DISSERTAÇÃO DE MESTRADO

SENSORES DE ESTRELAS (STAR TRACKERS) DE BAIXO CUSTO PARA NAVEGAÇÃO AUTÔNOMA NO SISTEMA SOLAR

LOW COST STAR TRACKERS FOR AUTONOMOUS SOLAR SYSTEM NAVIGATION

DANIEL ARAÚJO MIRANDA

Brasília, Agosto de 2017

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

SENSORES DE ESTRELAS (STAR TRACKERS) DE BAIXO CUSTO PARA NAVEGAÇÃO AUTÔNOMA NO SISTEMA SOLAR

DANIEL ARAÚJO MIRANDA

DISSERTAÇÃO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA MECÂNICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM SISTEMAS MECATRÔNICOS

APROVADA POR:

Prof. Dr. Fabiano Araújo Soares, GAMA/UnB
Orientador

Prof. Dr. Carlos Humberto Llanos, PPMEC/UnB
Membro Interno

Prof. Dr. Giancarlo Santilli, GAMA/UnB
Membro Externo

BRASÍLIA/DF, 24 DE AGOSTO DE 2017

Araújo Miranda, Daniel
Low cost Star Trackers for autonomous solar system navigation - Sensores de estrelas (star trackers) de baixo custo para navegação autônoma no sistema solar / DANIEL ARAÚJO MIRANDA. –Brasil, 2017.
103 p.

Orientador: Fabiano Araújo Soares

Dissertação (Mestrado) – Universidade de Brasília – UnB
Faculdade de Tecnologia – FT
Programa de Pós-Graduação em Sistemas Mecatrônicos – PPMEC, 2017.

1. Star tracker. 2. Low cost. 3. Space Navigation. 4. Image processing. 5. Autonomous. I. Fabiano Araújo Soares, orientador. II. Universidade de Brasília. III. Faculdade de Tecnologia.

Agradecimentos

A Deus, meu Pai do céu, e a todos os que tornaram este trabalho possível: Clara, meus filhos, Lídia, Prof. Fabiano e Prof. Daniel, Prof. Clóvis, Everton, Prof. Lotufo, minha família, meus amigos e aos professores Carlos, Giancarlo e Cristiano

DANIEL ARAÚJO MIRANDA

RESUMO

Neste trabalho é demonstrado o uso de câmeras de baixo custo para determinar a orientação e posição de satélites e sondas espaciais. O Sensor de Estrelas (Star Tracker) é um dispositivo que determina a orientação - ou atitude - de uma espaçonave utilizando a posição de estrelas em uma imagem do céu. É possível triangular a posição do observador se houver planetas nas imagens. Sensores de Estrelas têm sido utilizados para a determinação de atitude desde ao menos 1964. A posição e a velocidade, no entanto, têm sido calculadas principalmente utilizando interferometria do sinal de comunicação, o seu desvio Doppler e o seu tempo de trânsito. Uma metodologia para implementação do algoritmo do sensor de estrelas e triangulação em hardware de baixo custo é apresentada e resultados obtidos a partir da superfície da terra são apresentados para a câmera Raspberry Pi NoIR e uma Nikon D5000. Os resultados apresentam erro angular próximo ao limite de difração de Rayleigh dos sistemas ópticos e um erro de posição para uma única sessão de medição cerca de 200 vezes maior que os resultados obtidos com filtro de Kalman a bordo da sonda Deep Space 1.

ABSTRACT

In this work we demonstrate the use of low cost cameras to determine orientation and position of spacecraft. Star Trackers are devices which determine orientation - or attitude - of a spacecraft using the position of stars in a picture of the sky. It is possible to triangulate the position of the observer if there are planets in the field of view. Star Trackers have been used for attitude determination of spacecraft since at least 1964. Position and velocity, however, have been calculated mostly using interferometry of the communication signal, it's Doppler shift and round-trip time. Deep Space 1 launched in 1998 with the first experimental autonomous navigation using cameras. A methodology is presented to implement the star tracker algorithm and the position triangulation algorithm using low cost hardware and Earth based results are presented with the Raspberry Pi NoIR camera and a Nikon D5000. The results after a few minutes of measurement have an angular error near the Rayleigh diffraction limit of the optical systems and positional error for a single measurement session about 200 times larger than the Kalman filtered measurements taken from space by Deep Space 1.

SUMMARY

RESUMO ..... **i**

ABSTRACT ..... **ii**

FIGURE LIST ..... **iv**

TABLE LIST ..... **viii**

1 Introduction **1**
1.1 Objectives 5
1.2 Main contributions 5

2 State of the art **6**
2.1 Missions 6
2.1.1 Deep Space 1 7
2.1.2 Stardust 9
2.1.3 Hayabusa 9
2.1.4 Rosetta 10
2.1.5 Deep Impact 11
2.1.6 Hayabusa 2 12
2.1.7 OSIRIS-REx 12
2.2 Algorithms 12
2.2.1 Star identification 13
2.2.2 Accuracy increase by using Kalman filters 15
2.2.3 Star identification without camera calibration 16
2.3 State of the art summary 16

3 Methodology **18**
3.1 Hardware 19
3.1.1 DSLR Camera 19
3.1.2 Raspberry Pi Camera Module 20
3.1.3 Older generation compact camera 21
3.2 The star catalog dataset 22
3.3 The *starprint* 25
3.4 Star Catalogue processing 28
3.4.1 Catalogue cleanup 28
3.4.2 Calculate *starprints* of reference stars 28

3.4.3 Coordinate change and epoch adjustment 30
3.5 Acquisition 31
3.5.1 Planning 31
3.5.2 Software 31
3.5.3 Parameter selection 32
3.5.4 Geometric Camera Calibration 34
3.5.5 Radiometric Camera Calibration 35
3.6 Image processing 37
3.6.1 Preprocessing 37
3.6.2 Segmentation 41
3.6.3 *Starprint* calculation 42
3.7 Attitude calculation - star tracker algorithm 42
3.8 Position calculation 44
3.8.1 Planet absolute positions 45
3.8.2 Planet identification and position calculation 45

4 Results 47
4.1 Star identification algorithm 47
4.1.1 Test under standard conditions 47
4.1.2 Test with dark field of view obstruction 53
4.1.3 Test with blurred stars 58
4.1.4 Test with high quality image 62
4.1.5 Test with few stars 70
4.1.6 Test with picture from the surface of Mars 73
4.1.7 Test with foreign body: city lights 78
4.1.8 Summary 83
4.2 Triangulation algorithm 84
4.2.1 Nikon D5000 - BSC 84
4.2.2 Nikon D5000 - Hipparcos 86
4.2.3 Raspberry Pi Camera NoIR version 1 89
4.3 Error sources 92
4.3.1 Diffraction limit 92
4.3.2 Light speed delay 93
4.3.3 Geometric distortion 94
4.3.4 Noise (centroid finding accuracy) 94
4.3.5 Angular motion of camera (centroid finding accuracy) 94
4.3.6 Catalog inaccuracies 94
4.3.7 Tadeoff between tolerances and field of view for star identification 95

5 Conclusion 96
5.1 Future work 96

FIGURE LIST

1.1 Astro APS star tracker developed by Jena Optronik. Technician is holding the baffle around the camera lens. Picture published by the European Space Agency in 2013 (26). 1
1.2 A 2004 cartoon strip by Caltech's Jet Propulsion Lab (27) illustrates the star tracker algorithm. .. 2
1.3 Modern star trackers from Berlin Space Technologies and Hyperion Technologies B.V. (2). With an accuracy between 10 and 30 arcseconds ($0.0028°$ to $0.0083°$) and 0.65W of power, they are much lighter and consume much less power than the 2001 models identified in **Table 1.1**. ST400 has been in spacecraft since 2015. 2
1.4 Paluszek's 2-dimensional illustration of the geometry of the problem of determining the position of the observer. 4
1.5 Paluszek's prototype mounted on a heavy duty tripod (Figure 13 from (52)). 4

2.1 Comparison of current navigation approach versus autonomous optical navigation by NASA's JPL (46). 8
2.2 Orbital errors in AutoNav calculations in relation to conventional measurements for Deep Space 1 (4). 8
2.3 Sequence of STARDUST AutoNav images of Wild 2 surround encounter (4). 9
2.4 STARDUST cruise configuration showing the navigation camera periscope and the two star trackers (48). 10
2.5 Hayabusa geometry of reference frames (33). 10
2.6 Hayabusa landing site: MUSES-Sea (33). 11
2.7 Asteroid 2867 Steins as seen from the ROSETTA navigation camera on August 4th 2008. The extended white spots are stars. Almost all other small white spots represent noise from the CCD sensor (pixel dark current) (37). 11
2.8 Impactor part of the Deep Impact Spacecraft. High precision star tracker, Impactor Target System (ITS) and hydrazine course correction thruster shown but not highlighted. Source: (41). . 12
2.9 Basic flow of the star trackers' processing. The square indicates the processes involved in star identification. (Figure 1 from (20)) 13
2.10 Basic pattern matching. Robust, but requires a large number of stars. (Figure 2 from (20)). 14
2.11 Measurement of angular separations between stars (Figure 3 from (20)). 14
2.12 Tracking stars from the previous frame to solve the current frame.(Figure 3 from (20)) 14
2.13 Star tracker, horizon sensor and inertial navigation system (INS) integration block diagram (Figure 6 from (56)) 15
2.14 Comparison of errors obtained by INS, INS-star tracker combination (without KF), and INS-star tracker integration (with KF) for a ballistic missile simulation. (A) is the height error, (B) is the longitude error and (C) is the latitude error. (Figure 8 from (56)) 16

3.1 Outline of the star identification algorithm. Square cornered cells represent input and output data, round cornered cells represent processing steps. 18
3.2 Lenovo Thinkpad notebook used to start the capture. The Nikon D5000 and the Raspberry Pi camera are controlled by a Raspberry Pi V2 minicomputer powered by a TP-Link 10400mAh Power Bank. 19
3.3 Nikon D5000 mounted on a tripod ready to start dark frame calibration. 20
3.4 Assembly of the Raspberry PI V2 with the NoIR camera on a small tripod, connected to a TP-Link 10400mAh Power Bank, the Nikon D5000 via USB and the notebook via Ethernet. The shroud is used to perform dark frame calibration. 20
3.5 Canon PowerShot S1 IS with 3.2 megapixel CCD (15) 21
3.6 Bright star availability for a camera with a 40° **circular full field of view** centered at each pixel. 22
3.7 Bright star availability for a camera with a 20° **circular full field of view** centered at each pixel. 23
3.8 Bright star availability for a camera with a 10° **circular full field of view** centered at each pixel. 24
3.9 Bright star availability for a camera with a 2.5° **circular full field of view** centered at each pixel. 24
3.10 Necessary detection performance (y axis) as a function of field of view diameter (x axis). Logarithmic plot of **Table 3.1** with colors blue for **n=1**, green for **n=2**, red for **n=3**, light blue for **n=4**, pink for **n=5** and gold for **n=6** 25
3.11 *Starprint* diagrams for Sirius. Circles were drawn at a distance $R = \pi/9$ radians (20^o) and $R/8$. The stars in P are connected to S (see **Algorithm 1**). Axes depict coordinates of the projected stars as seen from a pinhole camera with unity focal distance. 27
3.12 First 100 characters of lines 89 to 95 of the Bright Star Catalog plain text version (21). Records number 92 and 95 were thought to be stars when the catalog was first made. Record 92 was later found to be a supernova and record 95 a globular cluster. 28
3.13 Stars in the BSC (21). Right ascension and declination lines shown. 29
3.14 Stars in the BSC (21) of the area displayed in **Figure 3.13** without the right ascension and declination grid. 29
3.15 Reference stars chosen from the Bright Star Catalogue (21) using a 4° radius circular window (θ=4°, σ=0.4°)The region is the same as the one in **Figure 3.13** and **Figure 3.14**. 30
3.16 Red led imaged at the edge of the frame of the Nikon 50mm f/1.8 D lens on a D5000 at ISO 200 with different apertures. 33
3.17 Histogram sum of 120 dark frames taken with the Raspberry Pi camera at ISO 800 and 1/2 s exposure time. The compensated histograms are the sum of the histogram of the 120 frames after subtracting the dark frame (fixed pattern noise). 36
3.18 Mean dark frame (fixed pattern noise) of the Raspberry Pi camera averaged over 120 dark frames. 37
3.19 Raw image obtained from the Raspberry Pi camera 38
3.20 Final image after all preprocessing steps. 38
3.21 300×300 crop of a Raspberry Pi camera image. 39
3.22 Bayer pattern illustration. Part 1 illustrates the ideal signal, part 2 shows the grayscale data acquired by the sensor at each pixel, part 3 shows the raw data with the sampled colors and part 4 shows a reconstruction of the original signal from the sampled data. Copyryght Anita Martinz (39) 39
3.23 Filters applied to the raw image after dark frame subtraction. 40
3.24 Segmentation of the image in **Figure 3.20** showing pixel areas above the detection threshold and the bad areas for star detection. Part of the building wall was automatically flagged as a bad region. 41

