

HORIZON ACQUISITION FOR ATTITUDE DETERMINATION USING IMAGE PROCESSING ALGORITHMS – RESULTS OF HORACE ON REXUS 16

Jochen Barf, Thomas Rapp, Matthias Bergmann, Sven Geiger, Arthur Scharf, and Florian Wolz

Team HORACE

*Chair of Computer Science VIII – Julius Maximilian University of Würzburg c/o Prof. Dr. Hakan Kayal
Sanderring 2, D-97070 Würzburg, Germany*

Email of corresponding author: jochen.barf@horace-rexus.de

ABSTRACT

The aim of the Horizon Acquisition Experiment (HORACE) was to prove a new concept for a two-axis horizon sensor using algorithms processing ordinary images, which is also operable at high spinning rates occurring during emergencies. The difficulty to cope with image distortions, which is avoided by conventional horizon sensors, was introduced on purpose as we envision a system being capable of using any optical data. During the flight on REXUS¹ 16, which provided a suitable platform similar to the future application scenario, a malfunction of the payload cameras caused severe degradation of the collected scientific data. Nevertheless, with the aid of simulations we could show that the concept is accurate ($\pm 0.6^\circ$), fast ($\approx 100\text{ms}/\text{frame}$) and robust enough for coarse attitude determination during emergencies and also applicable for small satellites. Besides, technical knowledge regarding the design of REXUS-experiments, including the detection of interferences between SATA and GPS, was gained.

Key words: HORACE; Horizon Acquisition Experiment; Horizon Sensor; Attitude Sensor; OpenCV; REXUS 16.

1. INTRODUCTION

The Attitude Determination and Control System (ADCS) is an essential subsystems of the satellite bus for many missions. As a result, a broad field of sensor systems for attitude determination has been developed, using different references and physical effects. From sun sensors and magnetometers to horizon sensors and star trackers, which are selected and designed regarding the mission specific requirements and thus come along with different advantages and disadvantages for specific application scenarios. Despite the developments in this field many missions still fail due to malfunctions of the ADCS or exceedances of its capabilities during

¹Rocket Experiments for University Students

unforeseen non-nominal situations, e.g. if the satellite tumbles with high rates after a collision with space debris. Hence, this paper presents another approach for attitude determination systems capable of providing at least coarse attitude information also during said non-nominal situations. This approach is an horizon sensor applying image processing techniques to optical data.

During the project life-cycle of the Horizon Acquisition Experiment (HORACE) in the scope of the REXUS/BEXUS programme² we developed a test platform and the image processing algorithms for the horizon acquisition. In order to prove or disprove the new concept under space-like conditions, it was tested on the flight of REXUS 16, which was successfully launched on May 28 2014 13:30 (UTC+2) from Esrange. The sounding rocket's unguided and uncontrolled flight to an altitude of 87km provided a suitable possibility to gather data of motions similar to those of a tumbling satellite.

In the following sections, first the general approach is outlined and justified, before the findings both of the development of HORACE and its flight are described and discussed thoroughly. This discussion regards not only the mission objectives but also lessons learned which are relevant for the design and operations of experiments on sounding rockets in general.

2. GENERAL APPROACH

Besides the mentioned capabilities for non-nominal situations, one would like the sensor system to add minimal extra weight and hardware so as to also be applicable for small satellites and as it is most likely only used during emergencies. During nominal phases other sensors commonly provide more suitable attitude information regarding the mission. At the same time it should consume only little energy as the total energy consumption is a life-limiting factor, especially during emergen-

²Rocket and Balloon Experiments for University Students

cies. It should additionally shall operate completely autonomously, not only to follow the general trend towards highly autonomous satellites but also as one has to expect not to have proper communication links during emergencies.

Taking into account these factors, among all types of attitude determination sensors an horizon acquisition sensor seems to be the best choice for several reasons. Firstly, no high accuracy is required, but unlike star trackers it can also cope with high spinning rates. Secondly, the central body of the orbit can almost always be distinguished from outer space, hence the sensor is mostly independent from the position in orbit, regarding eclipse-times (unlike sun sensors) or the central body’s magnetic field. Finally, it does not require regular calibration as inertial measurement units do.

In contrast to many existing horizon sensors which are sensitive to the infrared spectrum emitted by earth and which use dedicated infrared sensors, the presented approach is based on the processing of image data provided by ordinary cameras sensitive to the visible spectrum. This difficulty – in fact the horizon can be detected much more easily and accurately in the infrared spectrum – was introduced on purpose, as thus the software components and algorithms are strongly emphasized. This is the first step towards our vision of a generic software system which is capable of processing any optical data. The advantage of this is that many satellites already carry cameras or other optical instruments, for earth observation etc. Therefore most satellites may already possess the necessary hardware for the horizon acquisition, hence no extra weight has to be added for dedicated hardware (cf. Rapp et al., 2014, Sec. 1.1).

