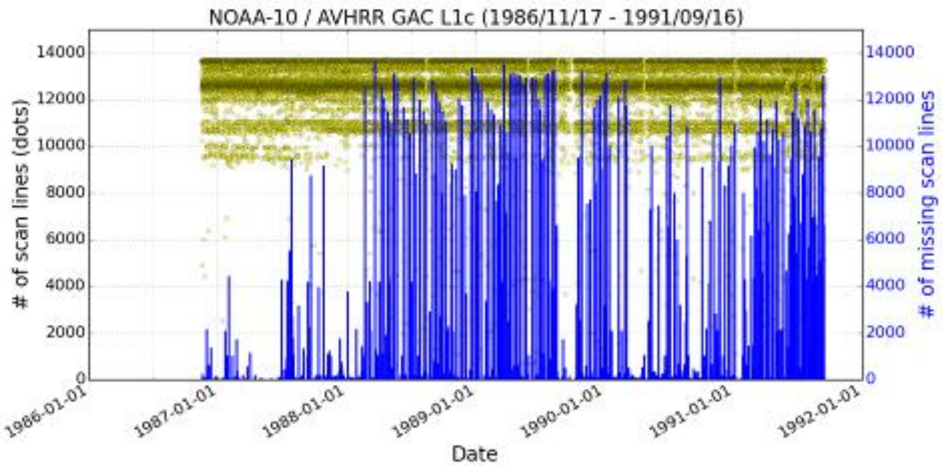
Platform	L1b orbits	whitelisted L1b orbits incl. pre blacklisting	whitelisted L1b orbits incl. proc & post blacklisting	whitelisted L1c orbits	blacklisted orbits	PyGAC failed on whitelisted L1b orbit during processing	Latest processing date
TIROSN	5457	5233	5233	0	224	5233	not yet processed
NOAA6	7007	6908	6814	6800	193	14	2015-11
NOAA7	16869	16574	16367	16271	502	96	2015-10
NOAA8	4242	4151	3956	3900	286	56	2015-11
NOAA9	18731	18456	18437	18291	294	146	2015-10
NOAA10	25400	24384	24012	23833	1388	179	2015-11
NOAA11	31802	30152	30116	29972	1686	144	2015-05
NOAA12	39846	37585	37464	37398	2382	66	2015-05
NOAA14	41148	39742	37530	37428	3618	102	2015-05
NOAA15	90001	89012	85061	84237	4940	824	2016-02
NOAA16	56970	56569	55431	55377	1539	54	2015-06
NOAA17	49896	49425	39313	39300	10583	13	2015-06
NOAA18	55161	54570	53723	53704	1438	19	2016-02
NOAA19	35774	35497	34469	34450	1305	19	2016-02
METOPA	53046	51843	51833	51548	1213	285	2016-02
METOPB	27908	27407	27383	27085	525	298	2016-02
TOTAL	**559258**	**547508**	**527142**	**519594**	**32116**	**7548**	
TOTAL [%]	**100,00%**	**97,90%**	**94,26%**	**92,91%**	**5,74%**	**1,35%**	

Table 6-4 *AVHRR visible calibration coefficients (S_0, S_1, S_2) provided by A. Heidinger.*