3.25 Catalog brightness versus measured brightness of stars identified in a jpeg image. Axes are logarithmic and have no units. An increment of 5 along each axis corresponds to a $100\times$ increment in brightness..... 43

4.1 Picture taken from the rooftop of a building in 2005-11-21 at 21h42 local time. An antenna, a lightning rod and part of a wall can be seen..... 48
4.2 Grid photographed by a camera of the same model from (14) before and after correcting with the distortion coefficients found from star identification..... 49
4.3 Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars..... 50
4.4 Star names as described in the BSC (21) for the stars identified in **Figure 4.1**..... 50
4.5 Paper map "The Night Sky" (6), the red square shows the alignment at 2005-11-11 at 21:45, the green circles show identified stars on **Figure 4.4**..... 51
4.6 "Southern Cross by Moonlight". New Zealand landscape photographed with a Canon 10D in January 2005, in Kaituna, Wairarapa, New Zealand. Exposure time was 10 s, with a 20mm lens at f/4 (54). A plant obstructs the view of the sky..... 54
4.7 Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars..... 55
4.8 Star names as described in the BSC (21) for the stars identified in **Figure 4.6**..... 55
4.9 Taurus constellation, long exposure, obtained from Starnamer's blog (63). The cluster of stars to the right of the image is the Pleiades globular cluster. 58
4.10 Right upper corner after application of Gimp's (30) difference of gaussians filter on **Figure 4.9** .. 59
4.11 Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars..... 60
4.12 Star names as described in the BSC (21) for the stars identified in **Figure 4.11**. 60
4.13 High quality picture of the Capricornus constellation obtained from (9). 63
4.14 Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars..... 64
4.15 Star names as described in the BSC (21) for the stars identified in **Figure 4.14**. 64
4.16 Image of Centaurus and the Southern Cross, with heavy light pollution. Obtained at (63). 71
4.17 Result of applying Gimp's (30) difference of gaussians filter on **Figure 4.16**..... 71
4.18 Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars..... 72
4.19 Star names as described in the BSC (21) for the stars identified in **Figure 4.18**. 72
4.20 Image of the Orion constellation taken by the Spirit (50) hover from the surface of Mars with an 8 s exposure time. 74
4.21 Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars..... 75
4.22 Star names as described in the BSC (21) for the stars identified in **Figure 4.21**. 76
4.23 Picture of the Orion constellation taken with a Nikon Coolpix 885 in February 14th 2002 and exposure time of 10.27 taken from (28)..... 79
4.24 Edição da Figura 52 suprimindo a legenda em branco. 79
4.25 Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars..... 80
4.26 Star names as described in the BSC (21) for the stars identified in **Figure 4.25**. 81

TABLE LIST

1.1 Accuracy, mass and power of different attitude sensors listed in a university course tech report from 2001 (10) 2

2.1 Characteristics of spacecraft with autonomous navigation capabilities. 7
2.2 Summary of flybys which have used AutoNav until 2012. Adapted from (4). 8

3.1 Necessary magnitude detection performance for a star camera with a circular field of view of diameter ϕ to guarantee the detection of at least **n** stars in every region of the sky. 25
3.2 Part of the resulting starprint from **Figure 3.11**. The identifiers can be traced back to catalog stars: Alp CMa (2480), Kap Ori (1995), Gam Mon (2218), Bet CMa (2285), etc. The full table has 90 rows. 27

4.1 Matches found for **Figure 4.1** using the model **without distortions**. 52
4.2 Matches found for **Figure 4.1** using the model **with distortions**. 53
4.3 Matches found for **Figure 4.6** using the model **without distortions**. 56
4.4 Matches found for **Figure 4.6** using the model **with distortions**. 57
4.5 Matches found for **Figure 4.10** using the model **without distortions**. 61
4.6 Matches found for **Figure 4.10** using the model **with distortions**. 61
4.7 Matches found for **Figure 4.13** using the model **without distortions**. 65
4.8 Matches found for **Figure 4.13** using the model **with distortions**. 67
4.9 Matches found for **Figure 4.17** using the model **without distortions**. 73
4.10 Matches found for **Figure 4.17** using the model **with distortions**. 73
4.11 Matches found for **Figure 4.20** using the model **without distortions**. 77
4.12 Matches found for **Figure 4.20** using the model **with distortions**. 77
4.13 Matches found for **Figure 4.24** using the model **without distortions**. 82
4.14 Matches found for **Figure 4.24** using the model **with distortions**. 82
4.15 Comparison of cost and performance of solar system navigation alternatives. The underlined numbers are the error that was provided or measured in the experiments, the opposite column is a calculated quantity. 92

Chapter 1

Introduction

A star tracker system is a camera and a small computer that are embarked in a spacecraft to acquire and process images of the sky to calculate the orientation - or attitude - of the spacecraft. Such a capability is required for accurate pointing of instruments and for performing maneuvers. It consists of a fixed focus camera with a dedicated computer. The casing contains communication and power connections and has a distinct baffle around the camera lens that absorbs light coming from outside the field of view. In **Figure 1.1** a modern model is seen. The technician's hands in the picture give a sense of the scale of the equipment.

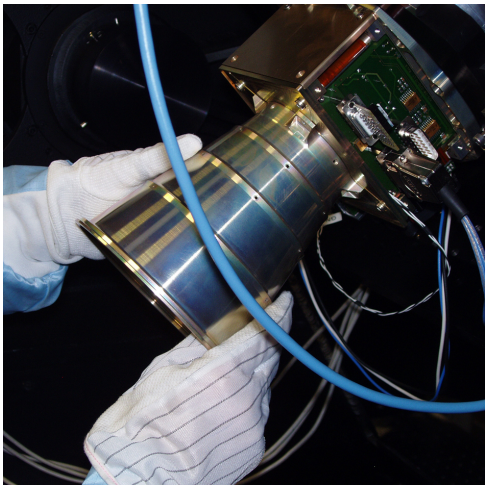


Figure 1.1: Astro APS star tracker developed by Jena Optronik. Technician is holding the baffle around the camera lens. Picture published by the European Space Agency in 2013 (26).

The basic operation of the star tracker involves acquiring an image, comparing it to a database and inferring where the camera is pointing to based on the identification and relative position of the imaged stars. **Figure 1.2** illustrates the operation of a star tracker, however the cartoon is not entirely accurate since the information displayed would allow to calculate "where I am pointing to", not "where I am". Within the precision of current star trackers, the stars seem to be in the same position relative to each other whether they are being observed from one side of Earth's orbit of from the other. In fact, such relative changes in the star positions due to position of the observer (the star parallaxes) are so small that the first successful measurement attempt happened only in 1838 (3), 230 years after the invention of the telescope.

Star trackers have been successfully used for attitude determination of spacecraft from at least 1964 with the launch of the Mariner 4 probe to Mars (61) and are present in most space probes since that time (4). Those

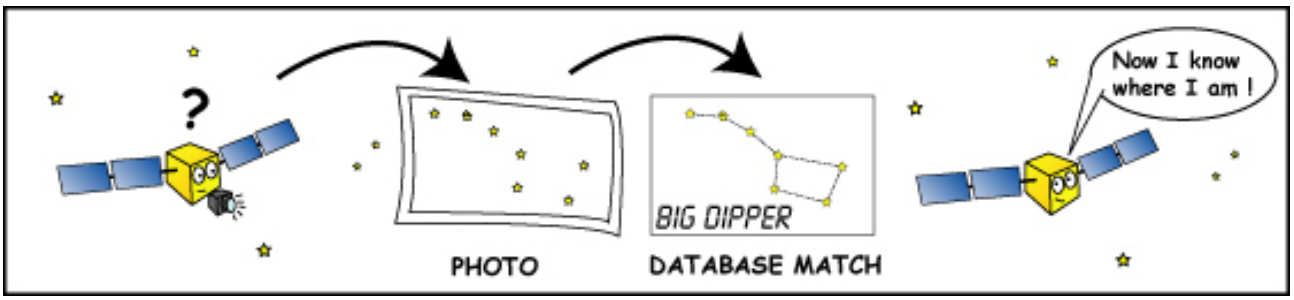


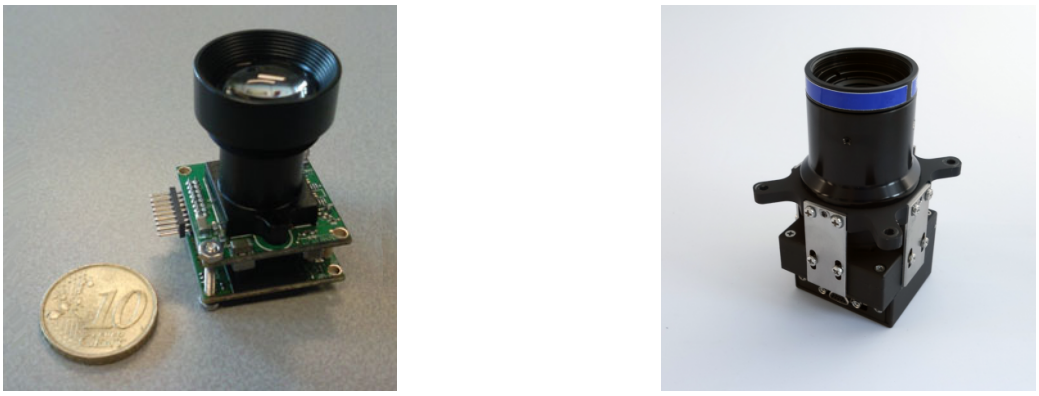
Figure 1.2: A 2004 cartoon strip by Caltech's Jet Propulsion Lab (27) illustrates the star tracker algorithm.

early instruments had several drawbacks in relation to their modern counterparts such as the need for moving parts, the need to point to a specific star, high weight and low sensitivity. In 1995 "low-cost, all purpose star trackers" became available in the commercial market which were able to measure attitude with a precision of 20 arc seconds (0.006° or 0.0001rad) (36) and where able to perform a measurement even if the initial attitude was unknown.

Table 1.1 contains a list from an university report from 2001 (10) comparing the precision of the attitude determination systems at the time. **Figure 1.3** shows an example of more modern and lighter star trackers.

Table 1.1: Accuracy, mass and power of different attitude sensors listed in a university course tech report from 2001 (10)

Sensor	Accuracy	Mass (kg)	Power (W)
sun sensors	0.005° to	0.1° to 2	0 to 3
star sensors	0.0003 to 0.01	2 to 5	5 to 20
horizon sensors	0.1 to 1	0.5 to 4	0.3 to 5
Magnetometers	0.5 to 3	0.3 to 1.2	<1



(a) ST200 star tracker: 30×30×38 mm, 40g **(b)** ST400 star tracker: 48×57×89 mm, 280g.

Figure 1.3: Modern star trackers from Berlin Space Technologies and Hyperion Technologies B.V. (2). With an accuracy between 10 and 30 arcseconds (0.0028° to 0.0083°) and 0.65W of power, they are much lighter and consume much less power than the 2001 models identified in **Table 1.1**. ST400 has been in spacecraft since 2015.

