An effective Satellite Remote Sensing tool combining hardware and software solutions

Francesco Gugliuzza[1,⋆,(✉)], Alessandro Bruno[2,1,⋆,⋆⋆], Edoardo Ardizzone[1], and Roberto Pirrone[1]

[1] Dipartimento dell'Innovazione Industriale e Digitale (DIID),
Università degli Studi di Palermo,
Viale delle Scienze Ed. 6 - 90128 Palermo (PA), Italy
{francesco.gugliuzza,alessandro.bruno15,edoardo.ardizzone,roberto.pirrone}@unipa.it
http://www.diid.unipa.it/cvip

[2] INAF-IASF Palermo Istituto Nazionale di Astrofisica - Istituto di Astrofisica Spaziale e Fisica Cosmica,
Via Ugo La Malfa 153 - 90146 Palermo (PA), Italy
bruno@iasf.inaf.it
http://www.iasf-palermo.inaf.it

Abstract. In this paper we propose a new effective remote sensing tool combining hardware and software solutions as an extension of our previous work. In greater detail the tool consists of a low cost receiver subsystem for public weather satellites and a signal and image processing module for several tasks such as signal and image enhancement, image reconstruction and cloud detection. Our solution allows to manage data from satellites effectively with low cost components and portable software solutions. We aim at sampling and processing of the modulated signal entirely in software enabled by Software Defined Radios (SDR) and CPU computational speed overcoming hardware limitation such as high receiver noise and low ADC resolution. Since we want to extend our previous method to demodulate signals coming from various meteorological satellites, we propose a new high frequency receiving system designed to receive and demodulate signals transmitted at 1.7 GHz. The signals coming from satellites are demodulated, synchronized and enhanced by using low level image processing techniques, then cloud detection is performed by using the well known K-means clustering algorithm. The hardware and software architecture extensions make our solution able to receive and demodulate high frequency and bandwidth meteorological satellite signals, such as those transmitted by NOAA POES, NOAA GOES, EUMETSAT Metop, Meteor-M and FengYun.

[⋆] Francesco Gugliuzza and Alessandro Bruno contributed equally to this work.

[⋆⋆] The contribution of Alessandro Bruno falls within the activities of the current project titled "I telescopi Cherenkov per lo sviluppo tecnologico e culturale della Sicilia" at INAF-IASF Palermo, under the scientific supervision of Researcher Dr. Anna Anzalone.

Keywords: Remote sensing, satellite communication, signal processing, cloud detection

1 Introduction

Remote sensing techniques allow us to measure information about an object without touching it. In the last decades remote sensing has been used in various applications such as Earth observation, weather and storm predictive analysis, atmospheric monitoring, climate change, human-environment interactions. Sensors on airborne and satellites have been recording signals from space for many years, giving rise to a huge amount of data. Some data are processed on-board but others are treated and post-processed in ground stations. Signal and image processing are widely applied on data coming from satellite to extract meaningful information for the aforementioned tasks. Satellites and ground stations communicate with each others by using several transceivers and techniques; for instance communication is the largest sector of satellite services. Since the cold war a lot of scientific progress has been made both in navigation and in signal communication and processing. We are interested in communicating with satellites equipped with Advanced Very High Resolution Radiometer (AVHRR). The AVHRR is a cross-track scanning system allowing the acquisition of signals by using five spectral bands having a resolution of 1.1 km and a frequency of two Earth scans per day. NOAA weather satellites broadcast an APT (Automatic Picture Transmission) and a HRPT (High Resolution Picture Transmission) signal containing a live weather image of the area overflown by the satellite. The scanning lines are oriented perpendicular to the motion of the sensor platform. By using a system that includes a rotating mirror each line is scanned from one side of the sensor to the other. As the platform moves forward over the Earth, successive scans build up a two-dimensional image of the Earth's surface. As said before, AVHRR scanning system acquires signal by using five spectral bands and, for instance, the visible light image and one of the infrared images are combined in a row vector; the combination is performed using the Automatic Picture Transmission system (APT).

Despite many scientific progresses over last few years, many scientific issues still affect NOAA satellites: information calibration, physical layer, synchronization, data detection, channel coding. Several scientific experiments are ongoing to overcome those issues, and we focused our efforts to propose a new effective satellite remote sensing tool for signal processing. In greater detail the tool consists of a low cost receiver subsystem for public weather satellites and a signal and image processing module for several tasks such as signal and image enhancement, image reconstruction and cloud detection. Our solution allows to manage data from satellites effectively with low cost components and lightweight computations. As we will show in the experimental results section the overall performances are very promising in terms of signal quality and image reconstruction. The rest of the paper is organized as it follows: in section 2 we describe the state of the art methods for satellite signals modulation and imagery, in section 3 we show in greater detail our proposed platform for the reception of satellite signals, in section 4 we show the experimental results for the reconstructed images and the section 5 ends the paper with conclusions and future works.

In this paper we propose the extension of our previous work [1] as an effective tool for satellite data elaboration: the signals coming from satellites are demodulated, synchronized and enhanced by using low level image processing techniques, then cloud detection is performed by using the K-means clustering algorithm [2]. Also, we propose a new hardware and software architecture able to receive and demodulate high frequency and bandwidth meteorological satellite signals, such as those transmitted by NOAA POES, NOAA GOES, EUMETSAT Metop, Meteor-M and FengYun.

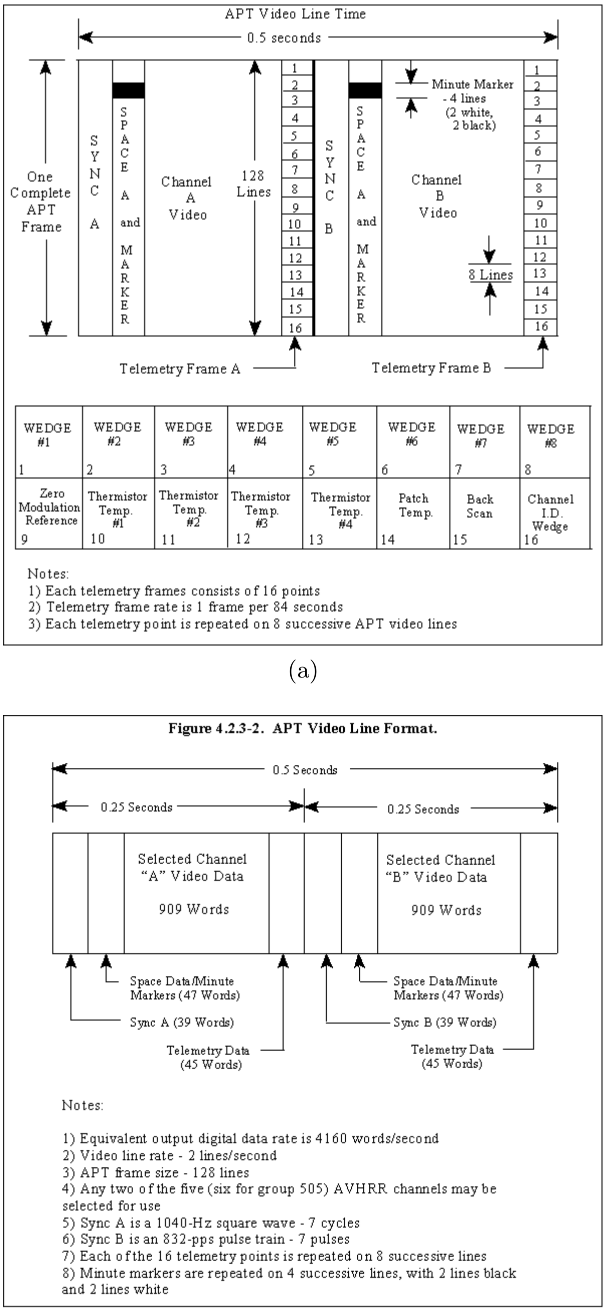
2 State of the art

In this section we give an overview of the widely adopted techniques for the demodulation of signals coming from satellites. Since we perform cloud detection on remote sensing imagery, we give a brief description of techniques of the state of the art for cloud detection.

2.1 Satellite communications and signal reception

In this section it is our interest to give the reader a brief description of some aspects related to our research with respect to the satellite communications and signal reception systems. In the last decades satellites have played a critical role interconnecting mankind through complex antennas and receiving systems. Scientific literature shows many technical solutions for antenna elements, sensors, array types, hyperspectral cameras. Depending on the satellite, we can find a low bandwidth communication system (VHF/UHF) or a high bandwidth communication system (more recent solutions have been dealing with communications by using arrays of high bandwidth microwave transponders with the purpose to offer multimedia services [3]). Scientific community has paid particular attention to reconfigurable antennas and ground receiving station, focusing the efforts towards user demands [4]. In this respect, several studies have been conducted to assess such aspects as signal propagation, tracking control, higher frequencies, arrays and reflectarrays, mechanical aspects, frequency selective surfaces [4].