3. EXPERIMENT OVERVIEW

3.1. Electrical & Mechanical Setup

The flight segment of the experiment consists of five subsystems which are mainly implemented with off-the-shelf components to reduce development effort and cost: a camera, the core system, a power distribution unit, a measurement unit and the mechanical structure (cf. Fig. 1).

The CMOS-camera mvBlueCOUGAR-X102b from Matrix Vision passes the image data directly to the core system via a GigE-Vision interface. This industrial camera model was chosen as it has an adjustable exposure time between $10\mu\text{s}$ and 1s provides a resolution of $1024\text{ px} \times 768\text{ px}$ (cf. MATRIX VISION GmbH, 2012). The core system, a MIO-2260 single-board industrial computer from Advantech, is the central subsystem. It not only receives the image data from the camera, but also stores it directly via its SATAII interface to a solid state disk (Transcend SSD320) and processes it for the horizon acquisition. The results of the algorithm are stored to the solid state disk as well. The core system furthermore interprets the signals and controls the up- and downlink provided by the REXUS Service Module (RXSM) and provides an experiment-wide global timestamp. The

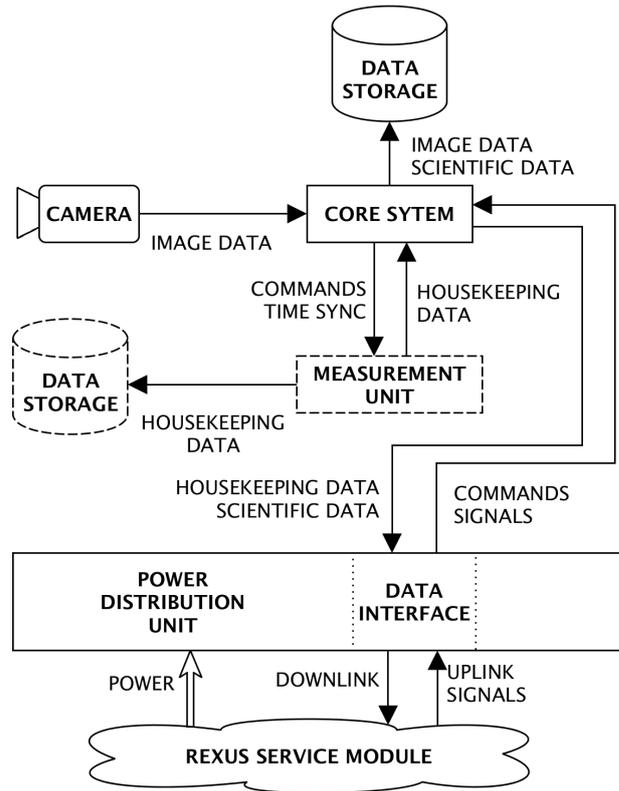


Figure 1: Block diagram of the flight segment. The measurement unit & its data storage (both dashed) are implemented by the master system only.

measurement unit is based on an Arduino Leonardo and collects housekeeping data (temperatures and currents) at various points of the experiment for monitoring and check outs. Depending on the state of the experiment, the housekeeping data is either stored on a microSD-card or passed to the core system for downlink. The power distribution unit implements the electrical interface to the RXSM both for power- and communication-lines. Hence, it conditions the 28V unregulated power provided by the RXSM with three LM2596 DC/DC-Converter modules to the voltages required by the other subsystems (2x 12V and 5V), takes care of galvanic isolation for the signal lines and converts the up- and downlink channels from RS-422 (on the RXSM side) to RS-232 (on the experiment side). To protect the sensitive electronics, they are housed in aluminum boxes, each subsystem in its own box to increase modularity and to simplify the replacement procedures in case of defective components. All those components are placed on a single bulkhead plate, which is mounted horizontally in the experiment module and is populated from both sides for easy access to all subsystems and optimum utilization of volume. The camera looks out of the experiment module through a hole in the module which is equipped with a borosilicate window to protect the components from the otherwise intruding hot airflow (cf. Wolz, 2015).