To fulfill mission objectives it is necessary to know not only the attitude, which the star tracker provides, but also the position and velocity of the spacecraft. Repeated observations of the doppler shift and round-trip time of the communication signal have been used to calculate the position and velocity of spacecraft (4); modern approaches use Very Long Baseline Interferometry (VLBI) (67) which measure the position of the radio source in relation to a frame of reference tied to quasars and can reach an accuracy of one nanorradian (11) - which means about 4.5km for a distance equal to Uranus' mean orbital distance to the sun or about 230m for Mars'

mean orbital distance to the sun. For spacecraft in Earth's orbit, it is also possible to use the global positioning satellite networks like GPS, Glonass, Galileo and Beidou, and can achieve a precision of 1m with commercial "off-the-shelf" components (40).

Optical navigation has been used from at least 1986 (64) in the Voyager spacecraft Uranus encounter. Images from cameras on the spacecraft can be used to calculate relative positions from observations, but until the Deep Space 1 mission the image processing has been performed on Earth (60). In early applications, an image of the camera was displayed to an analyst which identified the target body and the background stars and measurements were performed with the help of image processing algorithms that located the geometric center of the target body. The results of those measurements were merged with the radio data to produce orbit estimates (68).

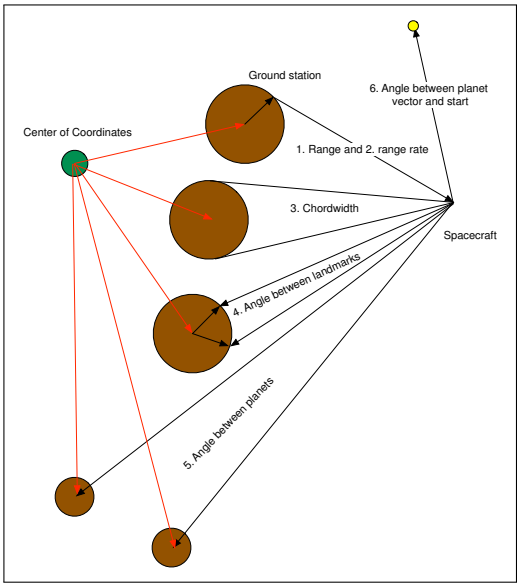
Star tracker hardware has become more capable with time, enabling ground-based processing of optical images for refinement of position (4). Position determination using Earth infrastructure, however, is limited by the speed of light which can impart a delay of several hours within the solar system. It is also limited by available antenna time of large ground based facilities that provide long range communications and telemetry, such as the three Deep Space Network (DSN) sites, which face an increasing load (35). DSN's antennas usage can be seen in real time at the DSN's website (22).

Spacecraft with electrical propulsion are good candidates to use that technology (18). Ion engines have very long "burn" times with a total duration of several years compared to several minutes for chemical propulsion. At the time of launch, the DAWN spacecraft had enough propellant to thrust continuously for more than 2000 (two thousand) days (44) and had a design life of 10 years (66). Monitoring a spacecraft that is under continuous thrust and therefore changing orbital parameters increases the burden on the ground infrastructure.

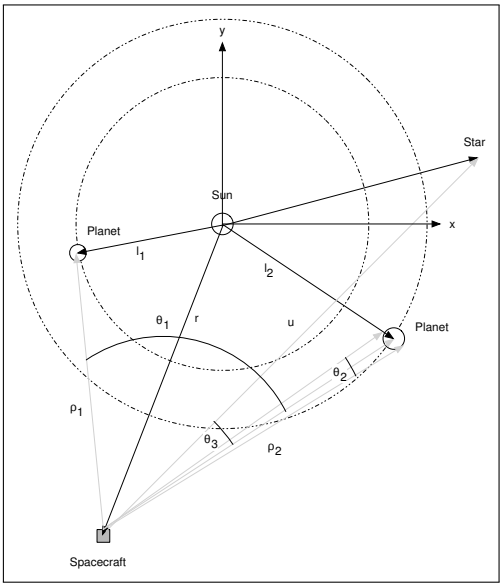
A low latency for position measurement and the execution of maneuvers enables missions not otherwise viable. The Deep Impact mission had the objective of impact the Tempel 1 comet with a relative speed of more than 10km/s. That feat was performed with the aid of autonomous navigation to provide the impactor with timely position updates for course correction (32).

Paluszek describes the geometry of the problem (52) and proposes a hardware setup and a prototype (**Figure 1.5**) to make the measurements from the ground. **Figure 1.4a** ilustrates the several variables that can be observed and **Figure 1.4b** ilustrates the measurement geometry. Simulation results were generated but no experimental results were displayed. The proposed hardware of that work involves high precision moving parts, the simulations take into account Kalmann filtering using several observations in the course of many months and the information from the internal inertial unit (IMU). In this work, we measure the equivalent of θ_3 in **Figure 1.4b** for several stars and at least two planets to estimate the position of the observer.

The study of techniques that enable autonomous navigation is therefore an important topic. Due to the weight and reliability constraints in spacecraft, it is desirable that such capability be achieved without the need for additional hardware. A successful implementation using star trackers may reduce or even eliminate the need of ground based techniques for tasks such as orbital maneuvers, antenna pointing and ephemeris (orbital parameters) estimate.



(a) Observable variables (Figure 1 from (52))



(b) Observation geometry (Figure 2 from (52))

Figure 1.4: Paluszek's 2-dimensional illustration of the geometry of the problem of determining the position of the observer.

Figure 1.5: Paluszek's prototype mounted on a heavy duty tripod (Figure 13 from (52)).

1.1 Objectives

The general objective of this work is to study the usage of low cost star trackers not only for autonomous attitude determination, but also for autonomous position estimation.

The specific objective (the problem definition) of this work is to determine whether it is possible and if so with what accuracy can the position of an observer inside the solar system be measured using images of planets with discernible stars in the background using modest hardware, specifically the same hardware that can be used for attitude determination.

The use of other non-planetary bodies such as the Earth's moon, the Sun, asteroids, comets and other planet's moons is out of the scope of this work.

Low cost star trackers are defined here as hardware and software able to provide attitude control and solve the "lost in space" problem with a hardware budget of under US$1000. This is a somewhat loose definition since the cost to make the hardware ready for space deployment is not taken into account and no performance constraints are enforced a priori. Additional hardware costs for space worthy hardware would likely include radiation shielding, sun shade, thermal adaptations, power supply adaptations, satellite bus physical interfaces and others. Performance constraints such as frequency of measurements and maximum angular speed tolerance are also not taken into account.

1.2 Main contributions

The first contribution of this work is a star tracker algorithm called *Starprint* that works with uncalibrated cameras wich improves on Klaus (31) by implementing a more efficient matching algorithm, using stars in the entire image and being robust to the appearance of planets and star-like objects in the scenes.

The second contribution of this work is a triangulation algorithm that works with low cost sensors and features:

- multi frame measurements
- light transit time correction
- integration with Nasa's Horizon database (47)
- image preprocessing and segmentation for low signal-to-noise ratio Bayer-sampled images
- angular and positional *lost in space* capability

The triangulation algorithm pre-process the images, performs segmentation for measuring the star-like objetcs, identifies the objects with the *Starprint* algorithm, identifies non stellar bodies, merges measurements of the non stellar bodies from several frames, determines the position of the observer and estimates the solution error.

The third contribution of this work is implementing and testing the algorithms on low cost hardware: a Raspberry Pi NoIR camera and a Nikon D5000 camera. Paluszek (52) proposes a much more complex hardware solution and only provides simulation results.

Chapter 2

State of the art

This chapter presents a review of the usage of Star Trackers for autonomously determining position of spacecraft since the Deep Space 1 mission in 1998 (46). Published studies and public information on past, current and planned missions are surveyed and described according to the hardware used and reference frame of positioning.

A few algorithms are also surveyed for the basic star tracker operation and for fusioning different sensors data to improve accuracy.

2.1 Missions

A number of missions were flown by NASA (North American Space Administration), ESA (European Space Agency) and JAXA (Japanese Space Agency) that demonstrate autonomous navigation. **Table 2.1** lists several missions with autonomous navigation capabilites. The DAWN mission (43) was left out of that list even though it features the necessary hardware, because it lacks autonomous navigation capability and relays data to Earth for processing and planning.

The extent of the autonomous capabilities of each mission is further described in the appropriate sections.

Table 2.1: Characteristics of spacecraft with autonomous navigation capabilities.

Mission	Agency	Launch date	Mission end	Specific hardware	Reference Frame
Deep Space 1	NASA	1998-10-24	2001-12-18	none	Solar system, target body orbit
Stardust	NASA	1999-02-07	2011-03-24	navigation camera	Solar system, target body orbit
Hayabusa	JAXA	2003-05-09	2010-06-13	laser rangefinder, wide view camera, target marker	Target surface marker
Rosetta	ESA	2004-03-02	current	dual navcams, science camera	Target body center
Deep Impact	NASA	2005-01-12	2011-10-01	framing camera, impactor target sensor	Solar system, target body orbit
Hayabusa 2	JAXA	2014-12-03	current	laser rangefinder, wide view camera, target marker	Target surface marker
OSIRIS – Rex	NASA	2016-09-03	planned	dual navcams, stowcam	surface features

2.1.1 Deep Space 1

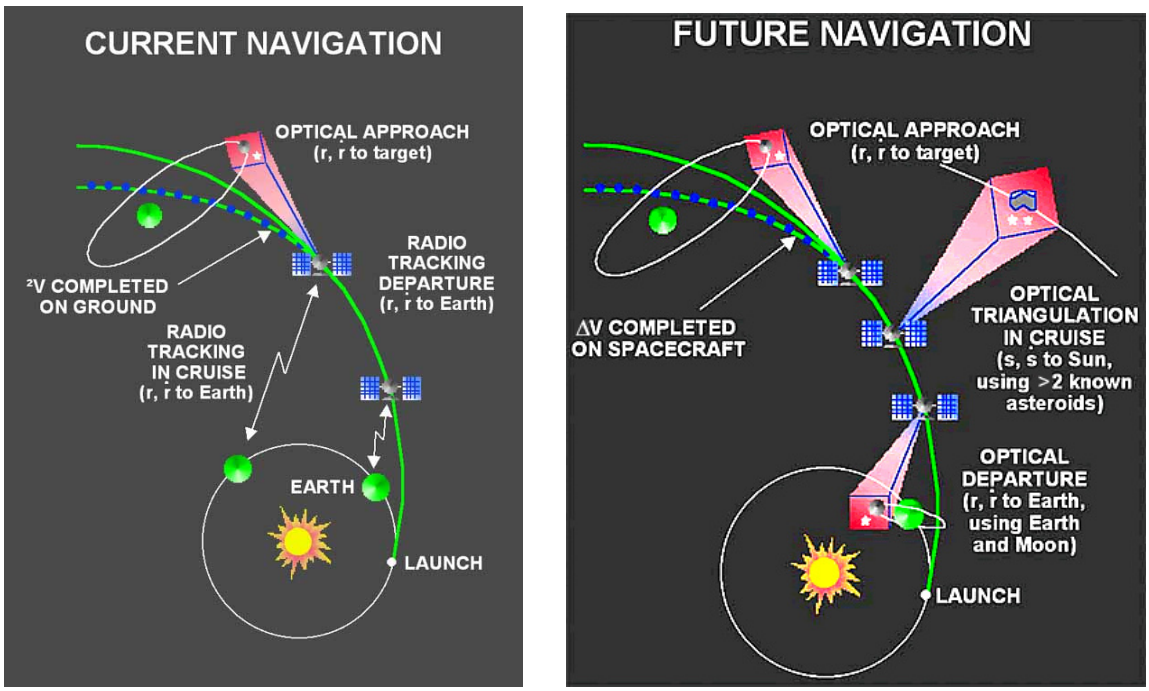
The Deep Space 1 spacecraft was launched in 1998 to test twelve advanced high-risk technologies in space amongst them the first autonomous navigation software - JPL's AutoNav (4). The software was able to use the star tracker cameras in the spacecraft to image bright known bodies against a star background, calculating a line-of-sight solution. Several such solutions can be used to estimate the orbital parameters as well as refine the relative position between the spacecraft and the target.

The Deep Space 1 autonomous navigation implementation did not use any specific hardware such as LIDAR (laser rangefinding) or RADAR (radio rangefinding). **Figure 2.1** shows the general idea of not relying on communications to ground and computing the trajectory correction maneuvers on the spacecraft. However, the mission team did use interferometry, the doppler shift and time-of-travel of the communication signal for comparison purposes, obtaining errors in positioning under 1000km as depicted in **Figure 2.2**.