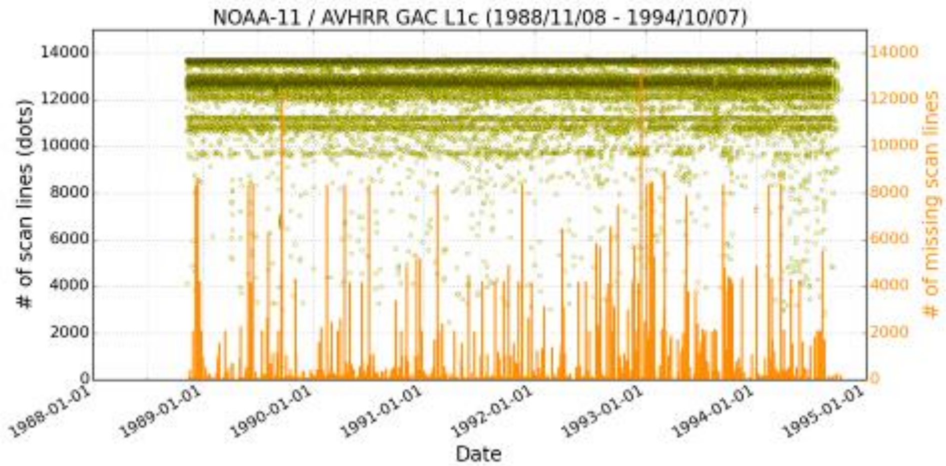
Satellite	Channel 1: S_0 | S_1 | S_2			Channel 2: S_0 | S_1 | S_2			Channel 3a: S_0 | S_1 | S_2		
noaa-6	0.087	22.164	-0.935	-0.059	-395.058	117.103	0.000	0.000	0.000
noaa-7	0.114	5.492	-0.585	0.126	6.801	-0.761	0.000	0.000	0.000
noaa-8	0.056	249.052	-103.086	0.130	-11.614	47.920	0.000	0.000	0.000
noaa-9	0.108	6.657	-0.082	0.120	5.340	-0.473	0.000	0.000	0.000
noaa-10	0.103	12.183	-2.010	0.134	3.679	-0.270	0.000	0.000	0.000
noaa-11	0.112	-0.876	0.270	0.118	-0.238	0.225	0.000	0.000	0.000
noaa-12	0.129	1.593	-0.130	0.151	2.958	-0.255	0.000	0.000	0.000
noaa-14	0.118	5.493	-0.549	0.150	2.349	-0.044	0.000	0.000	0.000
noaa-15	0.123	-0.633	0.031	0.139	0.248	-0.010	0.000	0.000	0.000
noaa-16	0.112	0.427	-0.003	0.117	2.392	-0.269	0.120	-12.115	4.462
noaa-17	0.117	0.199	0.093	0.132	3.635	-0.322	0.124	2.040	-0.123
noaa-18	0.111	2.058	-0.121	0.124	2.655	-0.155	0.000	0.000	0.00
noaa-19	0.108	2.247	-0.317	0.121	1.693	-0.132	0.000	0.000	0.000
metop-2	0.112	0.906	-0.024	0.133	0.814	0.025	0.126	2.020	-0.115
metop-1	0.119	-7.442	2.828	0.116	10.222	-3.271	0.141	-27.970	10.925



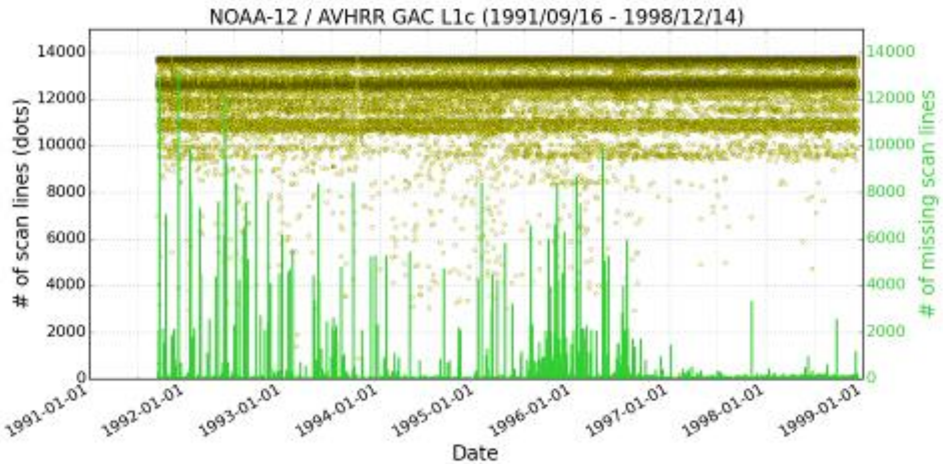
NOAA-6 / AVHRR GAC L1c (1980/01/03 - 1981/08/13)
NOAA-7 / AVHRR GAC L1c (1981/08/24 - 1985/02/01)
NOAA-8 / AVHRR GAC L1c (1983/09/20 - 1985/10/14)
NOAA-9 / AVHRR GAC L1c (1985/02/25 - 1988/11/07)
# of scan lines (dots)
# of missing scan lines
Date