In order to collect more scientific data, two completely independent clones of the system, which are different only

in minor aspects and hence denoted as master and slave system, are placed in the same module in a symmetrical manner. Hence, when the horizon is not in the field of view of one system it is most likely to be visible for the second system. All together, the complete experiment has a total mass of 7.84kg (including module) and consumed 2.76Wh (master) plus 2.74Wh (slave) during the flight (cf. Rapp et al., 2014, Sec. 7.3.1).

3.2. Optical Design

The optical design and its constraints are made with respect to the mechanical design of the experiment (i.e. symmetrical nature), the expected flight path (i.e. expected maximum altitude of ca. 80 km) and the alleviation of the algorithmic workload (i.e. reduction of outside effects, like lens flares caused by the sun).

Therefore, the optical design should fulfill two main requirements. Firstly, at the highest altitude, at least one camera should have the horizon in its field of view. Secondly, distortions in the resulting images caused by the lenses should be avoided, and the strength of lens flares or reflexes caused by the sun should be reduced as much as possible.

To fulfill the first requirement, the horizon of the earth has to be visible, if the image plane is perpendicular to the nadir direction at a height of 80km. This results in a minimal vertical field of view of about 18° and the focal length of the lens was selected accordingly to achieve this.

Additionally, the cameras are placed adverse to each other, so that when one camera is pointing to the vast emptiness of space, the other is pointing towards earth.

The second requirement is fulfilled by selecting a lens with a minimal distortion and mounting the camera with the lens as close as possible to the protective window. Furthermore, the window frames are anodized in black and the borosilicate windows are specially coated to reduce stray light with accompanied reflexes and lens flares.

The lens, which matches those requirements and was therefore selected, is a Lensagon BT8020N with a focal length of 8mm and a maximum distortion of -1.6% (cf. LENSATION, 2014). The distance between the front of the lens and the inside surface of the windows is about 2mm. Additionally, the lens has no moving parts to prevent changes of the optical properties (e.g. replacement of the focal point) due to vibrations during launch. Furthermore, it does not contain grouped lenses as embedded air within the groups could potentially crack a lens in the vacuum environment.

3.3. Software Design

The core systems runs *Arch Linux* (Griffin, 2015), a minimal Linux desktop derivate, as its operating system. Using Linux systems both on the core system and the ground station allowed an easier and faster implementation of the

software of both as they hence could adapt the same communication framework. The usage of an operating system instead of a bare metal implementation has two further advantages. Firstly, it allowed to regain the control over the flight software of the core system in case of a software crash by watch-dog bash scripts running in the background. Secondly, it facilitated testing in early stages of the software development. The minimalistic distribution, which requires the explicit installation and configuration of every component of the operating system, was chosen to minimize the risk of software malfunction due to unexpected behavior of the operating system.

The flight software running on top of this operating system is the heart of HORACE. One of its tasks is to integrate the algorithm for the horizon acquisition, to request image data from the camera via the camera API and to provide it to the algorithm. Telemetry and telecommands are also controlled by the flight software, including signal processing of the signals provided by the RXSM. To correlate the data collected by the measurement unit to the results of the horizon acquisition, the flight software of the core system also provided an experiment wide global time stamp.

To control the overall program flow of the experiment, three software modes are implemented in the core systems as well as in the measurement unit. While the rocket is still on the ground, HORACE is in a waiting position for the lift-off signal. In this standby-mode self-checks (e.g. testing communication, camera connection, access to data storages) are performed and the experiment provides housekeeping and information about the software state via the telemetry link to check out the flight segment. Immediately after the reception of the lift-off signal the software switches to the flight-mode. It starts collecting data and running the horizon acquisition. Only some basic scientific data is downlinked due to the limited bandwidth of the radio link, while the image data and more detailed results of the algorithm are stored on the SSD. The housekeeping data collected by the measurement unit is only stored on board and no longer sent to ground as it is no longer important for the operations of the experiment due to the lack of an uplink during flight. Hence, the bandwidth is reserved for the scientific data. At T+590s (close to the point when the experiments are switched off by the RXSM) the software stops collecting and writing data in order to prevent corruption of data by switching to a shut-down mode.

Except for a few bash scripts with minor tasks running in the background, all software components are developed in C++. Aside from *OpenCV* (Willow Garage, 2014), which is a powerful library for image processing and therefore used by the algorithm for the horizon acquisition, and *QT* (QT Company, 2015), used for the implementation of the graphical user interface of the ground-station, only standard C++ libraries were used in order to prevent unexpected behavior of the software.