Obtaining the images for calculating the position required that the spacecraft performed weekly activities to aim the star tracker at the available targets.

As a test platform for experimental technologies, DS1 was expected to encounter problems. The star tracker cameras used were found to have lower performance than expected. At one point the star tracker failed, but the mission crew was able to use one of the imaging instruments, MICAS, in substitution for the star tracker. The mission was considered successful with regard to autonomous navigation in spite of the difficulties found.

The AutoNAV software was also used in the STARDUST and Deep Impact missions, as well as their follow-ons EPOXI (the follow-on mission to Deep Impact) and STARDUST NExT (the follow-on mission to STARDUST). **Table 2.2** lists the flybys where the software was used until 2012. No further references to the use of AutoNav in missions where found.



(a) Current navigation

(b) Future navigation

Figure 2.1: Comparison of current navigation approach versus autonomous optical navigation by NASA's JPL (46).

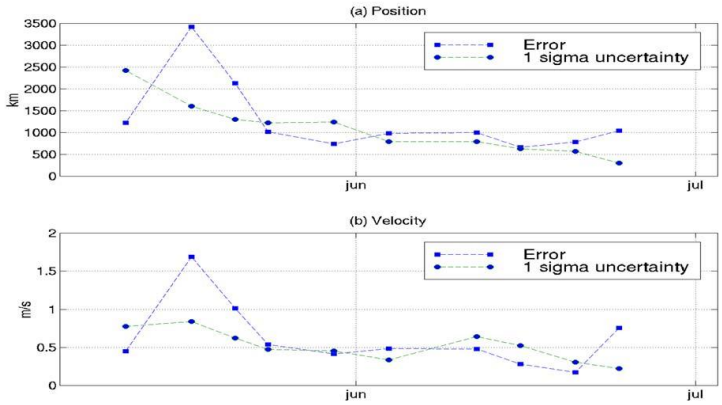


Figure 2.2: Orbital errors in AutoNav calculations in relation to conventional measurements for Deep Space 1 (4).

Table 2.2: Summary of flybys which have used AutoNav until 2012. Adapted from (4).

Mission	Target	Flyby radius (*km*)	Flyby velocity (*km/s*)
Deep Space 1	Borrelly	2171	16.6
STARDUST	Annefrank	3076	7.2
STARDUST	Wild 2	237	6.1
Deep Impact	Tempel 1	500	10.2
EPOXI	Hartley 2	694	12.3
STARDUST NexT	Tempel 1	182	10.9

2.1.2 Stardust

Stardust was launched in 1999 with the objective to collect samples from the tail of comet Wild 2 and return samples to Earth (48). While en route to Wild 2, Stardust flew by asteroid Annefrank and had an opportunity to perform tests before the approach to the final target.

The main difference between this mission and DS1 was that the camera used for the AutoNav had a rotating mirror and was not one of the main star trackers. That camera, however, performed as a star tracker for purposes of line of sight calculation against the star background. The images of asteroid Wild 2 obtained for navigation are seen on **Figure 2.3** and the general spacecraft diagram is presented on **Figure 2.4**.

Figure 2.3 shows the images used by the AutoNav algorithm to calculate the relative position of the spacecraft and the target. The target centroid is detected and the stars in the background are identified. The resulting line of sight is calculated by the position of the target body to the star field in the background.

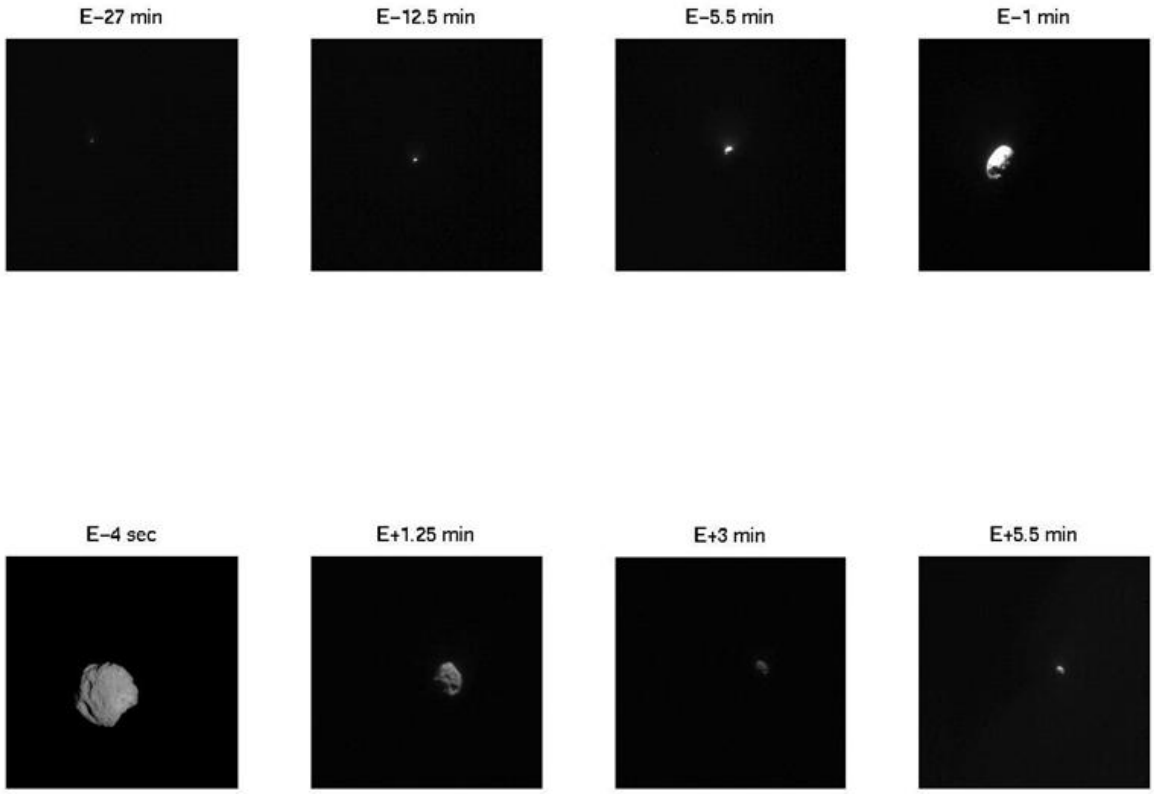


Figure 2.3: Sequence of STARDUST AutoNav images of Wild 2 surround encounter (4).

2.1.3 Hayabusa

The Hayabusa spacecraft (25) was the first to return samples of the surface of an asteroid. It did not use autonomous navigation to get to the asteroid, but used it to approach the surface. The spacecraft had additional hardware in it in order to determine the position in relation to the asteroid surface - notably a LIDAR rangefinder. It relied on a synthetic optical target placed on the surface to be able to cancel the relative velocity

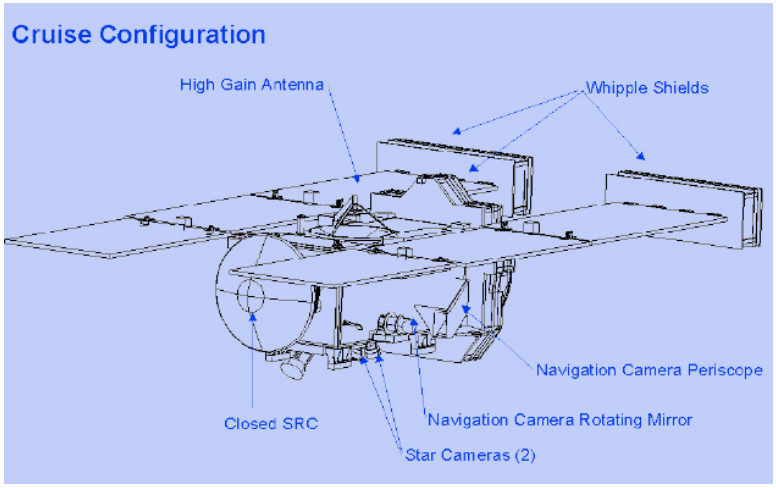


Figure 2.4: STARDUST cruise configuration showing the navigation camera periscope and the two star trackers (48).

and eventually land. The initial intention of the designers was not to have Hayabusa land on the asteroid, but it stayed on the surface for about half an hour.

The reference frame used is shown on **Figure 2.5** (33), and depicts the LIDAR/Laser Rangefinder measurements and the optical target. A picture of the actual landing site is shown on **Figure 2.6**, which was chosen because the rest of the asteroid has loose boulders which make landing more difficult.

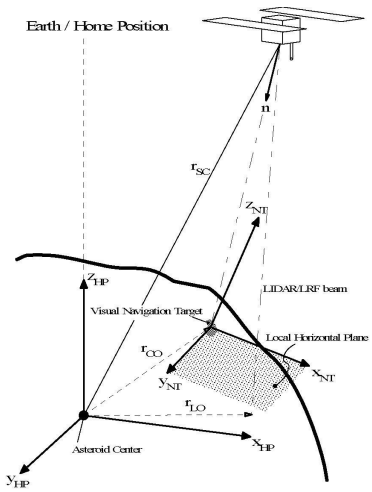


Figure 2.5: Hayabusa geometry of reference frames (33).

2.1.4 Rosetta

The Rosetta mission was launched in 2004 and released a lander to the surface of Comet 67P/ Churyumov-Gerasimenko on August 2014, passing on the way by asteroids 2867 Steins and 24 Lutetia. It was the first mission that accomplished a rendezvous with a comet and deployed a lander unto it (16). It uses the dual onboard navigation cameras and a specific instrument - the OSIRIS science camera - for navigation. The only autonomous function the spacecraft performs, however, is the tracking of asteroids during flyby to maintain the instruments pointed in the right direction (37). The lander descended autonomously with pre-programmed commands.

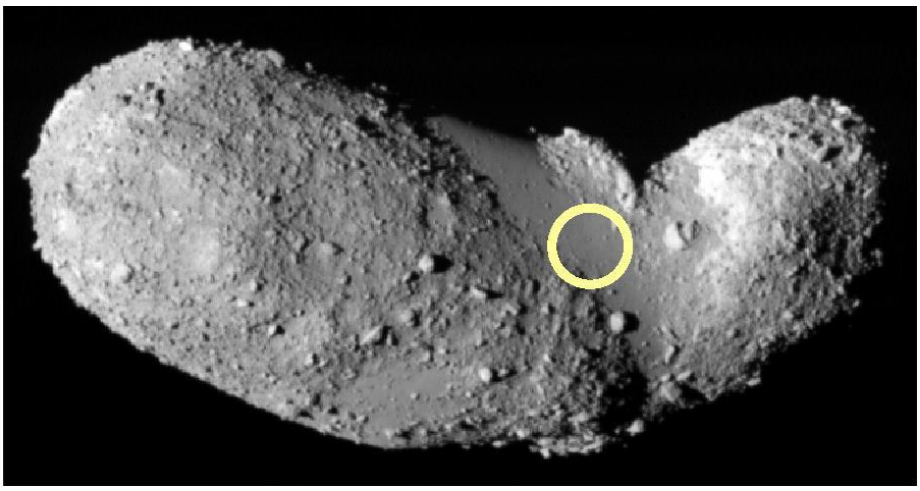


Figure 2.6: Hayabusa landing site: MUSES-Sea (33).

Figure 2.7 shows the asteroid Steins as seen from the navigation camera on August 4th 2008. At the distance at the time, the asteroid is easily mistaken for noise in the camera. Several exposures at different times allow for reliable detection of an object with brightness so close to the camera noise level.

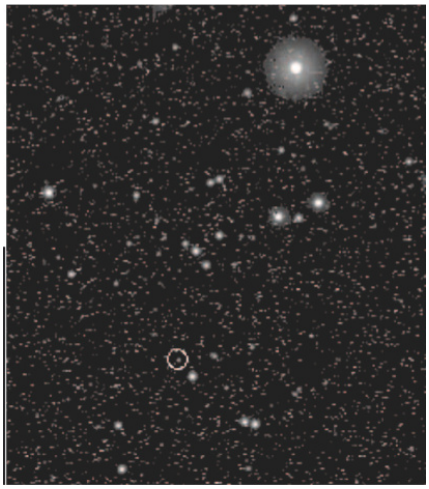


Figure 2.7: Asteroid 2867 Steins as seen from the ROSETTA navigation camera on August 4th 2008. The extended white spots are stars. Almost all other small white spots represent noise from the CCD sensor (pixel dark current) (37).