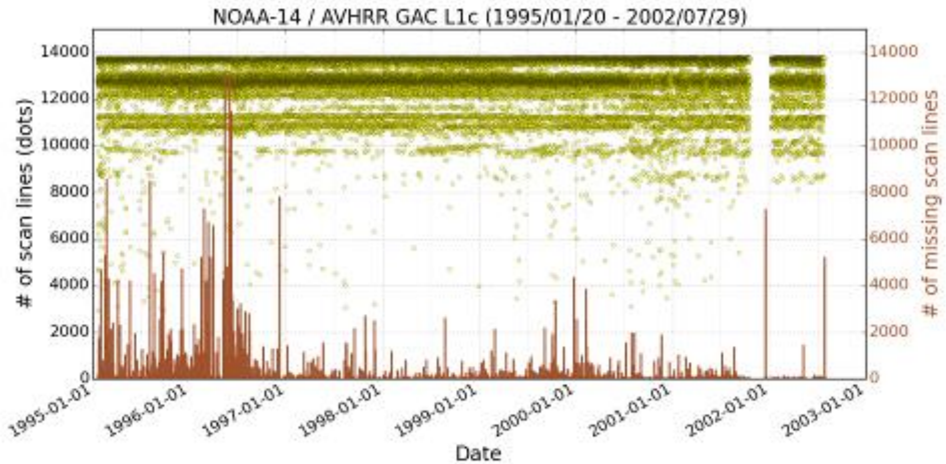
NOAA-10 / AVHRR GAC L1c (1986/11/17 - 1991/09/16)
# of scan lines (dots)
# of missing scan lines
Date



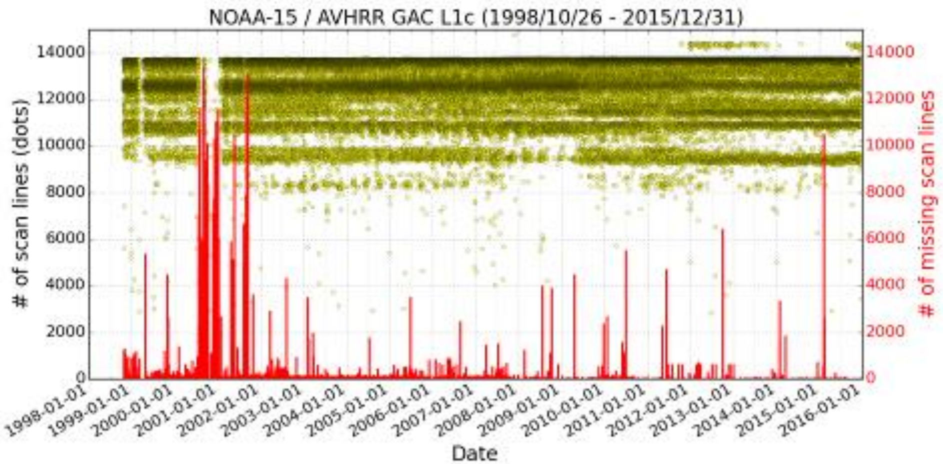
NOAA-11 / AVHRR GAC L1c (1988/11/08 - 1994/10/07)
# of scan lines (dots)
# of missing scan lines
Date



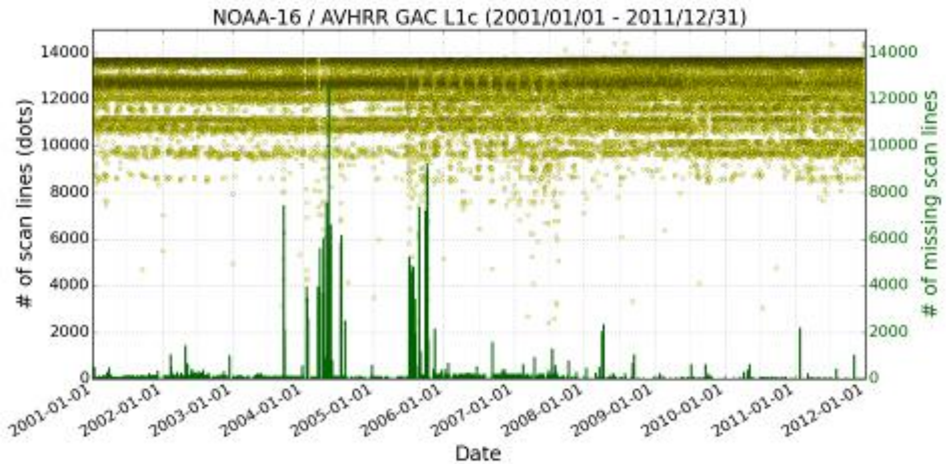
NOAA-12 / AVHRR GAC L1c (1991/09/16 - 1998/12/14)
# of scan lines (dots)
# of missing scan lines
Date