Modern communication systems allow satellites to receive and transmit simultaneously thousands of signals: radiometers and hyperspectral cameras have been widely used to analyze Earth's visible and infrared radiation from the spacecraft, Synthetic Aperture Radars (SAR) have been used to map surface features and texture, even through dense cloud cover. Due to the aforementioned reasons, we are able to analyze soil/vegetation moisture (to detect anomalies and risks) [5] or produce 3D models of remote ares of the Earth (up to 1 meter of spatial resolution). Noise and distortions in raw sensor data are usually linked to the sensor nonlinearities. As a general rule they are usually corrected by using classification or regression methods [6].



WEDGE #1 1	WEDGE #2 2	WEDGE #3 3	WEDGE #4 4	WEDGE #5 5	WEDGE #6 6	WEDGE #7 7	WEDGE #8 8
Zero Modulation Reference 9	Thermistor Temp. #1 10	Thermistor Temp. #2 11	Thermistor Temp. #3 12	Thermistor Temp. #4 13	Patch Temp. 14	Back Scan 15	Channel I.D. Wedge 16

Notes:
1) Each telemetry frames consists of 16 points
2) Telemetry frame rate is 1 frame per 84 seconds
3) Each telemetry point is repeated on 8 successive APT video lines

(a)

Figure 4.2.3-2. APT Video Line Format.

Notes:

1) Equivalent output digital data rate is 4160 words/second
2) Video line rate - 2 lines/second
3) APT frame size - 128 lines
4) Any two of the five (six for group 505) AVHRR channels may be selected for use
5) Sync A is a 1040-Hz square wave - 7 cycles
6) Sync B is an 832-pps pulse train - 7 pulses
7) Each of the 16 telemetry points is repeated on 8 successive lines
8) Minute markers are repeated on 4 successive lines, with 2 lines black and 2 lines white

(b)

Fig. 1. (a) APT frame and (b) row format.

The APT format was developed around 1960 by U.S. National Aeronautics and Spatial Administration (NASA) as a communication system enabling weather satellites to take pictures over wide areas and transmit them to ground stations on Earth [7] [8]. From that time on, APT has been slightly modified and many stations still keep using analog radio equipment designed in the 1980s or the 1990s. The APT signal is formed by amplitude modulating a 2400 Hz carrier with each line of pixel data obtained by merging two images and calibration data, and then frequency modulating the result; the ease of building a basic receiver composed of a FM demodulator followed by an AM demodulator and an analog-to-digital converter has sparked the interest of researchers, scientists and radio operators. Several aspects such as Doppler effects, free space attenuation (varying with the distance from the satellite) make the reception and demodulation challenging tasks for signals coming from non-geostationary satellites. Furthermore, due to satellite's variable rotation relative to Earth's axis, linearly polarized antennas become unsuitable. For this reason circularly polarized antennas are used to fix the problem, assuring reliable and effective demodulation and synchronization, as well as a good quality output image. Other than demodulation and frequency tracking, image synchronization and calibration are usually performed by means of software solutions rather than hardware systems [9] because they are non-real time tasks that require some CPU power; it is also important to say that it would be somewhat challenging to implement those tasks in hardware (at least for non-experts). Sensor data can then be further corrected to compensate small sensor alignment errors, orbit uncertainty and on-board clock offset, by comparing expected and measured emissivity over land and sea and binning results into a high-resolution grid [10]. Over the last years several researchers have dealt with satellite remote sensing tools addressing the issue of affordability of hardware components and software solutions, to accomplish the tasks mentioned in the Introduction section of the paper. The authors of [11] designed a land station receiving through the combination of a software-defined radio (SDR), adaptive antennas on VHF band and FTP servers for the remote analysis of the obtained images. A satellite receiver was designed by Mahmood et al. [12] with the help of Realtek Software Defined Radio (RTL-SDR), Quadrifilar Helicoidal Antenna (QHA) and Trifilar Balun. QHA and Balun were designed and constructed under a precise and controlled environment. The signals were fed to SDR Sharp (generic SDR software) and were decoded to receive the latest weather image using WXtoImg. In [13] the authors proposed an experimental kit where a radio signal is received into the computer via the digital TV tuner that operated with the custom driver called RTL-SDR, and then the received signal is processed with a Python signal processing script. In [14] a low cost alternative to USRP (Universal Software Radio Peripheral) was proposed using RTL-SDR (Realtek Software Defined Radio) which is only used for reception. Furthermore a mixer circuit maps the baseband signal to the band that can be received by RTL-SDR.

2.2 Cloud Detection

Cloud properties such as cloud top height, temperature, infrared radiance have big impact on atmospheric monitoring results (forecasting of meteorological phenomena) [15]. Because of the aforementioned reasons cloud detection in remote sensing imagery is a very critical step. It is not straightforward to extract cloudiness mask from a given image from satellite because of several issues with surface properties such as reflectance and emissivity. Since many satellite missions are involved into atmospheric monitoring, achieving good results in terms of detection accuracy is quite important for improving the quality of successive steps of image analysis. Depending on the satellite product we find several features to analyze to detect cloud from an image. In this section we give a brief overview of the state of the art methods focused on cloud detection in satellite imagery. Heidinger et al. in [16] performed a naive Bayesian approach for cloud detection. This kind of methodology has been adopted for cloud detection on NOAA AVHRR data. In [17] the authors used the region growing method for dust cloud segmentation. The authors of [18], in order to accomplish the task of cloud detection, proposed an automatic supervised approach based on the scene-learning scheme. In greater detail, they simulate a kind of cubic structural data by considering different image features such as color, statistical information, texture, and structure. In [19] the authors of the document described the methods used to perform cloud detection in MODIS imagery. Lin et al. [20] proposed a radiometric normalization step to perform cloud detection in optical satellite images using invariant pixels. In [21] the authors examined two probabilistic methods for cloud masking of images from NOAA satellites, obtained with the Advanced Very High Resolution Radiometer. Simpson and Gobat in [22] used AVHRR Split-and-Merge Clustering (ASMC) for cloud detection to overcome the problem of spatially and temporally varying land surface reflectance and emissivity. Split-and-merge clustering allows to segment the scene in its natural groupings and label them as cloud, cloud-free land, uncertain. Gonzàlez et al. in [23] performed cloud classification by using watershed image segmentation, this method has been tested on images from MSG-SEVIRI (Meteosat Second Generation-Spinning Enhanced Visible and Infra-red Imager). The idea behind the method is to segment multispectral images using order-invariant watershed algorithms computed by a multi-dimensional morphological operator. In [24] after a super-pixel segmentation step, the authors accomplish the cloud detection task through bag-of-words and object classification methods. Several approaches have been adopted to perform cloud detection on remote sensing imagery, Foga et al. [25] compared many state of the art methods with respect to Landsat imagery. Bai et al. [26] performed cloud detection with machine learning and multi-feature fusion based on a comparative analysis of typical features such as spectral and texture [27]. A supervised approach using a neural network is proposed in [28] for detecting clouds over the ocean using AVHRR data. Other than the aforementioned methods there are deep learning based techniques such as [29], in which the authors used Convolutional Neural Networks (CNNs) consisting of four convolutional layers and two fully-connected layers, which can mine the deep features of clouds.

3 Proposed method

Our method focuses on the possibility of directly sampling and processing of the modulated signal entirely in software enabled by recent breakthroughs on Software Defined Radios (SDR) and CPU computational speed. Our objective was achieving good results with low cost SDR hardware like RTL-SDR [14] (a repurposed DVB-T USB dongle) or LimeSDR: in particular, we had to overcome hardware limitations present in the RTL-SDR receiver: high noise figure and low ADC resolution. We also dealt with the inherent drawbacks caused by frequent tuner saturations. We developed an integrated hardware and software system able to perform the following steps: satellite pass prediction, time scheduling, signal demodulation, image cropping and filtering, and implemented two image processing algorithms to perform cloud detection.

3.1 APT signal

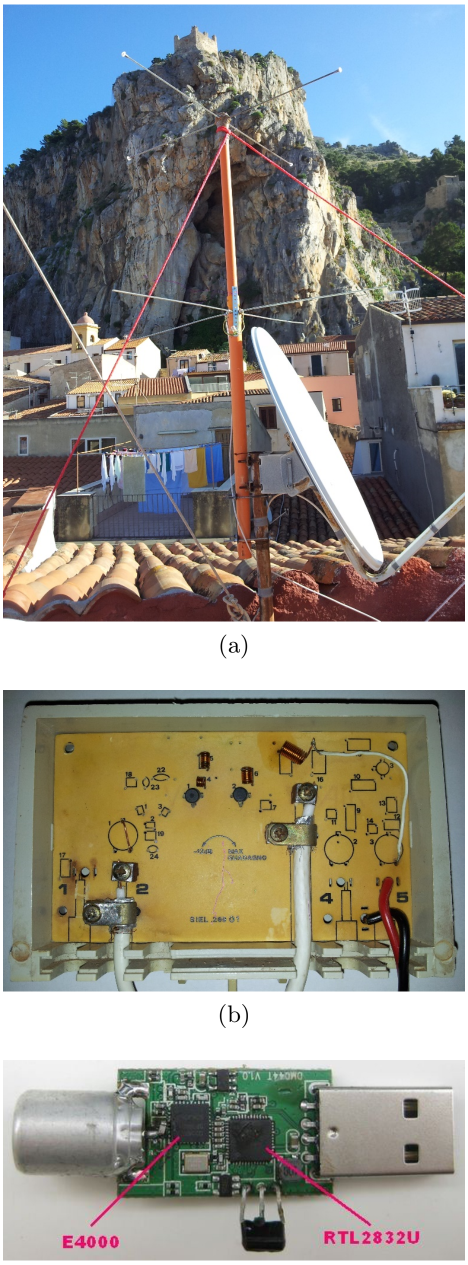
The APT was introduced in 1960s and it shows its age in some respects: it is a mixed modulation signal (AM+FM) carrying a completely analog payload. The standard APT format consists of around 2080 pixels row, which is divided in two sub-rows of 909 pixels each belonging to two different sub-images (A and B) and then padded with synchronization and diagnostic information words. Images A and B during daytime are recorded in the visible range and in the infrared range of the electromagnetic spectrum respectively. At night-time, image A is replaced with one recorded at an infrared wavelength different from that of B. Rows can be grouped logically in frames of 128 lines each: a complete frame contains image calibration data and dynamic range references (wedges) [8]. The payload is used to amplitude modulate a 2400 Hz carrier according to (1) (as in [1])