2.1.5 Deep Impact

The original mission of the Deep Impact probe was to deliver an impactor to the surface of comet Tempel 1 (41). The comet is 7.6 km long at it's largest dimension, which means the navigation system needed to have a positioning error roughly two orders of magnitude smaller than the one used in Deep Space 1.

The spacecraft approached the comet using the same AutoNav technology demonstrated in Deep Space 1 and Stardust, however it used a specific sensor. The Impactor Target Sensor aided by a high precision star tracker and a thruster were able to guide the impactor to a direct hit to the asteroid by autonomously computing and executing three targeting maneuvers (4). The 370 kg "projectile" part of the spacecraft is shown in **Figure 2.8**, although no indication was found of the actual location of the Impactor Target Sensor or of the high precision star tracker. The thruster nozzle is readily identifiable.

The original mission was extended with the objective to perform a flyby of comet Hartley 2 and extrasolar planets (45). The Hartley flyby revealed some difficulties, even though the AutoNav was being used only for pointing the instruments, some images were offset by more than a third of the frame.

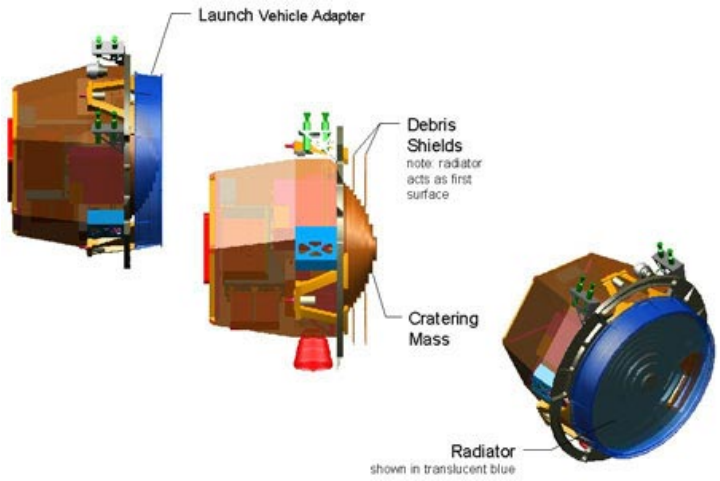


Figure 2.8: Impactor part of the Deep Impact Spacecraft. High precision star tracker, Impactor Target System (ITS) and hydrazine course correction thruster shown but not highlighted. Source: (41).

2.1.6 Hayabusa 2

Hayabusa 2 (24)was launched on December 2014 and is the sucessor of the first Hayabusa (25). It's configuration is mostly the same. The main differences are a new lander, a new antenna design and a device for creating a crater on the asteroid 1999 JU3 which was renamed to 162173 Ryugu as a result of an outreach campaign .

2.1.7 OSIRIS-REx

OSIRIS-REx was launched on September 8th 2016 with the objective to collect a sample from asteroid Bennu (49). It has strict requirements for precise navigation to perform a successful rendezvous, as was the case with Hayabusa and Rosetta. The spacecraft has redundant optical navigation cameras however no reference was found that indicated autonomous navigation capabilities.

It is a recent mission and still there is no mention of autonomous interplanetary navigation. It will operate autonomously only for the landing to collect samples, and for that it will use surface features of the asteroid (38) and aid of other instruments such as Laser Rangefinding (LIDAR).

In September 22nd 2017, OSIRIS-REx will perform an orbital maneuver called "gravity assist", in which it will pass nearby Earth in such a way that it gains speed and alters it's course to the trajectory needed to rendezvous with the asteroid Bennu.

2.2 Algorithms

Reza Karimi's 2012 PhD dissertation "Designing an interplanetary autonomous spacecraft navigation system using visible planets" (29) focuses on the orbit determination, and tests the proposed approach with a spacecraft using Earth as the navigation beacon and positioned halfway between Earth and Mars. No field results are

provided, though, and the error in the experiments converges to an order of magnitude lower than the numbers attained by Deep Space 1.

2.2.1 Star identification

In 2012, Koki Ho published "A survey of algorithms for star identification with low-cost star trackers" (20). The article presents a set of algorithms that work well for small satellites for initial star identification when the attitude is not yet known (the "lost in space" problem). The author also selects parts of the algorithm which he deems to be the best and combines them into a new algorithm and simulates the performance of the studied algorithms and the proposed one.

The author presents an objective definition of low cost start trackers CCDs with 1000×1000 pixels or less which therefore need a field of view of $4^\circ \times 4^\circ$ or narrower to provide the desired angular precision. That definition is very adequate in the scope of that work since geometrical constraints are enforced which make comparing algorithms easier.

The author also presents the concept of "unequal star trackers", that exploits the fact that most spacecraft have more than one star tracker mounted in different directions to have redundancy and increased precision. One of the star trackers can have a wider field of view, which will give less precision but greater ease of initial acquisition, and the other can have a narrower field of view, which will give more precision but may need data from the wider one to be able to acquire an initial solution.

The usage of asymmetric star sensors improves the robustness and precision of the final system. Several algorithms were evaluated for the extraction step and catalog search step. For the extraction step it was concluded that it is best to use a mix of both extended images and combined images. For the catalog search step, it was concluded that the "Group Catalogue" approach is best.

The basic processing flow described by Ho is in **Figure 2.9**. The basic pattern matching algorithm (**Figure 2.10**) is considered robust, but needs a large number of stars to provide a match. According to Ho, a good algorithm for narrow field star tracker uses the angular separation between the stars in it's field of view (**Figure 2.11**). It is also possible to use the identification result of one frame to improve identification on the next frame using the tracking technique (**Figure 2.12**), which increases robustness to errors in the image.

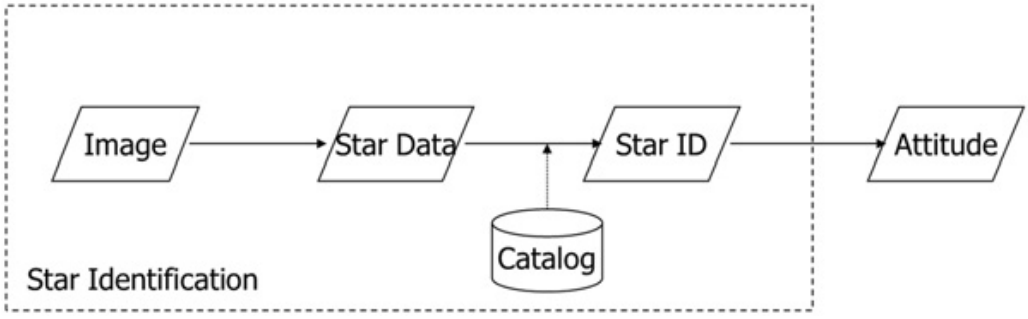


Figure 2.9: Basic flow of the star trackers' processing. The square indicates the processes involved in star identification. (Figure 1 from (20))

Another relevant survay was published in 2009 (62), and made performance comparisons of several high profile algorithms, with data on algorithm computational complexity, memory usage, and resilience to calibration drift by using non-dimensional algorithms.

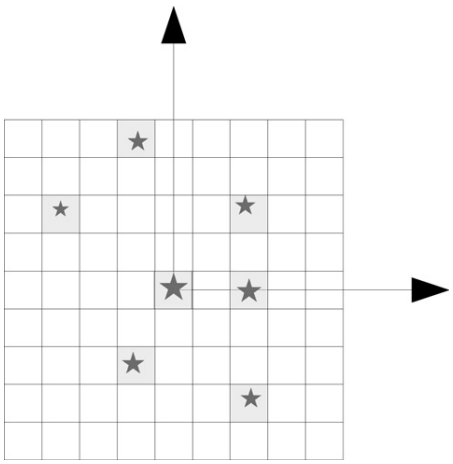


Figure 2.10: Basic pattern matching. Robust, but requires a large number of stars. (Figure 2 from (20))

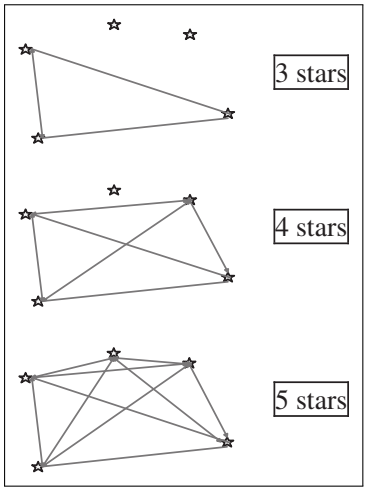


Figure 2.11: Measurement of angular separations between stars (Figure 3 from (20)).

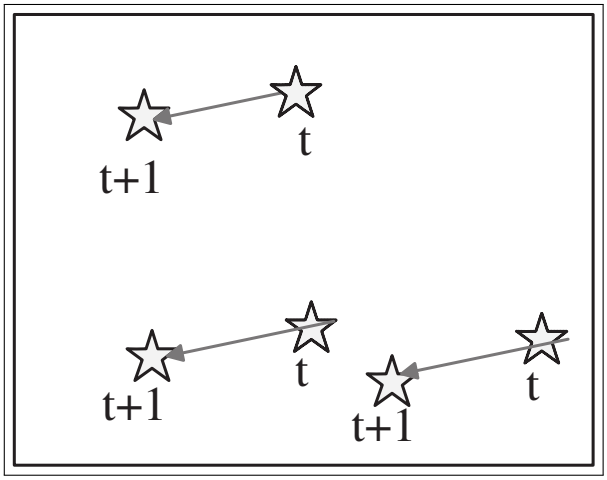


Figure 2.12: Tracking stars from the previous frame to solve the current frame.(Figure 3 from (20))

2.2.2 Accuracy increase by using Kalman filters

Although the use of Kalman Filters is commonplace, it is important to consider that sensor data fusion (either by Kalman Filtering or otherwise) is part of the process to attain state-of-the-art results.

Amir Rad published the article "Optimal Attitude and Position Determination by Integration of INS, Star Tracker, and Horizon Sensor" in 2014 (56) which demonstrates improvement in the measurement of attitude, position and speed obtained by an inertial navigation system (INS), which uses accelerometers and gyroscopes, by integrating measurements from a star tracker and a horizon sensor.

The integration is performed using a Kalman Filter. **Figure 2.14** from the article displays results of simulations showing greatly reduced error when the three sensors are used together.

The report "Autonomous Optical Navigation (AutoNav) DS1 Technology Validation Report" (59) from 2000 details the validation of Deep Space 1's AutoNav, and describes the use of a *"Kalman batch-sequential least-squares filter"* and that *"A typical data arc is about a month long, with four 1-week batches that correspond to the typical one Photo-Op event per week"*. This means that the accuracy obtained in DS1's navigation solutions relies on data that spans at least a month with samples spaced a week apart.