4. ALGORITHM

4.1. Basic Idea

The horizon sensor is a two-axis attitude determination sensor capable of measuring the roll and pitch angle of a spacecraft in relation to the central body. The HORACE algorithm obtains this by calculating the vector to the center of the 2D projection of the central body. For the calculation, ordinary images from cameras in the visible spectrum are used. The image processing algorithm requires the horizon of the central body to be in the field of view of the camera and a good contrast in the image. The basic concept is to distinguish the central body from the dark space by using the contrast. Logic is implemented to detect and eliminate disturbances.

4.2. Detailed Algorithmic Approach

The first step in the *Preprocessing* is to convert the colored image into a grayscale image of smaller size and to reject unusable images. This image is then converted by the *Threshold Filter* to a black-and-white image, also known as a binary image. This represents bright areas of the image with pixel values of 1 and dark areas with pixels values of 0, respectively. The border between a 1 area and a 0 area is called an edge. Since the contrast between the central body and space is high, the horizon is an edge in the picture. Unfortunately, it can happen, that the horizon is not the only edge in the picture. Therefore, it is necessary to find the edge that represents the horizon (*Line Detection*). Having found this edge, the horizon line still is a set of pixels that are part of a circle. The *Vector Calculation* performs a circle fit on these data points to find the corresponding radius and center coordinates. The desired vector is then the direction vector from the image center to the circle center. By comparing the results with expected values, errors can be detected and a further analysis is then executed on partial images during the *Division* step (cf. Barf, 2014, Sec. 2.1).

In the following sections several parameters of the algorithm are introduced. Reasonable values of these parameters had to be determined by simulations during development, or in case of physical constants, adapted from literature. The ones, that model the flight on REXUS satisfactorily and therefore were used in the implementation of HORACE, are listed later in Tab. 1.

4.2.1. Preprocessing

To decrease the calculation time, the image is scaled down by a bilinear interpolation algorithm from a height of H_{original} to H_{scaled} pixels, where the proportions of the image are kept. The ratio of these two values is the scaling factor

$$s := \frac{H_{\text{scaled}}}{H_{\text{original}}} \quad (1)$$

Table 1: Values of the Parameters in the HORACE Implementation. (d and p from LENSATION (2014) & MATRIX VISION GmbH (2012))

Symbol	Value	Unit
n	14213	# of pixels
b_{max}	235	8-bit value
b_{min}	15	8-bit value
b_{factor}	0.75	
z	5000.0	pixel
h_{min}	0.0	m
h_{max}	100.0	km
d	8.0	mm
p	3.75	μm
r_{cb}	6371.0	km
H_{scaled}	400	pixel
H_{original}	768	pixel

However, there is a trade-off between calculation time and accuracy. The parameter H_{scaled} must be set regarding both requirements. Moreover, determining the brightness b of the picture is a key part of the *Preprocessing* step. Fig. 2a represents a typical image with disturbances. b is calculated by taking the average of the brightness of a number n of random pixels, where the brightness $b(p_i)$ of a pixel i at position p_i is the mean value of its channels. Images that exceed certain limits (b_{max} and b_{min}) are then rejected from further processing. The values n , b_{max} and b_{min} are parameters of the algorithm (cf. Barf, 2014, Sec. 2.2.1). A higher n increases the accuracy of the brightness measurement, but also increases the processing time. b_{max} and b_{min} must be determined experimentally for every camera system and configuration.

4.2.2. Threshold Filter

The *Threshold Filter* is a dynamic filter, as it converts the image with a k -Bit grayscale resolution, like in Fig. 2b, to a binary image by taking the overall brightness of the picture into account. The value $f(p_i)$ of a pixel i at position p_i gets the values 0 or 1 in the following manner

$$f(p_i) \leftarrow \begin{cases} 1, & \text{if } b(p_i) \geq b \cdot b_{\text{factor}} \\ 0, & \text{else} \end{cases} \quad (2)$$

The factor b_{factor} is a parameter of the algorithm and must be determined experimentally. The result of this conversion can be seen in Fig. 2c.

4.2.3. Line Detection

The *Line Detection* is a topological search with a subsequent selection of one line. The implemented algorithm was introduced by Suzuki & Abe (1983, Algorithm 2). Its basic behavior is to search for a pixel that satisfies the criteria for a border pixel, to follow that border until the starting pixel is reached and then to resume the search of