$$s(t) = [1 + \mathrm{m}(t)]\ A\ \cos(2\pi f_c t) \tag{1}$$

where $\mathrm{m}(t)$ is the modulating signal, A the carrier's amplitude and f_c the carrier's frequency. The amplitude modulated signal is then frequency modulated with a frequency deviation of about 18-20 kHz and transmitted in the 137 MHz band at about 5 W EIRP (36.99 dBm) and 4160 sym/s symbol rate. Right hand circularly polarized (RHCP) antennas transmit the signal, so particular care had to be taken when choosing which antenna type to use in the receiving station.

3.2 Hardware

NOAA weather satellites require a circularly polarized antenna: the "turnstile" crossed-dipoles design was chosen because of its simplicity and the good performance provided when placed over a ground plane. Ease of construction, durability and suitability of common materials (which can be bought in hardware



(a)

(b)

(c)

Fig. 2. (a) The turnstile antenna built for 137 MHz reception, (b) the modified TV amplifier, (c) a RTL2832U-based DVB-T dongle with an Elonics E4000 tuner.

store) compensate the drawbacks of the system: the medium gain offered (6 dBi) and many nulls present in such a design [30]. In particular, the antenna shown here has been built exclusively off PVC pipes, threaded bars as dipole elements, nuts and bolts and 75 Ω generic coaxial cable. The antenna is directly connected to a repurposed old TV amplifier (SIEL .269 01), modified to behave like a wide-bandwidth amplifier. The input variable attenuator and high-pass filter have been removed, and the RF output has been separated from DC power path, allowing power without a bias tee. The amplifier uses two active components:

- BFR90A - 16 dB gain, 1.8 dB NF
- BFR91A - 14 dB gain, 1.6 dB NF

The results are 30 dB gain and $\sim$ 1.83 dB NF (calculated using Friis' formula).

Ignoring noise introduced by amplifier's passive components, and considering a room temperature of 290 K and 40 kHz bandwidth, the noise output of the amplifier can be calculated with formula (2) (as in [1]).

$$N_{in} + g + NF \tag{2}$$

N_{in} is the input noise (-127.95 dBm), g the gain (30 dB) and NF the total noise figure (1.83 dB). The result is a noise output of -96.12 dBm. Input noise has been calculated using formula (3) (as in [1]).

$$10 \log_{10}(kTB) + 30 \tag{3}$$

k is the Boltzmann constant, T is absolute temperature and B is signal bandwidth, while +30 has been added to convert results from dBW to dBm. Considering NOAA satellites transmit at 5 W EIRP, when they are at receiving station's zenith the signal power at the receiver's input can be calculated as follows (as in [1])

$$\text{36.99 dBm - 133.34 dB + 6 dBi - 3 dB + 30 dB = -63.35 dBm} \tag{4}$$

133.34 dB is free space loss and 3 dB is the estimated loss due to cable and connectors. Output SNR in best-case conditions is then 32.77 dB, more than enough to ensure good image quality. The amplifier's output is connected to a SAW filter (Tai-Saw TA1581A) centered on 137.5 MHz to attenuate out-of-band signals and reduce the severe distortion caused by high-power FM broadcast stations and GSM signals. The TA1581 is a 50 Ω device, but an impedance matching circuit has not been installed because of the low mismatch loss (0.177 dB from antenna to filter and another 0.177 dB from filter to next RF device, which has 75 Ω impedance) compared to filter's 3.5 dB insertion loss. We added a standard ferrite bead near the receiver to shield common-mode interference caused by high-frequency equipment (PC, lab instruments, electronic ballasted lamps, etc.). The SDR hardware is widely available at a very low cost and is built around two chips: a RF tuner and an ADC/COFDM demodulator combo chip. The first used to be an Elonics E4000 (zero-IF), but has been replaced on new models with two superheterodyne alternatives, the Rafael Micro R820T and R820T2. The demodulator chip, a Realtek RTL2832U, is commonly used to demodulate the QPSK or QAM DVB-T signal into a MPEG stream and send it to a PC via an USB interface, but has a hidden passthrough mode which allows passing the ADC sample stream instead. The samples are internally filtered by a 32 coefficients FIR filter. For this study a dongle containing an E4000 tuner has been used.

3.3 High frequency high bandwidth hardware

Our new high frequency receiving system is designed to receive and demodulate the signals of various meteorological satellites (not only NOAA) transmitting at 1.7 GHz. At this band the signals can have a larger bandwidth and convey data from multiple sensors at higher resolution (e.g. High Resolution Picture Transmission - HRPT [8]).

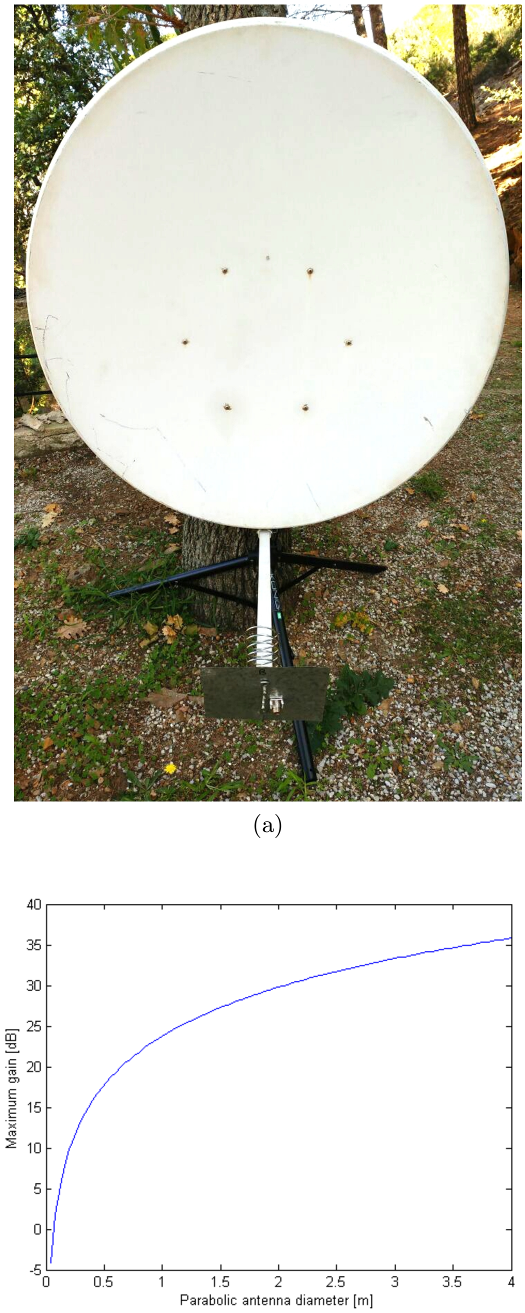
Antenna The system has been designed around an offset parabolic antenna having a diameter of 1.3 m and an efficiency factor of 0.75 (Fig. 3a). Because traditional systems have been built using prime-focus dishes having an efficiency of 0.6, our dish can actually have the same performance of a 1.5-1.6 m prime-focus dish, while being lighter and putting less stress on the antenna rotators. As it can be calculated using formula (5), where k is the efficiency factor, D the diameter and λ the signal wavelength, our antenna has a theoretical gain of 26 dB at 1.7 GHz (Fig. 3b).

$$10 \log_{10} \left[k \left(\frac{\pi D}{\lambda} \right)^2 \right] \tag{5}$$

The antenna's structure holds a helical feed designed for a dish having a focus/diameter (f/D) ratio of 0.8 (Fig. 4).

Antenna rotator In order to rotate and keep the antenna pointed towards the satellites, a Yaesu G-5500 rotator has been chosen for its durability and reliability, as it has been observed by various amateur radio operators. It can be controlled by the computer via the Yaesu GS-232A interface, which also allows to change azimuth rotation speed to ensure better satellite tracking.