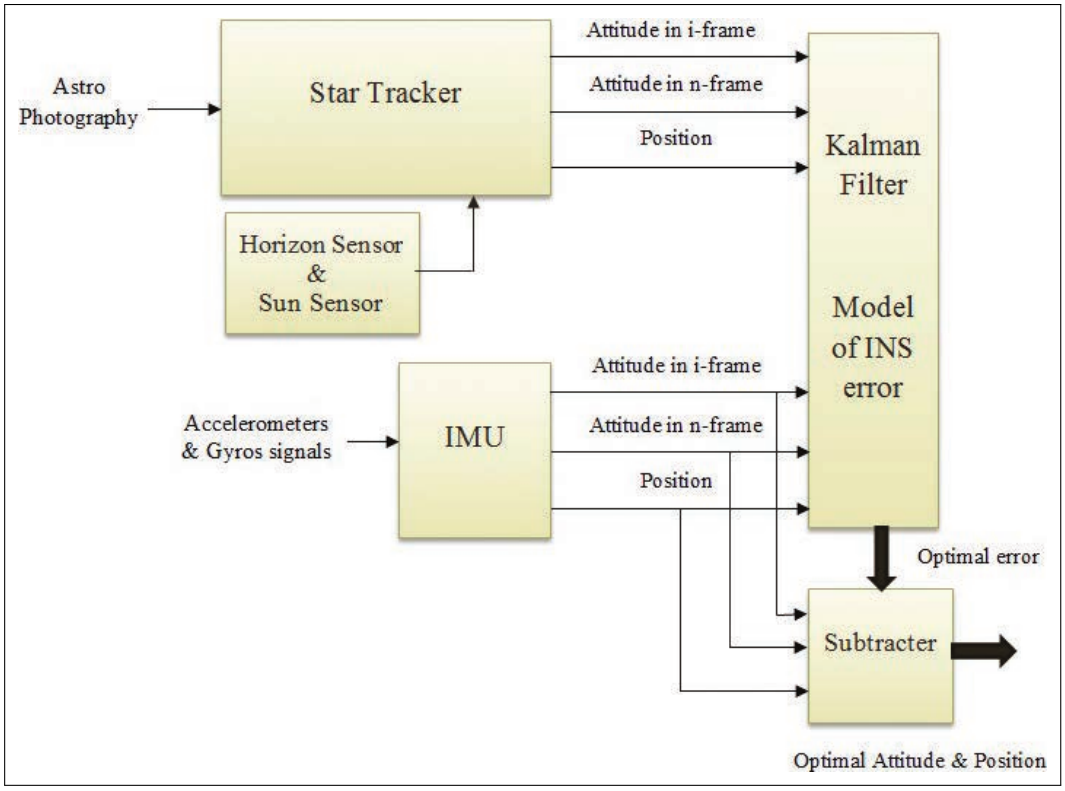


Figure 2.13: Star tracker, horizon sensor and inertial navigation system (INS) integration block diagram (Figure 6 from (56))

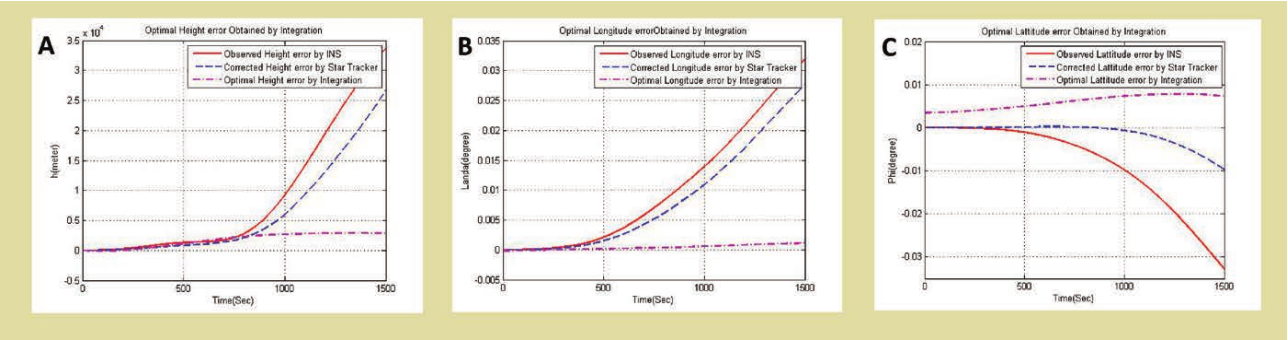


Figure 2.14: Comparison of errors obtained by INS, INS-star tracker combination (without KF), and INS-star tracker integration (with KF) for a ballistic missile simulation. (A) is the height error, (B) is the longitude error and (C) is the latitude error. (Figure 8 from (56))

2.2.3 Star identification without camera calibration

Klaus et al. (31) describes a method for calibrating a camera from a single night sky image, much like the star identification section of this work.

This work presents a different algorithm, and improves upon Klaus et al. (31) by avoiding the brute force approach by means of a precomputed index of star characteristics (*starprints*), by reducing the catalog search space by the selection of “guide stars” without compromising sky coverage, and by demonstrating robust identification in detrimental image conditions.

2.3 State of the art summary

Even though the technology for autonomous navigation is available, as demonstrated from all the AutoNav missions, it is not being widely adopted in new missions when there is no pressing need.

Rosetta’s autonomous tracking of the asteroids during flyby, for example, is a task that is infeasible to perform from the ground on real time due to the communications delay. Without precise locations known "a priori" or a comfortable time window, it would be difficult even to schedule the attitude adjustments ahead of time. Bhaskaran (4) mentions timing errors in the predicted flyby from 12 to 50 seconds.

The state of the art for autonomously planning maneuvers seems to be the AutoNav used in Deep Impact mission. Even so, on the follow-up part of the Deep Impact Mission, the autonomous navigation was used only for pointing the spacecraft in the right direction, not for computing and executing maneuvers, and produced some images with a third of a frame offset from the correct center.

Automatic navigation based on surface features, however, is more advanced in the Hayabusa series, with a successful landing on an asteroid. It is expected that Hayabusa 2 improves on this technology due to the experience in the first mission.

The OSIRIS-REx (49) is a planned asteroid sample retrieval mission from NASA, and there is no mention of autonomous interplanetary navigation but there are redundant cameras for ground based optical navigation. Most current missions (the DAWN mission (43), for example) still relays data to Earth for processing and planning, there is a small number of missions that are capable of autonomous navigation.

The hardware capability is available on recent spacecraft, but there are still risks associated with the use of

the technology, as seen on Deep Space 1, or risks simply associated with the complexity of the mission. Lorenz (38) describes significant difficulties encountered while developing OSIRIS-REx's algorithm for autonomous navigation using surface features of an asteroid not yet properly charted. The trend continues to be to make the spaceborne part of the mission as simple as it can be. This technology is still not being adopted widely for lack of a pressing need, especially in latency considerations. Such a need is expected to arise in the future with the increase in the use of electrical propulsion, the increase in the number of spacecraft and the adoption of ever-stricter mission requirements.

Chapter 3

Methodology

This chapter describes the hardware utilized, provides insights about the characteristics of the star catalog dataset and describes the processing flow to obtain a position solution from images of the sky.

Figure 3.1 illustrates the three steps involved in the star identification algorithm. The results of those steps are then used to calculate the relative angular position of unknown bodies, match them to planets and extract a location.

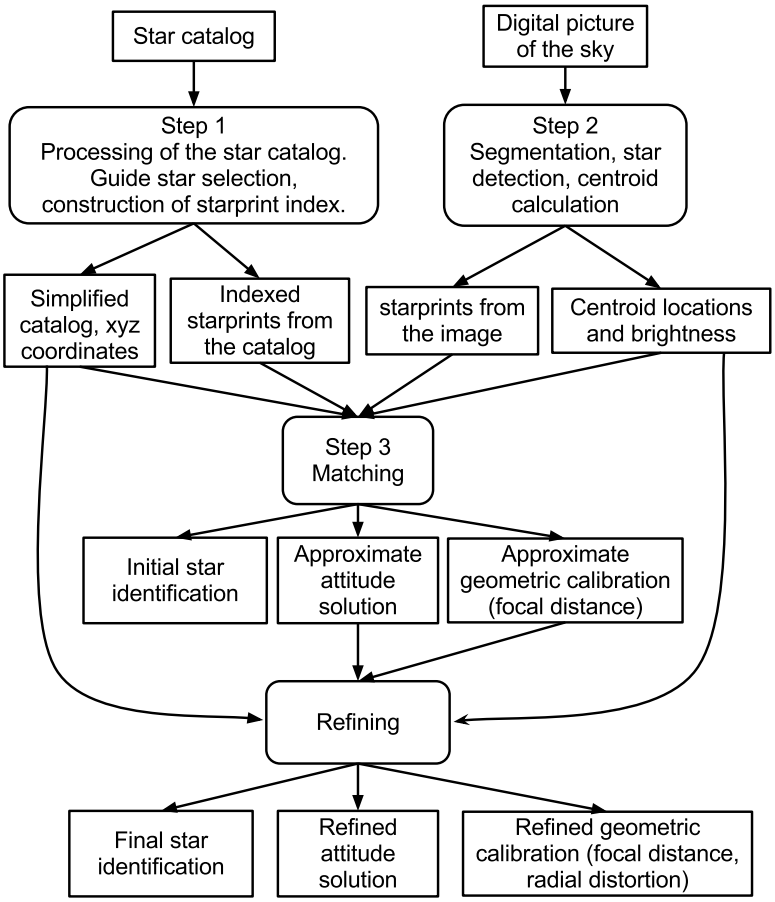


Figure 3.1: Outline of the star identification algorithm. Square cornered cells represent input and output data, round cornered cells represent processing steps.

3.1 Hardware

The triangulation experiments were performed using low cost cameras - a Nikon D5000 with a simple manual lens in it and a Raspberry Pi NoIR camera. An older generation Canon compact camera was used during development of the *Starprint* algorithm for identifying stars in pictures of uncalibrated cameras. Several pictures used for testing the recognition part of the algorithm (**Section 4.1**) were taken from other sources and have a wide variety of hardware associated with them.

The Nikon D5000 and the Raspberry Pi camera are controlled by a Raspberry Pi V2 minicomputer powered by a TP-Link 10400mAh Power Bank. The Python *picamera* library was used to control the NoIR camera and the *gphoto2* library was used to control the Nikon D5000. **Figure 3.2** displays the arrangement ready to start an imaging session. A Lenovo Thinkpad notebook was used to start the capture, program the raspberry pi and process the data afterwards. Processing could have been performed in the Raspberry Pi V2 minicomputer, but the desktop is faster and it was helpful to have debug information and access to auxiliary software during development. Also, the Raspberry Pi internal memory would not be able to keep all the intermediary data for posterior analysis, it would need to be erased as more data came in.

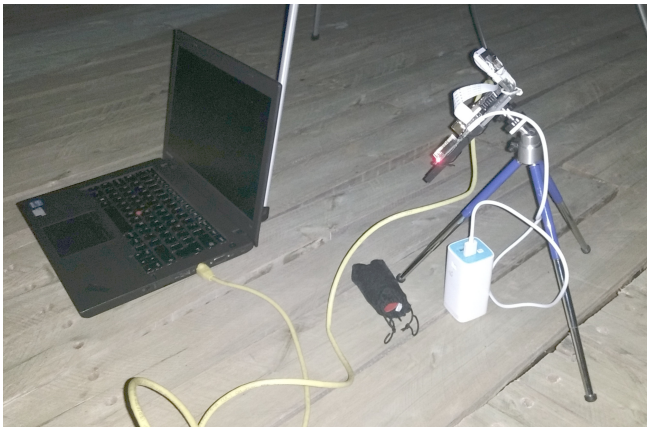


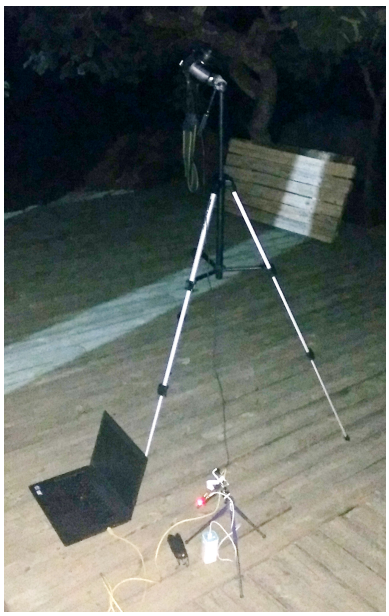
Figure 3.2: Lenovo Thinkpad notebook used to start the capture. The Nikon D5000 and the Raspberry Pi camera are controlled by a Raspberry Pi V2 minicomputer powered by a TP-Link 10400mAh Power Bank.

3.1.1 DSLR Camera

One of the cameras used for triangulation was a Nikon D5000 (51) with a Nikon 50mm D f/1.8 manual lens. The camera and lens were bought used for under USD 500 in Brasília-Brazil in 2015. A new D5300 (current model) and a new 50mm D f/1.8 are available for under USD 650 in the U.S. online as of 2017-07-09. **Figure 3.3** displays the camera mounted on a tripod ready to start taking pictures with the lens cap on for dark frame calibration.

According to the camera specs (51), the camera has 12.3 million effective pixels, the sensor is a CMOS sensor measuring 23.6 x 15.8 mm and the image size is 4,288 x 2,848 pixels. It is capable of taking pictures in "raw" mode, which provides the actual values read by the A.D. converter for each pixel without processing.