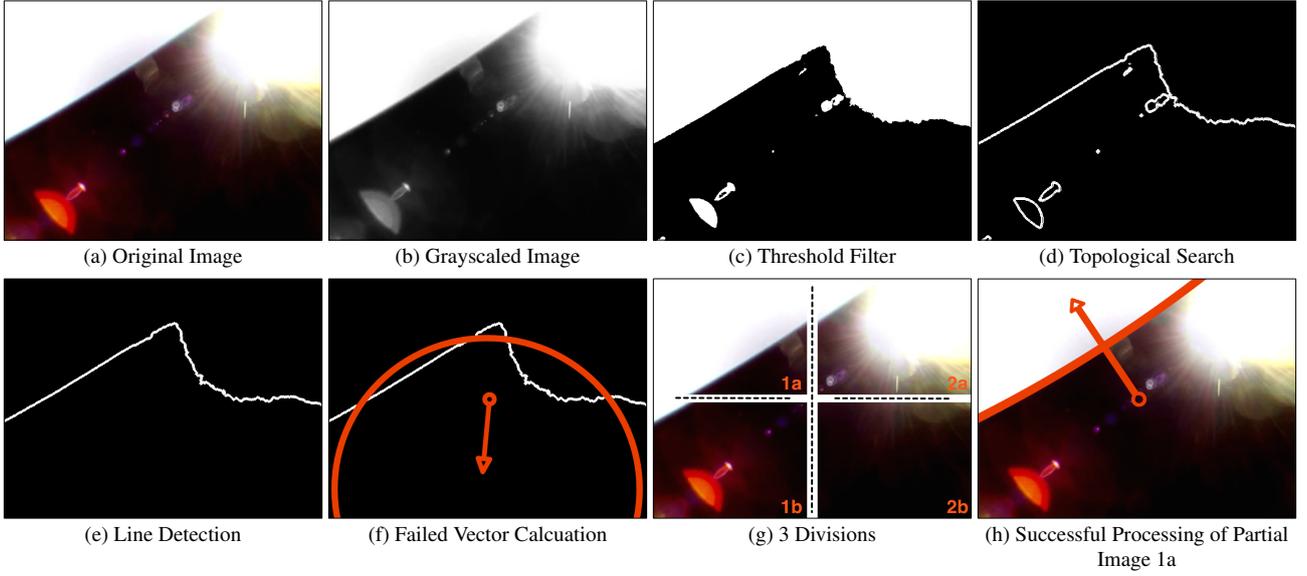


Figure 2: States of Image data during Processing (Barf, 2014, Sec. 2)

further border pixels. The output of this algorithm applied to Fig. 2c is illustrated in Fig. 2d (note: image was enhanced). Every border is marked individually, so that the longest border can then be selected as the horizon line (cf. Fig. 2e; note: image was enhanced) (cf. Barf, 2014, Sec. 2.2.3 Method 1).

4.2.4. Vector Calculation

Once the horizon-line is found the data of all pixels forming the line must be used to determine the circle, the line is part of. Therefore, a circle fit is performed by the least square method. This method delivers the radius r_s and center $(c_{x,s}, c_{y,s})$ of the circle in the scaled image frame coordinates. The values in the non-scaled coordinate system are then

$$r = \frac{r_s}{s} \quad (3)$$

$$(c_x, c_y) = \left(\frac{c_{x,s}}{s}, \frac{c_{y,s}}{s} \right) \quad (4)$$

Because of disturbances, like the sun or other orbs, the calculation may be wrong. The failed *Vector Calculation* is shown in Fig. 2f (note: image was enhanced). To detect failed calculations, the radius of the circle is compared to the expected radius. Taking the optical system into account, the limits for a valid radius in pixel can be determined by

$$r_{p,\max} = \frac{r_{cb}d}{p\sqrt{h_{\min}(h_{\min} + 2r_{cb})}} + z \quad (5)$$

$$r_{p,\min} = \frac{r_{cb}d}{p\sqrt{h_{\max}(h_{\max} + 2r_{cb})}} - z \quad (6)$$

where r_{cb} [m] is the radius of the central body, d [m] is the distance between the lens and optical sensor surface, p [m/pixel] is the width of one pixel on the sensor, h_{\min} [m] and h_{\max} [m] are the minimal and maximal possible heights above ground of the satellite and z [pixel] is a parameter to add some margin (cf. Barf, 2014, Sec. 2.2.4). By determining z experimentally, the atmosphere and the albedo of the central body can be taken into account. This parameter must be chosen wisely: too high values cause false positives whereas too low values cause false negatives. The term "false negative" in this context stands for a failed detection of an image that actually has a well defined horizon in it, whereas the term "false positive" stands for a detection of an horizon although no horizon is visible in the image. If the determined radius r exceeds the limits $r_{p,\max}$ and $r_{p,\min}$, the calculation is invalid. For successful calculations, the vector from the center of the image to the center of the circle is calculated. Note that the parameters z , $r_{p,\max}$ and $r_{p,\min}$ can be significantly larger than the actual dimensions of the image, as depending on the other quantities it is probable that only a small fraction of the circle is seen in the image.