Low noise amplifier In order to amplify the received signal power to an easily demodulable level, we designed and built a dedicated LNA board. It has two stages: the first stage uses an ATF-36163 pHEMT transistor having a very high gain (>20 dB) and low noise figure (<0.5 dB) at 1.7 GHz. The second stage further boosts the signal and is based on a more traditional GaAs MESFET, the ATF-13284, which offers a 16 dB gain and 0.6 dB noise figure ad 1.7 GHz. Because those are high frequency and high gain components, PCB design was very critical and we had to take various approaches to avoiding unwanted emissions,



(a)

(b)

Fig. 3. (a) The offset parabolic antenna, (b) theoretical gain vs. diameter of a parabolic antenna having an efficiency of 0.75.

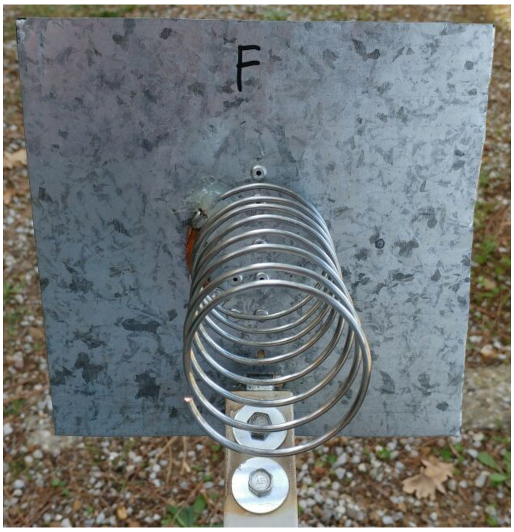


Fig. 4. The helical feed.

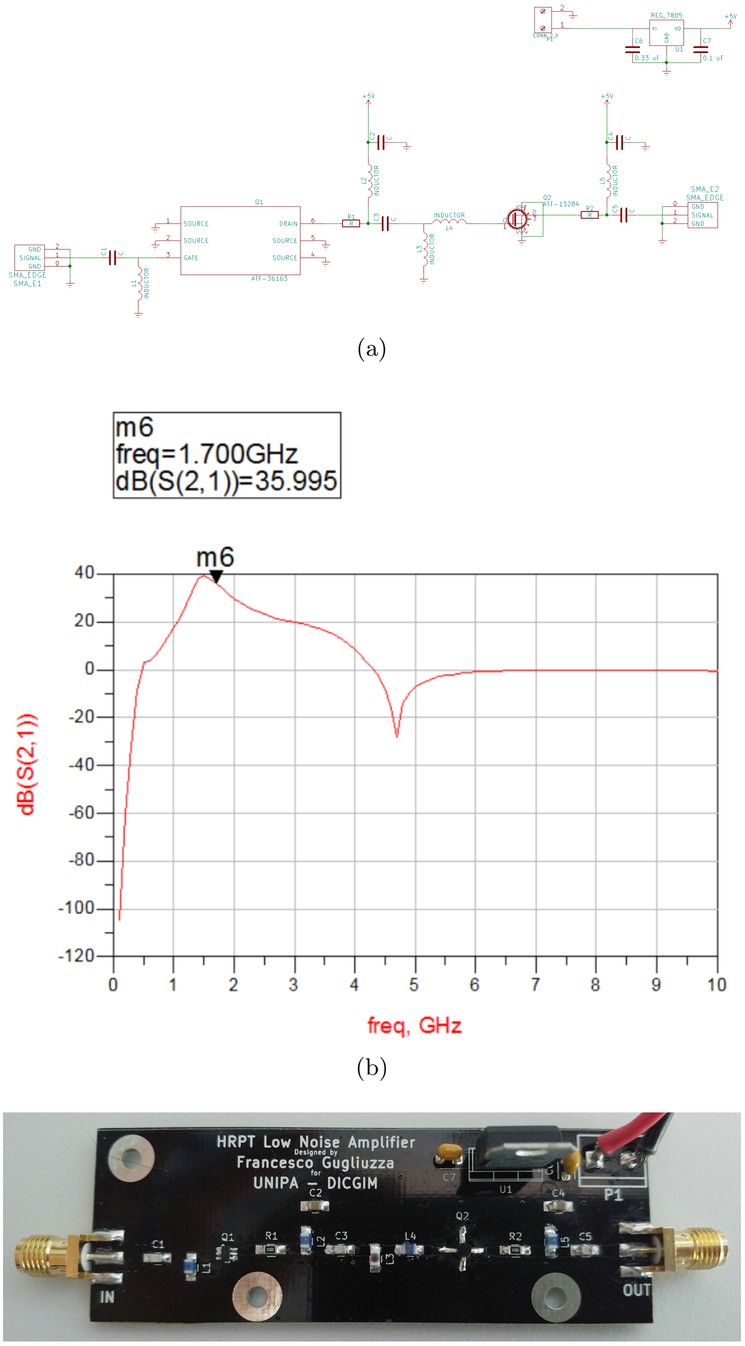
instabilities and impedance mismatch. The design has also been optimized for minimum noise figure (0.5 dB) at 1.7 GHz, while also keeping a very high gain (>35 dB) and being easily reconfigurable by simply changing discrete inductors and capacitors. The circuit schematic can be seen in Fig. 5a, while a simulation of its performance is reported in Fig. 5b. Finally, the final version of the PCB after it has been populated with components is in Fig. 5c.

SDR transceiver The SDR hardware we chose for our system is the LimeSDR, because it has an extremely wide bandwidth of 61.44 MHz, it is fully open source and it uses the latest Lime Microsystems RF chip, the LMS7002M. The LimeSDR is capable of continuous tuning from 100 kHz to 3.8 GHz and its digital design will allow to offload the computational complexity of the demodulation task from the CPU to the integrated FPGA in the future.

3.4 Software

The RTL2832 and the tuner require specific initialization and tuning commands, so we decided to rely on a third-party hardware abstraction library coded by the RTL-SDR team [31], used by the official Mathworks RTL-SDR support package [32]. The latter allows easy access to the SDR hardware directly from MATLAB and Simulink environments. As a consequence, this study has been focused on the development of DSP algorithms on Simulink, taking advantage of its model-to-code conversion functionality.

We originally decided to use the very efficient Kyle Keen's rtl_fm [33] FM demodulator, but we later found that it was very easily saturated probably due



(a)

(b)

(c)

Fig. 5. (a) The amplifier's schematic, (b) its gain vs. frequency plot, (c) the populated PCB.

to some issues in its integer math calculations. Therefore, it was necessary to implement a simple FM demodulation algorithm directly in the Simulink model.

The software we developed consists of the following parts:

- satellite pass predictor
- digital down-converter
- FM demodulator
- FIR low-pass filter
- AM demodulators
- synchronization block
- pixel dynamic range calibration block
- model's parameters generator.

Satellite pass predictor Because most weather satellites follow a polar orbit, they periodically pass on the same regions of the Earth. Each pass must be precisely predicted to schedule when to start and stop signal reception and demodulation, so John Magliacane's PREDICT [34] software has been used to predict start/end times and maximum elevations of the next satellite passes. In our system this software is called from a script and configured via command line parameters, and its output gets parsed to make it compatible with Microsoft Windows' task scheduler or, in the future, Linux/UNIX's cron scheduler.

RTL-SDR library for Simulink The Mathworks RTL-SDR library we used includes all the secondary required files, such as the precompiled low-level hardware abstraction library, sample MATLAB code and a Simulink signal source block.

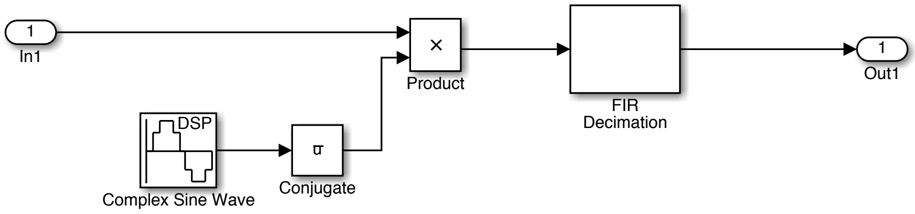


Fig. 6. Digital down-converter.