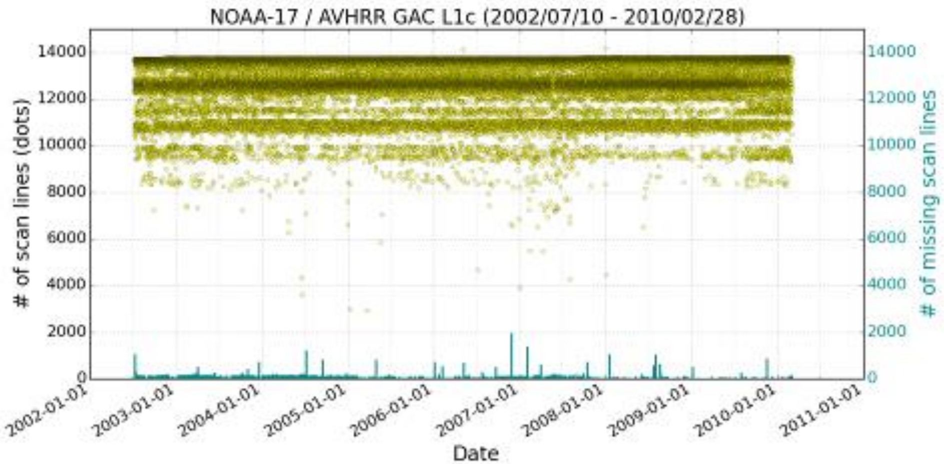
NOAA-14 / AVHRR GAC L1c (1995/01/20 - 2002/07/29)
# of scan lines (dots)
# of missing scan lines
Date



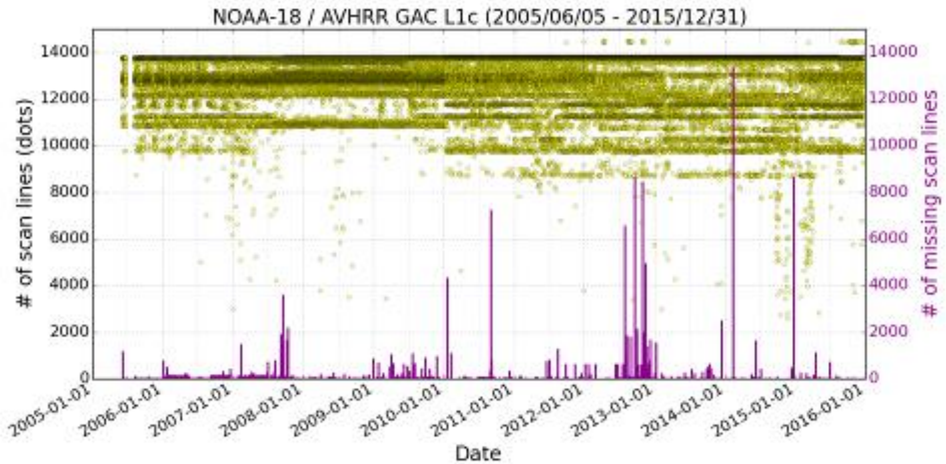
NOAA-15 / AVHRR GAC L1c (1998/10/26 - 2015/12/31)
# of scan lines (dots)
# of missing scan lines
Date



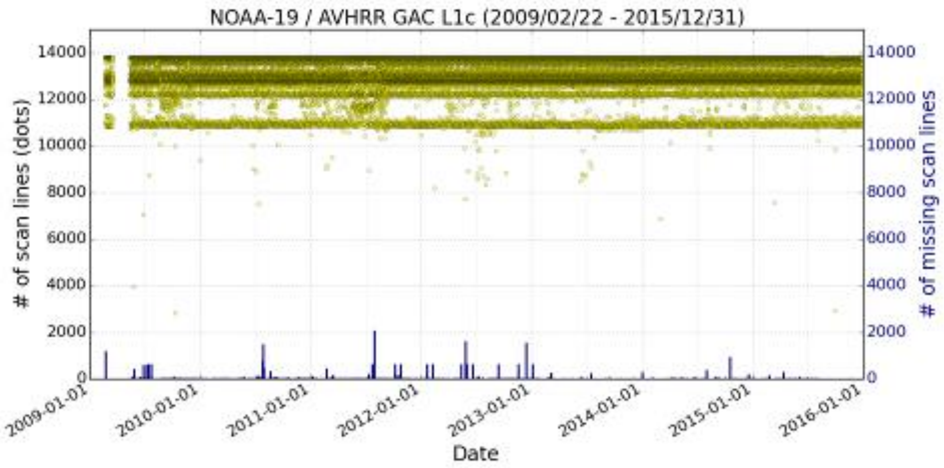
NOAA-16 / AVHRR GAC L1c (2001/01/01 - 2011/12/31)
# of scan lines (dots)
# of missing scan lines
Date