From the nominal specifications, each pixel measures 5.5×5.5 μm and a 50mm lens would make each pixel cover 0.11 milliradians, or $0.0063°$.



(a) Camera mounted on a standard tripod and connected to the Raspberry Pi V2.

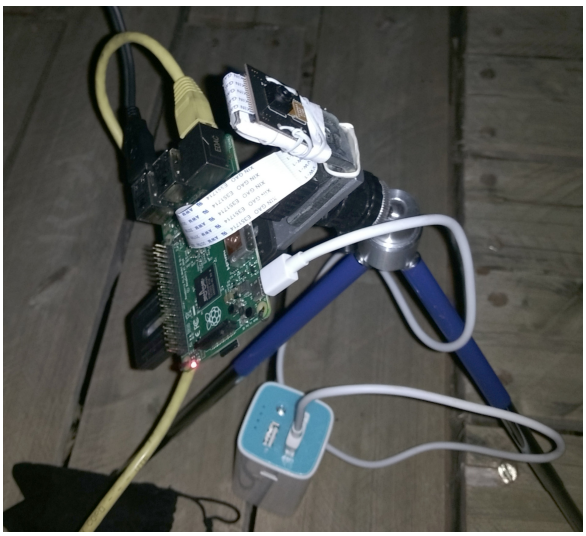
(b) Close up of the camera showing the 50mm D f/1.8 Nikon lens with the cap attached.

Figure 3.3: Nikon D5000 mounted on a tripod ready to start dark frame calibration.

3.1.2 Raspberry Pi Camera Module

The other camera used for the triangulation tests is a Raspberry Pi NoIR 5MP camera (57). It was bought new for USD 28.61 alongside a Raspberry Pi V2 minicomputer kit from CanaKit with a 2.5A power supply for USD 52.99 in the US in 2015.

Since the camera does not have a lens cap, a shroud was improvised with an opaque plastic tube inside a black felt bag. The shroud is used for dark frame calibration.



(a) Camera exposed to the sky.

(b) Camera covered by shroud.

Figure 3.4: Assembly of the Raspberry PI V2 with the NoIR camera on a small tripod, connected to a TP-Link 10400mAh Power Bank, the Nikon D5000 via USB and the notebook via Ethernet. The shroud is used to perform dark frame calibration.

The camera flashes a red light by default when taking pictures. The light was disabled in the configuration to reduce light pollution in the pictures.

The official specs of the camera module (57) are as follows:

```
Camera Module with no Infrared Filter
Sensor type: OmniVision OV5647 Color CMOS QSXGA (5-megapixel)
Sensor size: 3.67 x 2.74 mm
Pixel Count: 2592 x 1944
Pixel Size: 1.4 x 1.4 um
Lens: f=3.6 mm, f/2.9
Angle of View: 54 x 41 degrees
Field of View: 2.0 x 1.33 m at 2 m
Full-frame SLR lens equivalent: 35 mm
Fixed Focus: 1 m to infinity
Video: 1080p at 30 fps with codec H.264 (AVC)
Up to 90 fps Video at VGA
Board size: 25 x 24 mm (not including flex cable)
```

It is also able to generate "raw" format images when using the Python *picamera* library, and it produces a 10 bit per pixel bayer image in that format.

From the nominal specifications, each pixel measures 1.4×1.4 μm and a 3.6mm lens would make each pixel cover 0.39 milliradians, or $0.022°$.

3.1.3 Older generation compact camera

An older generation Canon PowerShot S1 IS (**Figure 3.5**) was used during initial development of the *Starprint* algorithm for identifying stars in pictures of uncalibrated cameras. It was manufactured in 2004 or 2005. It has a CCD sensor while the Nikon D5000 and the Raspberry Pi camera have CMOS sensors.

Figure 3.5: Canon PowerShot S1 IS with 3.2 megapixel CCD (15)

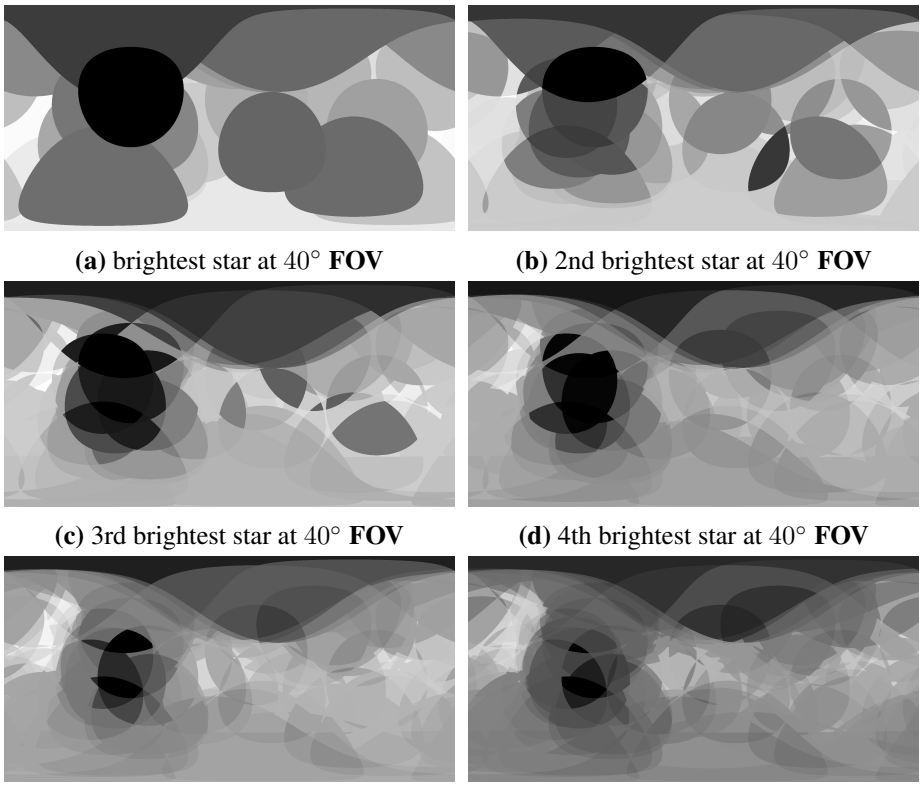
The camera has a 3.2 megapixel CCD measuring 5.27×3.96 mm (15). The resulting image files have 2048×1536 pixels. With the zoom set to a minimum, the JPG Exif information embedded in the pictures gives a focal distance of 5.8mm

For taking the pictures from the sky an exposure time of 15 seconds was necessary, much longer than the sub-second time used in the other two. This camera does not have a "raw" picture mode, only processed JPG files are available.

3.2 The star catalog dataset

The distribution of stars in the sky is not homogeneous. Characteristics of that distribution can be used to estimate what should be the camera's detection threshold and how many stars should be included in the catalog as a function of the camera's field of view.

The Hipparcos catalog (53) was used in this section. Star identification tests in **Section 4.1** used a smaller catalog (Bright Star Catalogue (21)) which also contains star names and makes interpreting the results easier.



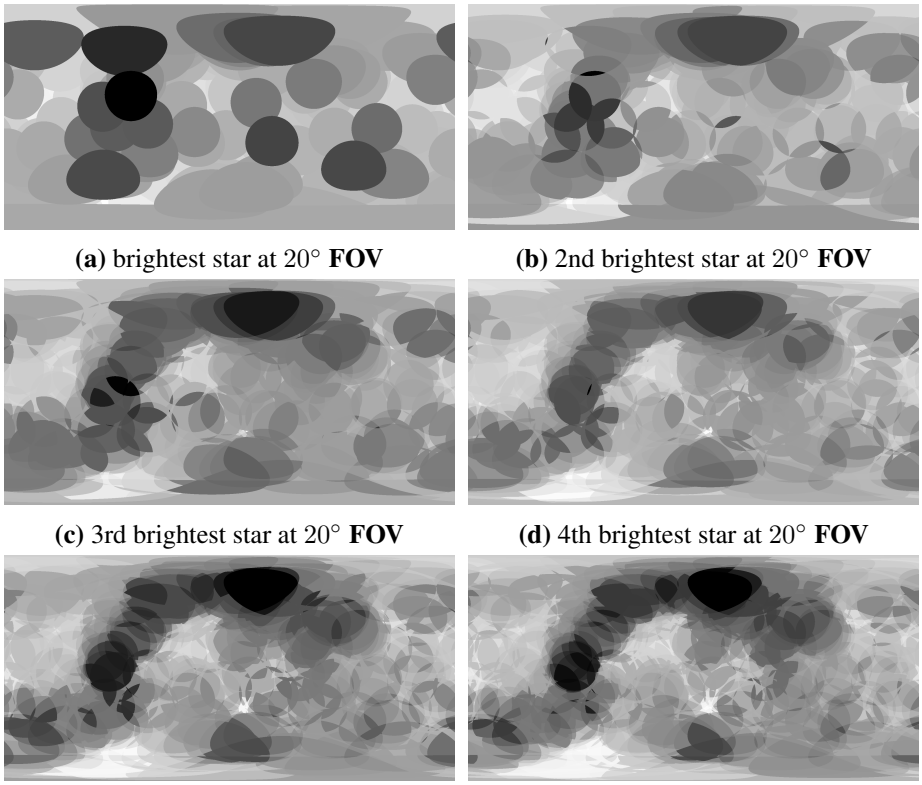
(a) brightest star at 40° **FOV** **(b)** 2nd brightest star at 40° **FOV**

(c) 3rd brightest star at 40° **FOV** **(d)** 4th brightest star at 40° **FOV**

(e) 5th brightest star at 40° **FOV** **(f)** 6th brightest star at 40° **FOV**

Figure 3.6: Bright star availability for a camera with a 40° **circular full field of view** centered at each pixel.

The image sequences in **Figures 3.6, 3.7, 3.8 and 3.9** display the availability of bright stars in the night sky. For better visualization, the full value range of each simulation was stretched, mapped to gray values from 0 to 255 and colors are inverted. Darker pixels represent locations with brighter stars, the pixel representing the field of view which has the brightest star of the entire image is fully black and the pixel representing the field of view in which the brightest star seen is dimmest than the brightest star seen from all the other locations is fully white.



(a) brightest star at 20° **FOV**

(b) 2nd brightest star at 20° **FOV**

(c) 3rd brightest star at 20° **FOV**

(d) 4th brightest star at 20° **FOV**

(e) 5th brightest star at 20° **FOV**

(f) 6th brightest star at 20° **FOV**

Figure 3.7: Bright star availability for a camera with a 20° **circular full field of view** centered at each pixel.

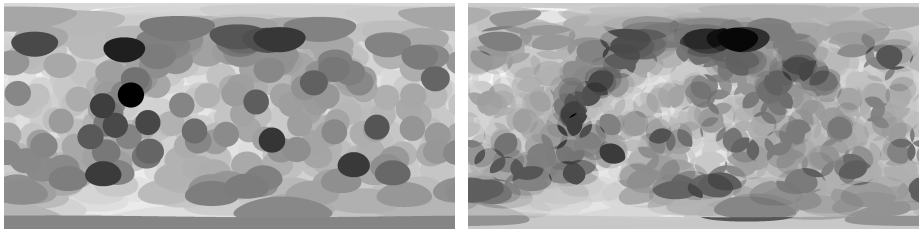
The images are the result of a simulation of thousands of pictures taken with a camera with a circular field of view. The field of view is fixed for each of the four sequences at $40°$, $20°$, $10°$ and $2.5°$, respectively. Each image in the sequence visually represents the brightness of the brightest star in the field of view centered at each pixel, then the brightness of the second star in the same field of view, then the third brightest star and so on. If one takes **Figure 3.6c**, for example, each pixel represents the brightness of the third brightest star in the field of view of a camera centered at that pixel with a full circular field of view of $40°$ (displays every star within a $20°$ angle of the line of sight).

The coordinates of the pixels range from $-180°$ to $180°$ of right ascension in a linear scale from left to right, and from $-90°$ to $90°$ of declination in a linear scale from bottom to top. Right ascension and declination are used according to the International Celestial Reference System (ICRS) (1).