Digital down-converter In order to mitigate the effects of the DC offset present in the Zero-IF tuner, we implemented an offset tuning mechanism: the tuner is set 500 kHz lower than the real signal frequency, and circular frequency shift is performed to shift the satellite's signal back to baseband in software. This mechanism is mathematically explained in (6) (as in [1]).

$$e^{-j\Omega_0 n} = e^{\frac{-j2\pi f_0 n}{f_s}} \tag{6}$$

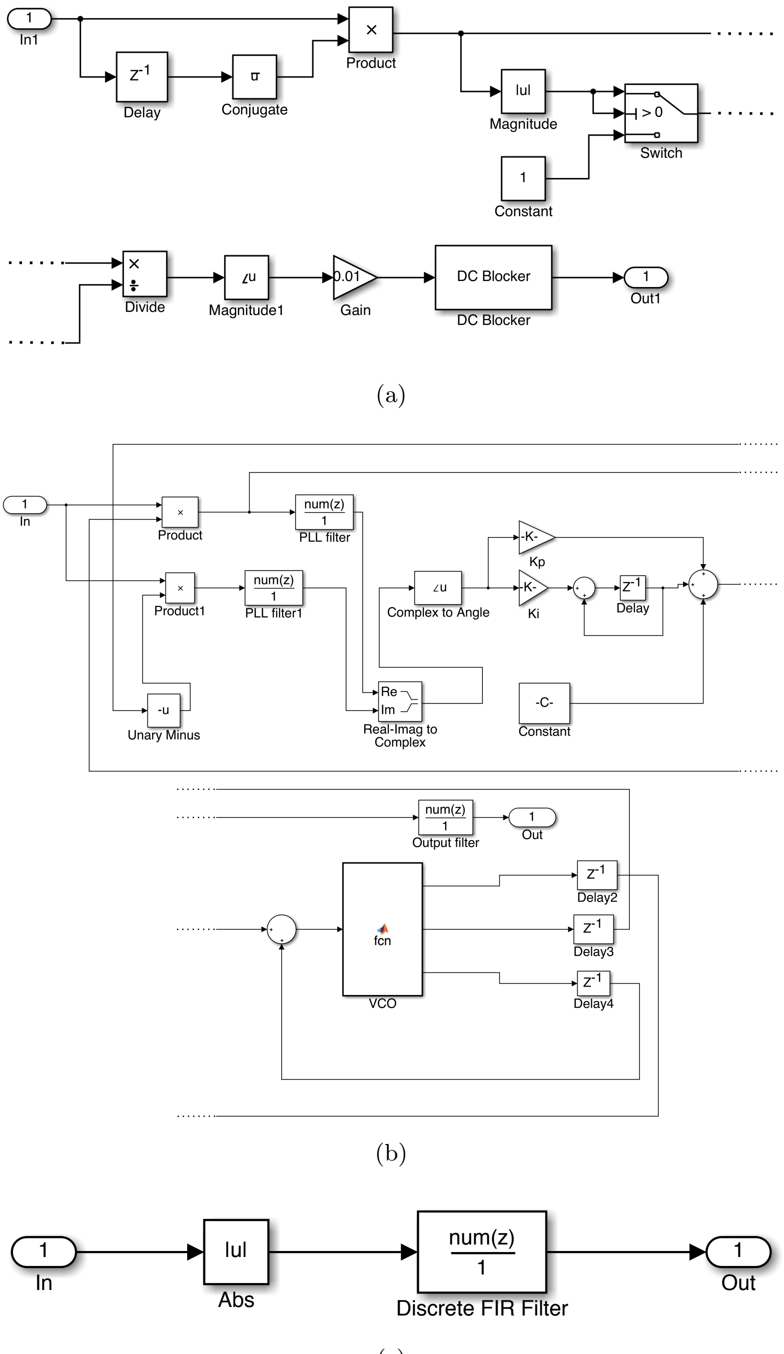
FM demodulator The original implementation of our system included a non-CPU intensive FM demodulator with automatic output amplitude rescaling (to compensate input amplitude fluctuations) [35], but it was prone to uncontrolled phase shifts in the demoduled output which heavily corrupted the 2400 Hz AM carrier. Therefore a traditional arcotangent-based FM demodulator [36] with amplitude rescaling was used in our work.

Low-pass FIR filter The high frequency noise present in the FM demodulator output is filtered with a linear phase FIR filter designed using a Kaiser window and having a cutoff frequency of 5000 Hz. The latter has been tweaked experimentally by studying receiver and FM demodulator behavior and output frequency content.

AM demodulators In our system two AM demodulators are present: the first is based on a discrete-time PLL followed by a low-pass filter with a cutoff of 1500 Hz to attenuate $2\pi f_c$ frequency components. Being a PLL-based receiver, the choice of K_i and K_p coefficients influences heavily its stability, performance and frequency tracking capability, especially when demodulating low SNR signals. We initially considered the delay of the FIR filter when tweaking the coefficients, as in [37], but the demodulator was still plagued by frequent losses of lock and low output SNR. The best results were obtained with the values reported in atpdec's [38] code.

The second demodulator is based on an envelope detector, similar to analog diode detector demodulators: the absolute value function rectifies the signal and is followed by a linear-phase FIR filter having steep frequency characteristics. This filter greatly attenuates the f_c component while leaving the modulating signal intact and having lower computational complexity than a PLL-based approach. Thanks to the steep characteristics, demodulation is very satisfactory when the input signal has high SNR; when its SNR degrades (as at the start or the end of a satellite pass) it is recommended to switch to the PLL-based demodulator, which is also capable to compensate the Doppler effect to a certain extent.

Synchronization block APT format is equipped with two synchronization pulses: horizontal synchronization (sync A) is used as a boundary between video lines and inter-channel synchronization (sync B) is used as a boundary between the two video channels [8]. Our system includes a DSP block that detects synchronization pulses and splits the input signal in image lines composed by 2080



(a)

(b)

(c)

Fig. 7. (a) FM demodulator, (b) PLL-based AM demodulator, (c) Envelope detector AM demodulator (as in [1]).

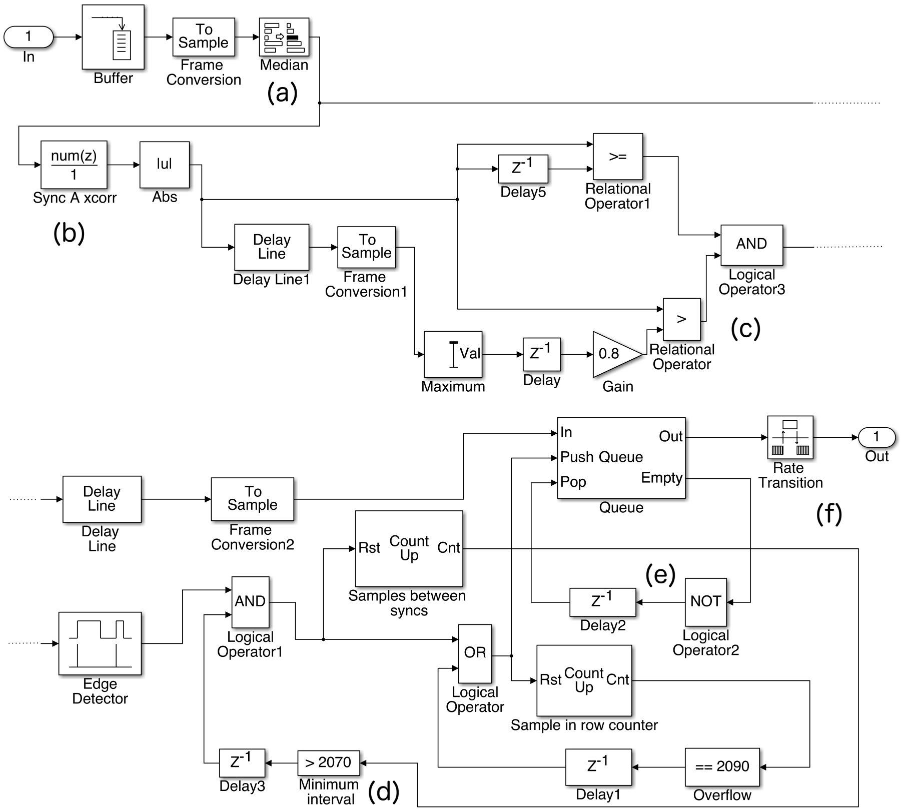


Fig. 8. Synchronization block (as in [1]).

pixels; the image is further decomposed in two sub-images later in the Image Processing code.

The synchronization block requires that the input signal is sampled at a multiple of the symbol rate (4160 symbols/s) and downsamples it by selecting the median of an array of N samples, therefore filtering the signal and rejecting impulse noise.

During the experimental stage we also tested a simple interpolation algorithm for signal sampling, but this solution introduced excessive smoothing to the image and therefore we preferred the median algorithm solution.

Part (a) of Fig. 8 down-samples the signal, while (b) correlates previous part's output with a pattern of sync A; synchronization pulses are detected comparing local correlation peaks with maximum correlation in the last 3/4 seconds (c) (time during which one and a half video lines get transmitted). If correlation at discrete instant k-1 is a local peak and is greater than 80% of maximum correlation in said interval, it is considered as a synchronization pulse and a row formed by the last 2080 samples is inserted as a new element at the end of queue (e). Sometimes, however, synchronization pulses are lost or are erroneously detected due to signal fading or noise, so a filtering technique was devised (d): it filters pulses detected less than 2070 samples after last one. If a pulse has not been detected after 2090 samples the line is inserted at the end of the queue anyway and the 2090 samples counter is reset; the 2070 samples counter is not reset not to interfere with subsequent synchronization pulses. Finally, the Rate Transition block (f) outputs two lines of 2080 pixels each per second.

Pixel dynamic range calibration block Each frame, composed of 128 rows, contains 16 "wedges": the first nine carry data about the image dynamic range (wedges 1-8 represent values from 1/8 to 8/8, while 9 is the zero reference). Our calibration block correlates a column of pixels of the wedge region with an example pattern, and then finds the exact position of each wedge. The values of wedges 8 and 9 are sampled and intensity offset and dynamic range are calculated. Offset is subtracted from pixel values and the result is divided by the detected dynamic range to normalize pixel values to the [0 1] interval: values outside this range are considered noise or saturation and clipped to 0 or 1.