4.2.5. Division

If the calculation in the previous step indicates that the radius is not valid, the image is divided in two parts by cutting the longer side in the middle. As illustrated in the activity diagram in Fig. 3, this is also done for every resulting image until the image is either too small for further calculation, was rejected in the *Preprocessing* step or returned a valid result. The result of the whole image is then the average of all results of the partial images. Thanks to this divide and conquer technique it is possible to distinguish between the central body and disturbances.

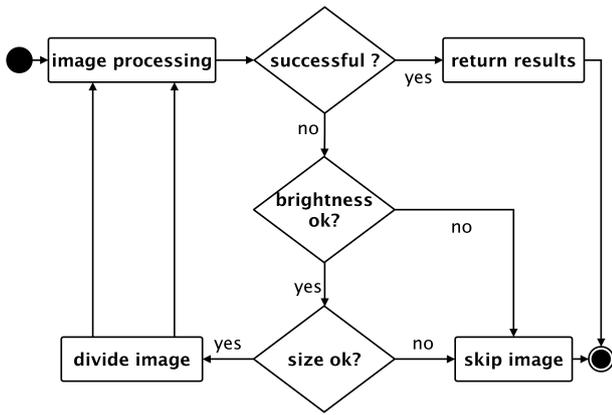


Figure 3: Division Activity Diagram (Barf, 2014, Figure 2.10)

Thus, the sensor is still operable if there are other orbs in the field of view. In the case of the example (cf. Fig. 2g), the image is divided three times: First into images 1 and 2, whereupon two successive cuts of both images result in images 1a, 1b, 2a and 2b. Image 2b is rejected during the *Preprocessing* due to a low brightness. In the images 2a and 1b, almost no horizon is visible. Hence, they are divided until they are too small. Only image 1a is processed successfully and delivers a result that is representative for the whole image (cf. Fig. 2h).

5. RESULTS AND DISCUSSION

The scientific data collected during the flight on REXUS 16 was unfortunately severely degraded due to a complicated malfunction of the payload cameras. It was only detected after the recovery of the experiment and caused heavy overexposure of the recorded footage (cf. Rapp, 2014, Sec. 2.4 for further failure investigation). With this set of data not a single horizon acquisition was possible, hence it was neither possible to prove nor to disprove the concept under space-like conditions as intended in the first place. Nevertheless, data from simulations which were conducted throughout the development of the experiment, the evaluable flight data and comparison with existing systems of similar complexity, revealed valuable results. Furthermore, several lessons regarding the design and operations of an experiment on sounding rockets were learned, which are discussed as well to help to improve future experiments.

5.1. Performance of the Algorithm

From the collected data during the flight it can be determined that the number of false positives as well as the number of false negatives was zero. The master system reached a frame rate of processed frames of 2.9 fps and the slave, 2.0 fps. The time needed to process an image was between 50ms and 70ms. Unfortunately, since the

images provided by the camera were completely overexposed, all frames were rejected. Thus, although the algorithm worked nominally, this set of data is not representative. In a nominal case, the processing time is expected to be higher (80ms - 200ms). The discrepancy between the frame rates and the processing time can be explained by the scheduling process of the operating system. It also degrades conclusions that can be drawn from the time data, since one frame may not get processed completely during one CPU time cycle, but paused and resumed multiple times (cf. Rapp, 2014, Sec. 3.5.3). To be able to make statements regarding the accuracy and occurrences of false positives and false negatives despite the degraded flight data, a simulation has been performed and evaluated by Rapp (2014, Sec. 3.5.4). It revealed a rate of false positives of 0% and a rate of false negatives of 10.29%, which occurred mostly when the horizon was nearly out of view. This happens because the algorithm needs a minimum number of data points to perform the circle fit. The accuracy, consisting of the offset from mean plus standard deviation, was determined to be $\pm 0.6^\circ$.