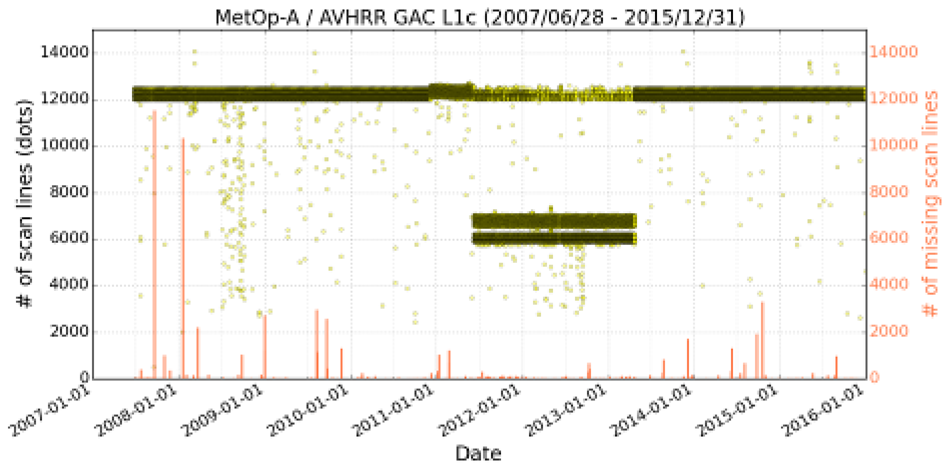


NOAA-17 / AVHRR GAC L1c (2002/07/10 - 2010/02/28)
# of scan lines (dots)
# of missing scan lines
Date



NOAA-18 / AVHRR GAC L1c (2005/06/05 - 2015/12/31)
# of scan lines (dots)
# of missing scan lines
Date





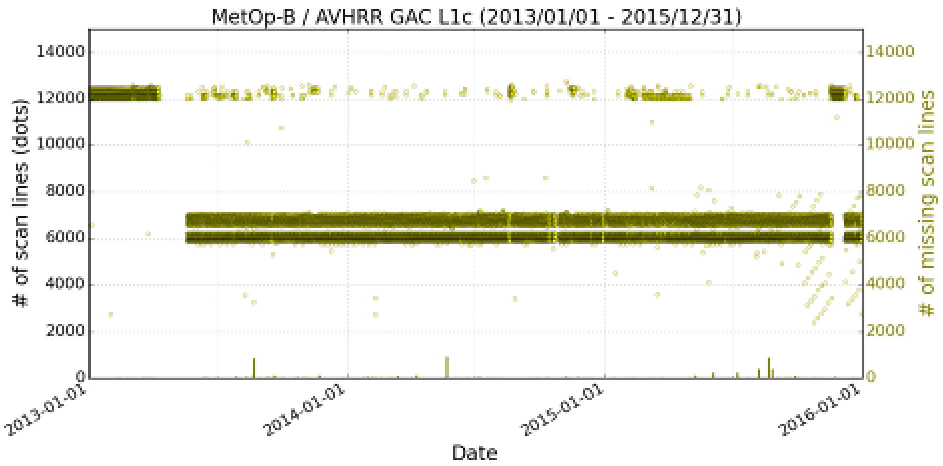


Figure 6-1 *Number of missing scan lines for each satellite as function of time. On the left y-axis the number of total scan lines (along_track) per orbit are plotted (yellow dots), while on the right y-axis the number of really missing scan lines per orbit are plotted (vertical lines). In the case of MetOp satellites one can see that the orbit length of about 12000 scan lines are cut in half for certain periods, i.e. separate orbit files for day and night observations. Moreover it is obvious that the number of missing scan lines is a pronounced feature of the AVHRR revision 1 and 2 sensors, while it is less dramatic for AVHRR/3 although about 15 scan lines are frequently missing at the beginning of the orbits (which is not visible in these plots due to the chosen y-axis).*