This block outputs a stream of rows composed of 2080 pixels, which are later processed using Image Processing algorithms.

3.5 New software

MathWorks Simulink is not optimized for real-time processing of high bandwidth signals, so we chose a different and more appropriate framework: GNU Radio [39]. GNU Radio offers a flowgraph representation of signal processing algorithms and its blocks are written in Python or C++. Blocks can take advantage of fast and efficient low level math functions via CPU abstractions (VOLK machines [40]) or, using third-party libraries, they can directly use GPU acceleration.

Many SDR transceivers and signal processing blocks are already supported in GNU Radio.

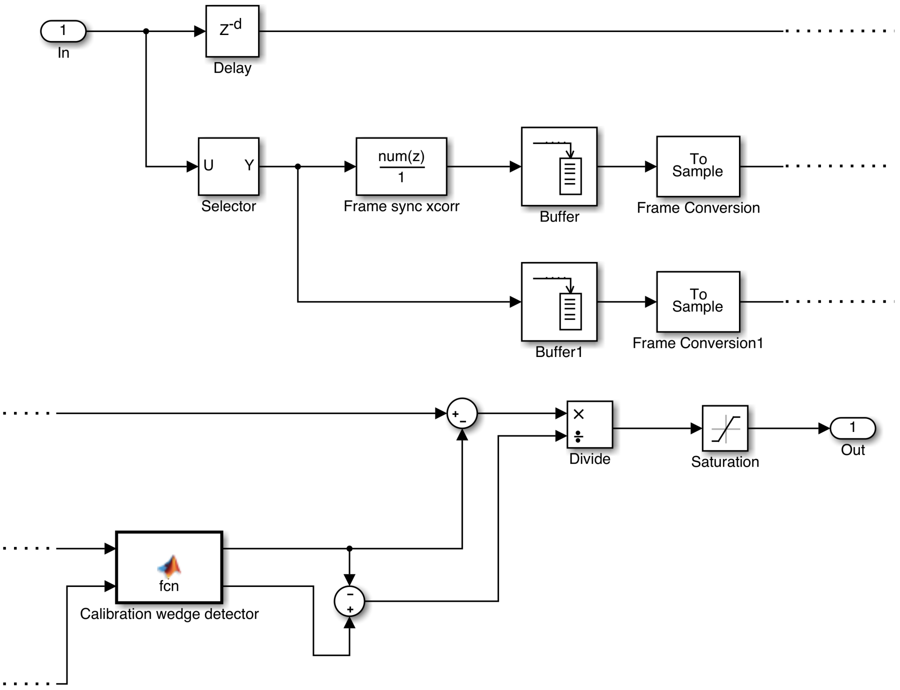
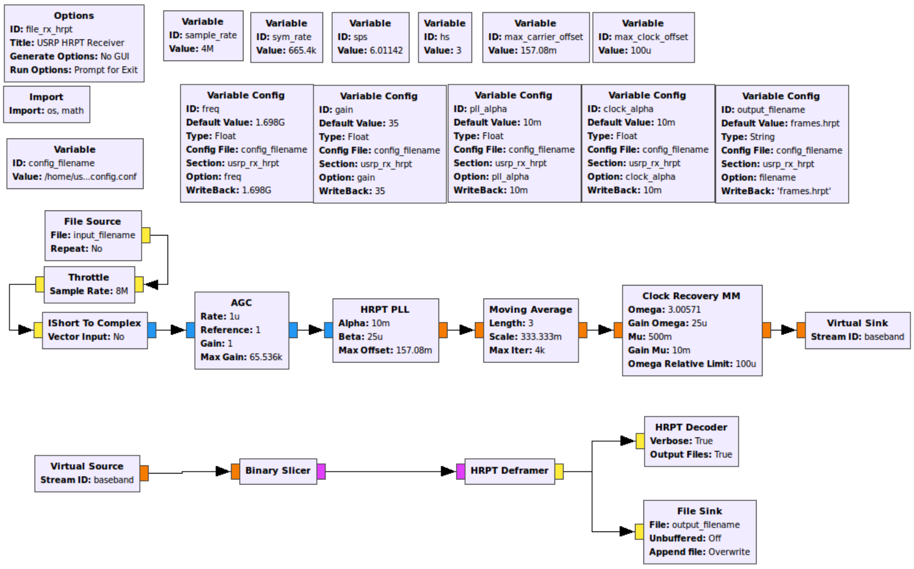


Fig. 9. Pixel dynamic range calibration block.



Options
ID: file_rx_hrpt
Title: USRP HRPT Receiver
Generate Options: No GUI
Run Options: Prompt for Exit

Variable
ID: sample_rate
Value: 4M

Variable
ID: sym_rate
Value: 665.4k

Variable
ID: sps
Value: 6.01142

Variable
ID: hs
Value: 3

Variable
ID: max_carrier_offset
Value: 157.08m

Variable
ID: max_clock_offset
Value: 100u

Import
Import: os, math

Variable
ID: config_filename
Value: /home/us...config.conf

Variable Config
ID: freq
Default Value: 1.698G
Type: Float
Config File: config_filename
Section: usrp_rx_hrpt
Option: freq
WriteBack: 1.698G

Variable Config
ID: gain
Default Value: 35
Type: Float
Config File: config_filename
Section: usrp_rx_hrpt
Option: gain
WriteBack: 35

Variable Config
ID: pll_alpha
Default Value: 10m
Type: Float
Config File: config_filename
Section: usrp_rx_hrpt
Option: pll_alpha
WriteBack: 10m

Variable Config
ID: clock_alpha
Default Value: 10m
Type: Float
Config File: config_filename
Section: usrp_rx_hrpt
Option: clock_alpha
WriteBack: 10m

Variable Config
ID: output_filename
Default Value: frames.hrpt
Type: String
Config File: config_filename
Section: usrp_rx_hrpt
Option: filename
WriteBack: 'frames.hrpt'

File Source
File: input_filename
Repeat: No

Throttle
Sample Rate: 8M

IShort To Complex
Vector Input: No

AGC
Rate: 1u
Reference: 1
Gain: 1
Max Gain: 65.536k

HRPT PLL
Alpha: 10m
Beta: 25u
Max Offset: 157.08m

Moving Average
Length: 3
Scale: 333.333m
Max Iter: 4k

Clock Recovery MM
Omega: 3.00571
Gain Omega: 25u
Mu: 500m
Gain Mu: 10m
Omega Relative Limit: 100u

Virtual Sink
Stream ID: baseband

Virtual Source
Stream ID: baseband

Binary Slicer

HRPT Deframer

HRPT Decoder
Verbose: True
Output Files: True

File Sink
File: output_filename
Unbuffered: Off
Append file: Overwrite

Fig. 10. Example flowgraph for HRPT reception in GNU Radio [41].

Flowgraphs for APT or HRPT signal decoding are already available on the Internet [42] [43] or in the official libraries [41] (Fig. 10). Our objectives include adding support for other signal types, as Color HRPT (CHRPT), Advanced HRPT (AHRPT), Meteor-M HRPT, High Rate Information Transmission (HRIT), GOES Variable (GVAR) and GOES Rebroadcast (GRB).

Some parts of the legacy software solution, like the satellite pass predictor, are general purpose and can be reused.

4 Experimental results

Our system generates a 2080x2*T image, where T is running time in seconds. Background white noise or artifacts caused by low signal SNR, fading, interferences and out-of-band intermodulation (Fig. 11) are cropped out by an algorithm using pixel variance in calibration wedges to estimate row reception quality. Only rows whose calibration wedges have an entropy below a certain threshold are kept. To further enhance image quality, various Image Processing filters have been applied to the output images and objective metrics (PSNR, RMSE, SNR_{rms} and SSIM) have been used to compare the results, which are reported in Table 1. The disk filter is shown to be achieving the best results (Fig. 12).

After image denoising we tried two classical approaches to image segmentation, in order to isolate cloud pixels both from land pixels and the rest of the

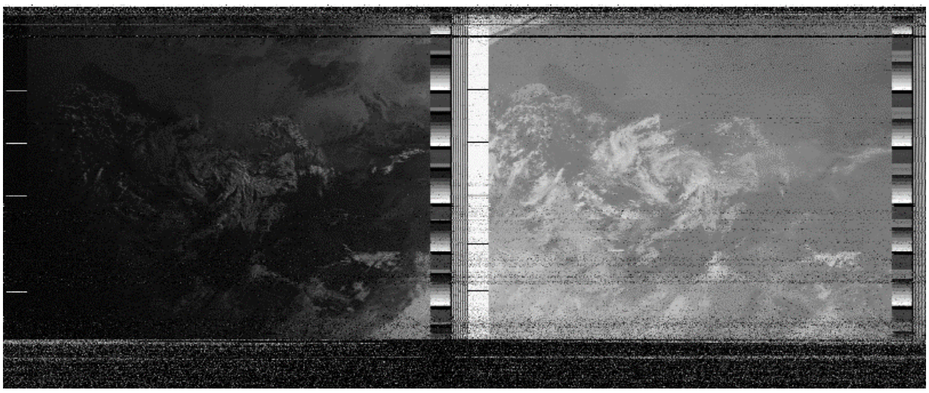


Fig. 11. AVHRR image after automatic cropping (as in [1]).