5.2. General Evaluation

Apart from the evaluation of the algorithmic performance, some more general questions were posed for investigation as mission objectives. To follow the trend towards clusters of small satellites instead of a single large satellite, the applicability of an horizon sensor based on image processing techniques for small satellites was to be evaluated. Of course, HORACE as the first step of basic research was not optimized regarding factors like mass, size or power consumption. Nevertheless, the chances are good that a future horizon sensor system for attitude determination will be applicable also for small satellites. This is thanks to the fact that no severe increase of algorithmic complexity was experienced compared to other systems, like the star tracker *STELLA* (Balagurin et al., 2011). This star tracker in particular is also based on image processing techniques and already proven to be applicable for small satellites. Hence, one can expect the hardware demands for the horizon acquisition to be in the same order of magnitude (cf. Rapp, 2014, Sec. 4). Furthermore, already throughout the development and the evaluation, one could identify the camera and the threshold filter as systematical limits for the concept, as both severely affect the algorithm's accuracy. While the development of higher autonomy for the threshold filter instead of experimentally determined values can overcome this systematical limit, deficiency of the raw-input data can hardly be compensated by the algorithm. Hence, in the case of a pure software solution (using existing optical instruments) degraded accuracy has to be expected, whereas for a solution with included hardware, a suitable device has to be selected.

5.3. Further Lessons Learned

In addition to the scientific results related to the mission itself, valuable experience regarding design and operations of a REXUS experiment were gained.

5.3.1. GPS-Interference

Only very late during one of the last tests before the launch opportunity it was unveiled that HORACE severely interfered with the GPS module of the RXSM, making a GPS fix impossible when HORACE was powered on. The exact source of the interference could not be identified on-site and direct measures (ferrite cores placed around various cables) did not solve the problem. But as the GPS data was not essential for any of the experiments or the vehicle itself, it eventually was approved to fly REXUS 16 without GPS information between T-600s and T+600s. The reason why this issue was discovered only that late and therefore could not be resolved before flight was that the exact flight configuration was run for the very first time during the mentioned test. Earlier test setups were different from the exact flight configuration especially regarding the nose-cone. It contains the GPS receiver but was not mounted to the experiment modules, hence also the corresponding cables were not fed through the payload stack. Further post-flight analysis with very basic methods showed that the SATA-II connection between the core system and the SSD, whose bus frequency bandwidth overlaps with the GPS bands, was the source of interference. It was not possible to pinpoint the exact blamable component involved in the SATA-II connection (core system, cable or SSD), therefore shielding all three together would have been the only dependable mitigation method. But this countermeasure would not have been possible even if the source had been located during the launch campaign due to too many changes of the experiment design. Hence, it is strongly recommended to test the very exact flight configuration latest during Bench Test (in case of REXUS (Mawn et al., 2014), or comparable stages for other vehicles) and the launch operators are encouraged to raise awareness of possible interference between GPS and SATA among experimenters with appropriate notice in the user manuals and during design reviews (cf. Rapp et al., 2014, Sec. 7.4.1.3).

5.3.2. Shifted Ground

Although the grounding concept of HORACE was designed in accordance with the guidelines of the REXUS manual, the core systems caused a shift of the ground potential within the HORACE flight segment. Proper grounding of the structure to the start point presumably then caused uncontrolled stray currents which led to unpredictable crashes of the flight software of the master core system. To this day the issue is not fully understood. Exchanging the master core system with a spare component and workarounds to safely restart the flight software

in case of a crash only reduced the probability and impact of the issue to an acceptable level but did not resolve it entirely. Complete electric isolation of the core systems is the only proposed mitigation method so far. In similar vein to the interference with GPS, this was mainly caused by insufficient testing and the failure to identify the software crashes and their cause in time for remedy prior to the launch. With properly grounded test setups this issue may have been avoided. Therefore, again we strongly recommend the launch operators to advise experimenters about proper grounding of the experiment via the structure during testing in the user manuals. Furthermore, the operators' staff shall ensure the proper grounding during tests with equipment the experimenters are not familiar with, like in case of REXUS the RXSM-simulator and RXSM itself (cf. Rapp et al., 2014, Sec. 7.4.1.2).

6. FURTHER WORK & OUTLOOK

In the scope of the HORACE project only the very basic research was conducted and due to the deficiency of the scientific data some questions are still open or only indirectly answered. So, there is still a long way to go to a fully operational system. But the first further steps towards that long term goal are already done or in progress. For instance, in the scope of Barf (2014) the original algorithm using the OpenCV-library was upgraded according to common coding directives for spaceflight and using standard C++ libraries only. Hence, it can be included as is in nearly any other experiment. Furthermore, a follow-up experiment, PATHOS, was selected for cycle 08 of the REXUS/BEXUS programme and is currently in the critical design phase and to be launched in early 2016. The aim of PATHOS is not only to overcome the problems of HORACE but also to miniaturize the system and separate the actual horizon acquisition from interface functionalities (Wagner et al., 2015, Sec. 1.2).