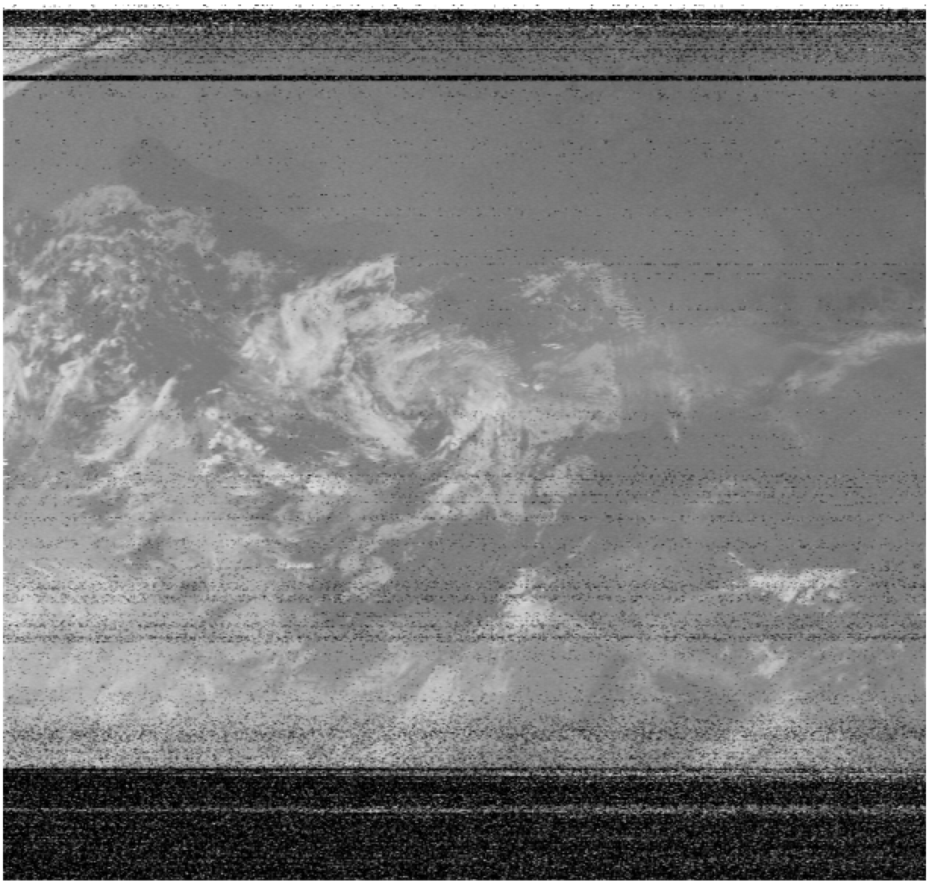
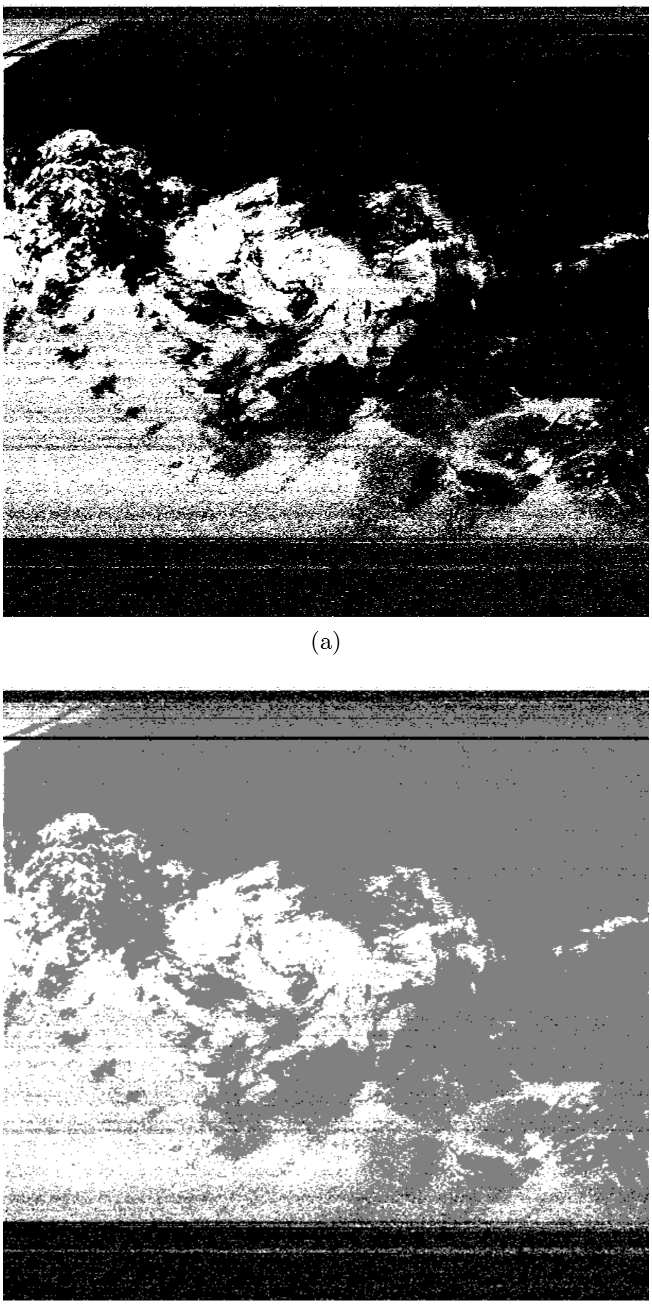


Fig. 12. Disk filtered remote sensing image from NOAA (as in [1]).



(a)

(b)

Fig. 13. (a) Cloud segmentation obtained with Otsu method (as in [1]) and (b) cloud segmentation achieved using the K-means algorithm (K = 3).

image: the first approach uses the Otsu method [44] (as in [1]), a global thresholding based on spatial clustering: the one dimensional version of the method uses only the grey level information of the pixel, otherwise the 2D version of the method uses both the grey level information of the pixel and the spatial correlation within the neighbourhood. It is widely used for its simpleness and effectiveness, and is a good solution for images coming from NOAA satellites because it is robust against noise (usually present in APT images). The results are satisfactory and are shown in Fig. 13a. The second approach is based on K-means segmentation [2] with K = 3, and its results are also satisfactory and are shown in Fig. 13b.

That being said, cloud segmentation is not our main objective: we built a new accessible platform to receive NOAA signals, we used some new signal processing techniques for the demodulation and synchronization of the information coming from satellites, and finally we adopted some Image Processing techniques to enhance noisy images and achieve better results in terms of remote sensing imagery. We also show the results of cloud segmentation achieved using the Otsu method and K-means segmentation.

Table 1. Objective metrics results for image enhancement filtering (extended version of the table reported in [1]).

Method	PSNR [dB]	RMSE	$\mathrm{SNR}_{\mathrm{rms}}$	SSIM
None (raw image)	17.788	0.129	4.253	0.557
3x3 arithmetic mean filter	17.981	0.126	4.228	0.606
3x3 geometric mean filter	13.655	0.208	2.488	0.472
3x3 harmonic mean filter	13.295	0.216	2.357	0.449
3x3 contraharmonic mean filter with Q = 1	18.023	0.126	4.499	0.625
3x3 gaussian filter	18.485	0.119	4.551	0.609
3x3 disk filter	18.590	0.118	4.570	0.634
Adaptive median filter (min: 3x3 - max: 5x5)	18.444	0.120	4.586	0.622
Adaptive filter 3x3	16.298	0.153	3.550	0.563

5 Conclusion and future works

In this paper we proposed a complete system for weather satellite data reception. It is able to predict and track satellite passes, receive and demodulate both legacy NOAA APT signals (thanks to our previous work reported in [1]) and high resolution data contained in newer signal formats transmitted by various satellites. Despite the new system being still a work in progress, its specifications already match or exceed those of commercially available solutions [45], while being very easily reconfigurable to support new signal and data formats. In our future works we want to improve our solution with modular upgrades, thus adding the following:

- Multiple feeds and LNAs for multiple frequency bands;
- Support for various meteorological signal and payload formats;
- Support for error correction codes (where available)
- An algorithm for thermal maps generation based on the analysis of multispectral infrared images;
- Noise profiling based on long time analysis of the radio frequency interferences when the interferences are located near the receiver.

We aim to develop a modular solution for the prediction of atmospheric phenomena, based on the analysis of the images retrieved from several satellites (each one transmitting at its own frequency and with its own modulation scheme) at different times of the day. As far as it concerns cloud detection we proposed a solution that is mainly based on a simple and straightforward clustering method, that is, K-means. The method performs well with K=3 but, generally speaking, the performance of a segmentation method based on clustering depends on the elements inside the image. We think that a a fixed number K=3 for K-means algorithm could not be sufficient for cloud detection whether the image contains elements such as soil, sea, cirrus, aerosols and clouds as well. In future works we want K cluster number to be adaptive with respect to the image we deal with. We could analyze the histogram of a patch to count the number of distribution modes, trying to establish a correspondence with the number of elements inside the image.