Further steps would be proper quantitative qualification of the system by flying it with separate accurate attitude determination sensors for reference (cf. Rapp, 2014, Sec. 5.1) and an upgrade to a full attitude determination as outlined in Rapp (2014, Sec. 5.2) would be the next steps. Of course, there are also several possibilities to improve the algorithm itself. The two main starting points would be to improve autonomy of the threshold filter, which is so far the most critical step, as it could drastically increase the accuracy, and to optimize the *Division* step, as it is likely to improve the calculation time (cf. Barf, 2014, Sec. 6).

7. CONCLUSION

Despite the degradation of the flight data due to the overexposure of the payload cameras, valuable scientific data could be retrieved, revealing main starting points for further work. We showed that the outlined approach of an horizon sensor, which is based on image processing techniques applied to optical data in the visible spectrum,

could achieve an accuracy of at least $\pm 0.6^\circ$ and calculation times in the order of 100ms to 200ms. Hence, it is not only accurate and fast enough for coarse attitude determination especially during emergencies, but also likely to be applicable even for small satellites. Furthermore, as all other subsystems worked nominally, a profound test platform to face the scientific questions as well as the algorithm for horizon acquisition for attitude determination was developed. Because this development was also defined as part of the complete mission, it greatly contributes to the overall partial success of 80% of the HORACE project (cf. Rapp, 2014, Sec. 4).

Evaluating the whole project, one has to take into account not only the scientific and technical aspects, but the educational aspects as well. With the great opportunity to conduct the project in the scope of the REXUS/BEXUS programme already during their Bachelor's studies, the contributing students gained deep insight into all aspects of the complete life cycle of a space project and thus into their future field of profession. To gain those experiences as early as possible and in a project frame where failures are "allowed" and do not have severe consequences, is the main purpose of the REXUS/BEXUS programme. As they are invaluable for the students' future careers, the gained experiences outweigh all deficiencies of the scientific results.

ACKNOWLEDGMENTS

First of all we want to thank Prof. Dr. Hakan Kayal and Dipl.-Inf. Gerhard Fellingner from the Chair of Aerospace Information Technology at University of Würzburg, not only for financial support and providing access to the chair's facilities and resources but also for their supervision throughout the whole project. Furthermore, we thank the German Aerospace Center (DLR) and the Swedish National Space Board for providing the programmatic frame of the REXUS/BEXUS programme which made the whole HORACE project possible in the first place, and all staff from all associated agencies and institutes – besides already mentioned, European Space Agency, Center of Applied Space Technology and Microgravity (ZARM), DLR Mobile Rocket Base and SSC Esrange – for the professional realization always on a par with the student teams. Finally, we owe great thanks to all other supporters and partners, individuals and those from industry, who supported us in valuable ways too numerous to be listed here.

REFERENCES

Balagurin, O., Wojtkowiak, H., & Kayal, H. 2011, in 8th IAA Symposium On Small Satellites for Earth Observation, IAA-B8-1201, Berlin

Barf, J. 2014, Development and Implementation of an Horizon Sensing Algorithm based on Image Processing Technologies (Julius Maximilian University of Würzburg, Am Hubland, D-97074 Würzburg)

Griffin, A. 2015, Arch Linux, <https://www.archlinux.org>, retrieved: 26 May 2015

LENSATION. 2014, Specification Lensagon BT8020N

MATRIX VISION GmbH. 2012, Technical details mvBlueCOUGAR-X

Mawn, S., Schmidt, A., Hellmann, H., et al. 2014, REXUS User Manual v7.13, EuroLaunch

QT Company. 2015, Qt — Cross-platform application & UI development framework, <http://www.qt.io>, retrieved: 26 May 2015

Rapp, T. 2014, Development and Post-Flight-Analysis of HORACE the Horizon Acquisition Experiment (Julius Maximilian University of Würzburg, Am Hubland, D-97074 Würzburg)

Rapp, T., Barf, J., Bergmann, M., et al. 2014, HORACE Student Experiment Document v 5.0, this document was written in the context of a REXUS project

Suzuki, S. & Abe, K. 1983, Topological Structural Analysis of Digitized Binary Images by Border Following (Shizuoka University, Hamamatsu 432, Japan)

Wagner, D., Aicher, M., Ehnle, J., et al. 2015, PATHOS Student Experiment Document v 1.1, this document was written in the context of a REXUS project

Willow Garage. 2014, OpenCV website @ONLINE, <http://opencv.org>, retrieved: 26 May 2015

Wolz, F. 2015, Entwicklung und Aufbau der Elektronischen und Mechanischen Komponenten von HORACE (Julius Maximilian University of Würzburg, Am Hubland, D-97074 Würzburg)