References

1. Ardizzone, E., Bruno, A., Gugliuzza, F., Pirrone, R.: A Low Cost Solution for NOAA Remote Sensing. In: Proceedings of the 7th International Conference on Sensor Networks (SENSORNETS 2018), SCITEPRESS Science and Technology Publications, Lda. All rights reserved (2018) 128–134
2. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Fifth Berkeley Symposium on Mathematical Statistics and Probability. Volume 1., Oakland, CA, USA (1967) 281–297
3. Farserotu, J., Prasad, R.: A survey of future broadband multimedia satellite systems, issues and trends. IEEE Communications Magazine **38**(6) (2000) 128–133
4. Rahmat-Samii, Y., Densmore, A.C.: Technology trends and challenges of antennas for satellite communication systems. IEEE Transactions on Antennas and Propagation **63**(4) (2015) 1191–1204
5. Al-Moustafa, T., Armitage, R.P., Danson, F.M.: Mapping fuel moisture content in upland vegetation using airborne hyperspectral imagery. Remote sensing of Environment **127** (2012) 74–83
6. Camps-Valls, G., Benediktsson, J.A., Bruzzone, L., Chanussot, J.: Introduction to the issue on advances in remote sensing image processing. IEEE Journal of Selected Topics in Signal Processing **5**(3) (2011) 365–369
7. Barnes, J.C., Smallwood, M.D.: Tiros-N series direct readout services users guide. National Earth Satellite Service, NOAA (1982)
8. Wallach, J.: User's Guide for Building and Operating Environmental Satellite Receiving Stations. National Environmental Satellite, Data, and Information Service, NOAA (1997)

9. Benabadji, N., Hassini, A., Belbachir, A.H.: Hardware and software consideration to use NOAA images. Revue Internationale des Energies Renouvelables, CDER **7**(01) (2004) 1–11
10. Moradi, I., Meng, H., Ferraro, R.R., Bilanow, S.: Correcting geolocation errors for microwave instruments aboard NOAA satellites. IEEE Transactions on Geoscience and Remote Sensing **51**(6) (2013) 3625–3637
11. Bosquez, C., Ramos, A., Noboa, L.: System for receiving NOAA meteorological satellite images using software defined radio. In: ANDESCON, 2016 IEEE, IEEE (2016) 1–4
12. Mahmood, S., Mushtaq, M.T., Jaffer, G.: Cost efficient design approach for receiving the NOAA weather satellites data. In: Aerospace Conference, 2016 IEEE, IEEE (2016) 1–6
13. Uengtrakul, B., Bunnjaweht, D.: A cost efficient software defined radio receiver for demonstrating concepts in communication and signal processing using Python and RTL-SDR. In: Digital Information and Communication Technology and its Applications (DICTAP), 2014 Fourth International Conference on, IEEE (2014) 394–399
14. Sruthi, M., Abirami, M., Manikkoth, A., Gandhiraj, R., Soman, K.: Low cost digital transceiver design for Software Defined Radio using RTL-SDR. In: Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013 International Multi-Conference on, IEEE (2013) 852–855
15. Feidas, H., Cartalis, C., Cracknell, A.: Use of Meteosat imagery to define clouds linked with floods in Greece. International Journal of Remote Sensing **21**(5) (2000) 1047–1072
16. Heidinger, A.K., Evan, A.T., Foster, M.J., Walther, A.: A naive Bayesian cloud-detection scheme derived from CALIPSO and applied within PATMOS-x. Journal of Applied Meteorology and Climatology **51**(6) (2012) 1129–1144
17. Alkhatib, M.Q., Cabrera, S.D., Gill, T.E.: Automated detection of dust clouds and sources in NOAA-AVHRR satellite imagery. In: Image Analysis and Interpretation (SSIAI), 2012 IEEE Southwest Symposium on, IEEE (2012) 97–100
18. An, Z., Shi, Z.: Scene learning for cloud detection on remote-sensing images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **8**(8) (2015) 4206–4222
19. Ackerman, S., Strabala, K., Menzel, P., Frey, R., Moeller, C., Gumley, L., Baum, B., Schaaf, C., Riggs, G.: Discriminating clear-sky from cloud with MODIS algorithm theoretical basis document (MOD35). ATBD Ref. ATBD-MOD-06, version **4** (1997) 115p MODIS Cloud Mask Team, Cooperative Institute for Meteorological Satellite Studies, University of Wisconsin.
20. Lin, C.H., Lin, B.Y., Lee, K.Y., Chen, Y.C.: Radiometric normalization and cloud detection of optical satellite images using invariant pixels. ISPRS Journal of Photogrammetry and Remote Sensing **106** (2015) 107–117
21. Karlsson, K.G., Johansson, E., Devasthale, A.: Advancing the uncertainty characterisation of cloud masking in passive satellite imagery: Probabilistic formulations for NOAA AVHRR data. Remote Sensing of Environment **158** (2015) 126–139
22. Simpson, J.J., Gobat, J.I.: Improved cloud detection for daytime AVHRR scenes over land. Remote Sensing of Environment **55**(1) (1996) 21–49
23. González, A., Pérez, J.C., Muñoz, J., Méndez, Z., Armas, M.: Watershed image segmentation and cloud classification from multispectral MSG–SEVIRI imagery. Advances in Space Research **49**(1) (2012) 135–142

24. Yuan, Y., Hu, X.: Bag-of-words and object-based classification for cloud extraction from satellite imagery. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **8**(8) (2015) 4197–4205
25. Foga, S., Scaramuzza, P.L., Guo, S., Zhu, Z., Dilley, R.D., Beckmann, T., Schmidt, G.L., Dwyer, J.L., Hughes, M.J., Laue, B.: Cloud detection algorithm comparison and validation for operational Landsat data products. Remote sensing of environment **194** (2017) 379–390
26. Bai, T., Li, D., Sun, K., Chen, Y., Li, W.: Cloud detection for high-resolution satellite imagery using machine learning and multi-feature fusion. Remote Sensing **8**(9) (2016) 715
27. Ardizzone, E., Bruno, A., Mazzola, G.: Scale detection via keypoint density maps in regular or near-regular textures. Pattern Recognition Letters **34**(16) (2013) 2071–2078
28. Yhann, S.R., Simpson, J.J.: Application of neural networks to AVHRR cloud segmentation. IEEE transactions on geoscience and remote sensing **33**(3) (1995) 590–604
29. Shi, M., Xie, F., Zi, Y., Yin, J.: Cloud detection of remote sensing images by deep learning. In: Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International, IEEE (2016) 701–704
30. Griffiths, M.: Turnstile design — DigitalHam (2014) http://www.digitalham.co.uk/design.
31. Osmocom: rtl-sdr - OsmoSDR (2012) http://sdr.osmocom.org/trac/wiki/rtl-sdr.
32. Sergienko, A.B.: Software-defined radio in MATLAB Simulink with RTL-SDR hardware. In: Computer Technologies in Physical and Engineering Applications (ICCTPEA), 2014 International Conference on, IEEE (2014) 160–161
33. Keen, K.: Rtl_fm Guide: Updates for rtl_fm overhaul (2013) http://kmkeen.com/rtl-demod-guide.
34. Magliacane, J.: PREDICT - A Satellite Tracking/Orbital Prediction Program (2001) http://www.qsl.net/kd2bd/predict.html.
35. Lyons, R.G.: Understanding Digital Signal Processing, 3/E. Pearson Education India (2004)
36. Shima, J.M.: FM demodulation using a digital radio and digital signal processing. Master's thesis, University of Florida (1995)
37. Wilson, J., Nelson, A., Farhang-Boroujeny, B.: Parameter derivation of type-2 discrete-time phase-locked loops containing feedback delays. IEEE Transactions on Circuits and Systems II: Express Briefs **56**(12) (2009) 886–890
38. Leconte, T.: ATPDEC Home Page (2003) http://atpdec.sourceforge.net.
39. The GNU Radio Foundation: GNU Radio (2001) https://www.gnuradio.org.
40. The GNU Radio Foundation: Vector-Optimized Library of Kernels (2015) http://libvolk.org.
41. The GNU Radio Foundation: NOAA POES HRPT receiver (2009) https://github.com/gnuradio/gnuradio/tree/master/gr-noaa.
42. Csete, A.: NOAA Weather Satellite Reception with GNU Radio and USRP (2010) http://oz9aec.net/radios/gnu-radio/noaa-weather-satellite-reception-with-gnu-radio-and-usrp.
43. Bülo, M.: A Simple GnuRadio HRPT Decoder (2018) https://tynet.eu/hrpt-decoder.
44. Zhang, J., Hu, J.: Image segmentation based on 2D Otsu method with histogram analysis. In: Computer Science and Software Engineering, 2008 International Conference on. Volume 6., IEEE (2008) 105–108

45. Dartcom Systems Ltd.: HRPT/AHRPT System (2018) https://www.dartcom.co.uk/products/hrpt-ahrpt-system.