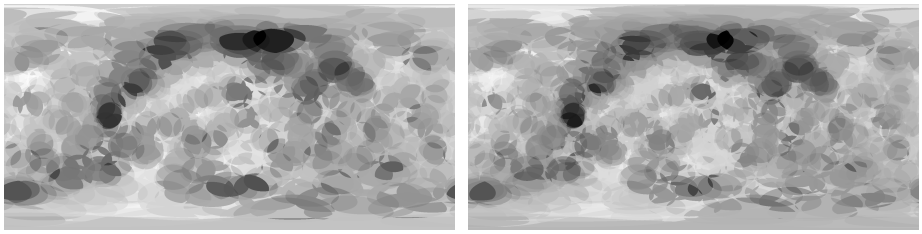
Table 3.1 gives a performance criteria based on the simulations. For a given field of view and number of required stars, the table gives the minimum detection threshold of the camera to guarantee that the required number of stars can be detected regardless of where the camera is pointing at the sky. It contains the value of the pixel corresponding to the dimmest brightness of **Figures 3.6, 3.7, 3.8 and 3.9** as well as simulation results for other fields of view. The **x** values represent "gaps" - there are circular regions with diameter ϕ=1.25° in which there are no more than two stars recorded in the Hipparcos catalog.

Figure 3.10 represents **Table 3.1** as a logarithmic plot. The curves for **n**=3 to **n**=6 lack data in the first point, which are the Hipparcos "gaps". A more complete catalog would be necessary to make those measurements.

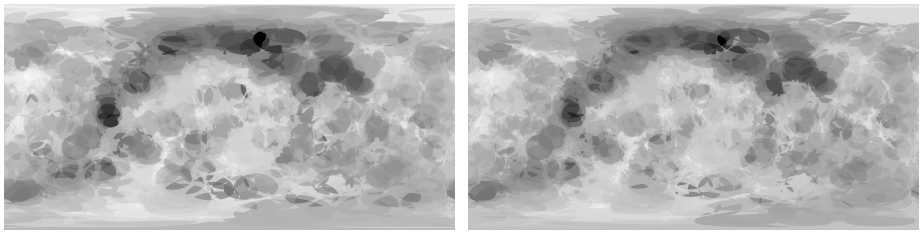
This study uses only the star's visual magnitude in the catalog. Depending on the type of sensor utilized to image the sky, the relative brightness values may change due to different sensitivity across the spectrum. A star with more infrared light in it's spectrum may, for example, become brighter in an infrared sensitive camera



(a) brightest star at 10° **FOV** **(b)** 2nd brightest star at 10° **FOV**

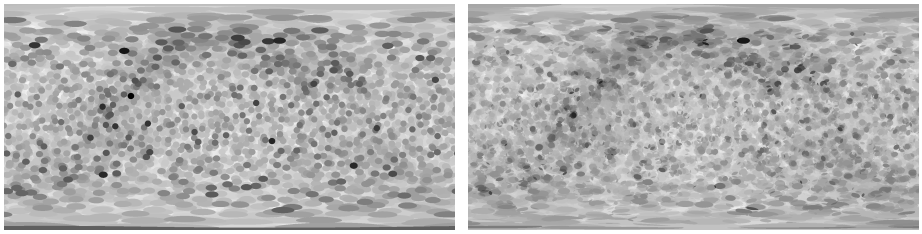


(c) 3rd brightest star at 10° **FOV** **(d)** 4th brightest star at 10° **FOV**

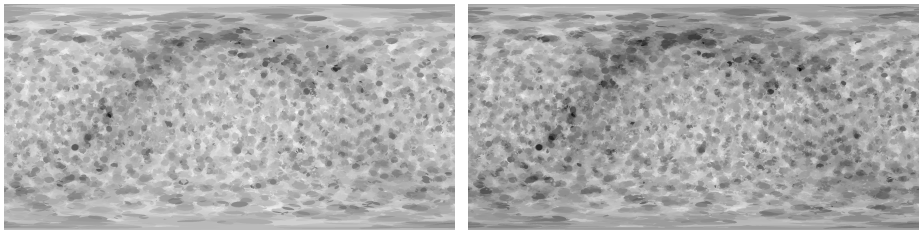


(e) 5th brightest star at 10° **FOV** **(f)** 6th brightest star at 10° **FOV**

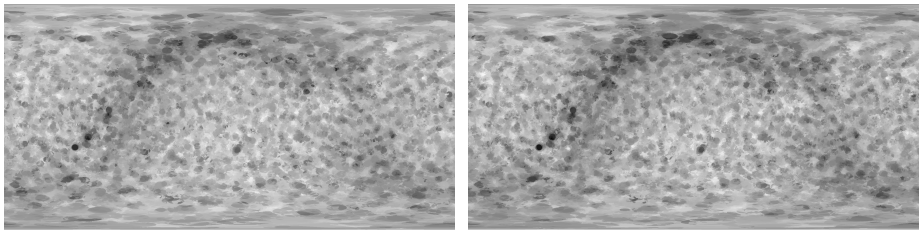
Figure 3.8: Bright star availability for a camera with a 10° **circular full field of view** centered at each pixel.



(a) brightest star at 2.5° **FOV** **(b)** 2nd brightest star at 2.5° **FOV**



(c) 3rd brightest star at 2.5° **FOV** **(d)** 4th brightest star at 2.5° **FOV**



(e) 5th brightest star at 2.5° **FOV** **(f)** 6th brightest star at 2.5° **FOV**

Figure 3.9: Bright star availability for a camera with a 2.5° **circular full field of view** centered at each pixel.

Table 3.1: Necessary magnitude detection performance for a star camera with a circular field of view of diameter ϕ to guarantee the detection of at least **n** stars in every region of the sky.

	ϕ=1.25°	ϕ=2.5°	ϕ=5°	ϕ=10°	ϕ=20°	ϕ=30°	ϕ=40°
n=1	9.4	7.7	6.3	5.1	3.7	2.9	2.1
n=2	10.4	8.0	6.6	5.3	4.0	3.5	2.4
n=3	x	8.1	6.7	5.3	4.2	3.5	2.5
n=4	x	8.4	6.9	5.5	4.3	3.6	2.9
n=5	x	8.4	6.9	5.5	4.3	3.6	3.0
n=6	x	8.5	7.1	5.7	4.4	3.6	3.3

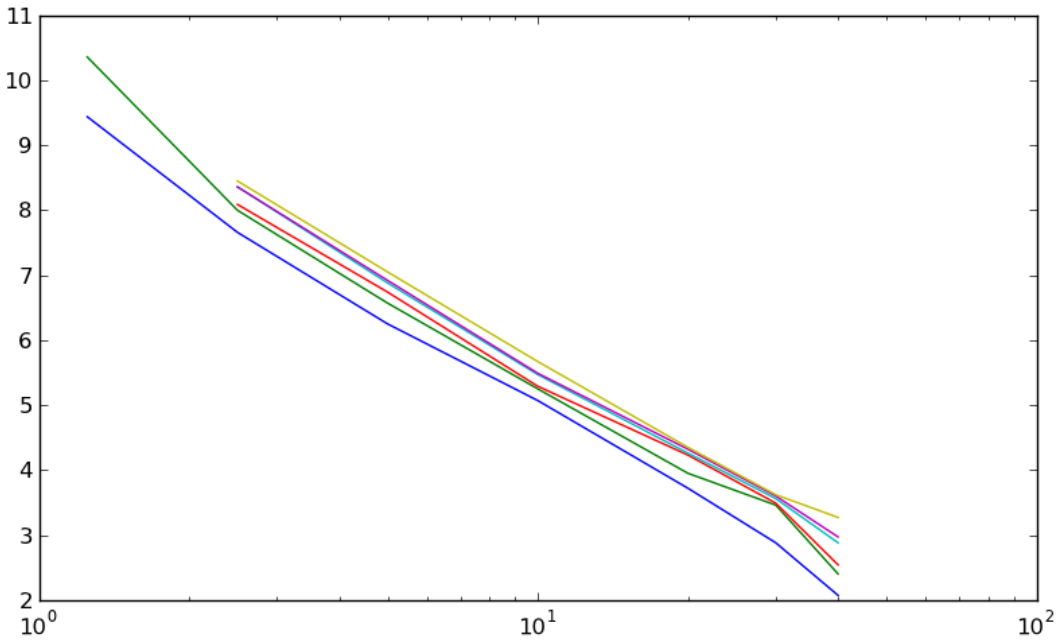


Figure 3.10: Necessary detection performance (y axis) as a function of field of view diameter (x axis). Logarithmic plot of **Table 3.1** with colors blue for **n=1**, green for **n=2**, red for **n=3**, light blue for **n=4**, pink for **n=5** and gold for **n=6**

than another star, which is seen to the human eye as the first one. The visual data here can be used as reference, but precise results for specific imaging devices should consider calibrated magnitudes.

3.3 The *starprint*

The *starprint* is a set of features of a star that is dependent on the relative brightness and spatial disposition of it's neighbors. It is designed to produce similar, easy to match results when calculated using a catalog or calculated from an image.

The *starprint* of a star S is calculated by picking its brightest neighbors in every disk slice region around S and computing the angle and distance ratios of all the combinations of two neighbors with S. The *starprint* is stored in a table containing the angles, distance ratios and star identifiers. Each row of the table is a *starprint* element. **Algorithm 1** details the calculation for a star in the catalog. Calculation for a star in an image requires minor modifications.

To better understand the concept, an analogy is made of the starprint area with a pizza. Star S can be compared to a large olive at the center of the pizza and we are going to chose other olives on the pizza to do our calculation. We define an angle θ such that every slice we cut from the pizza has to have an angle θ and, before cutting any slices, we remove all olives from a small circle at the center of the pizza (diameter $/8$). The rule is that an olive other than S is chosen if, and only if, it is possible to cut a pizza slice in which it is the largest olive in the slice.

The disk slice tested in the neighborhood of S goes from radius $R/8$ to R and covers an angle of θ. It is impossible to test every possible slice, but markers are placed just before and just after a star enters or leaves the limits of a slice and tests are performed only at the markers. The parameters $\theta = \pi/2$ radians ($90°$) and $R = \pi/9$ radians ($20°$) where used in the experiments.

Algorithm 1 *Starprint* calculation of star S using angle θ and radius R from a star catalog.

Acquire a star catalog
Define a radius R
Define "pie slice" angle θ
Choose a star S from the catalog
Project stars within range R of S simulating a picture taken by a pinhole camera with unity focal distance aimed at S
Create array N with the projected coordinates in polar notation
Sort N in descending order of magnitude (brightest first)
Create an empty floating point array T
for n in N **do** //populates T with the list of angles to test
    $T.append(n.angle)$
    $T.append((n.angle + \theta) mod (2\pi))$
end for
Create an empty star array P
for t in T **do** //get the brightest neighbor in each θ-sized slice:
    for n in N **do**
        $a \leftarrow (n.angle - t) mod (2\pi)$
        $r \leftarrow n.radius$
        if $a < \theta$ and $r < R$ and $r > R/8$ **then**
            //if it is in the slice, add to the list
            if not n in P **then**
                $P.append(n)$
            end if
            //the brightest neighbor is the first on the list
            $break$ //skip the others, go to next angle
        end if
    end for
end for
Create empty table $result$ with 5 columns
for i in $0 \rightarrow len(P) - 1$ **do**
    for j in $(0 \rightarrow len(P) - 1, j \neq i)$ **do**
        $\phi \leftarrow (P[j].angle - P[i].angle) mod (2\pi)$
        $ratio \leftarrow log_{10}(P[j].radius / P[i].radius)$
        $result.append(S.id, P[i].id, P[j].id, \phi, ratio)$
    end for
end for

Matching is performed by calculating *starprints* from the image and searching matching elements in a table concatenating all the *starprints* from selected stars in the catalog. This table is sorted by ratio to speed up

searches. Tolerances are established for angle and distance ratio so that maximum expected lens distortion and off-axis starprint distortion are accommodated.

This method of *starprint* calculation was designed so that the results of the calculation are independent of scale and rotation, suffer limited change from translation (off-axis *starprints* are not equal but are similar to on-axis starprints) and are resistant to false stars or occlusions, since the calculations are repeated starting with every one of the selected neighbors. Extra bodies or missing bodies would cause some extra elements to appear or be suppressed, but the rest of the *starprint* would remain the same.

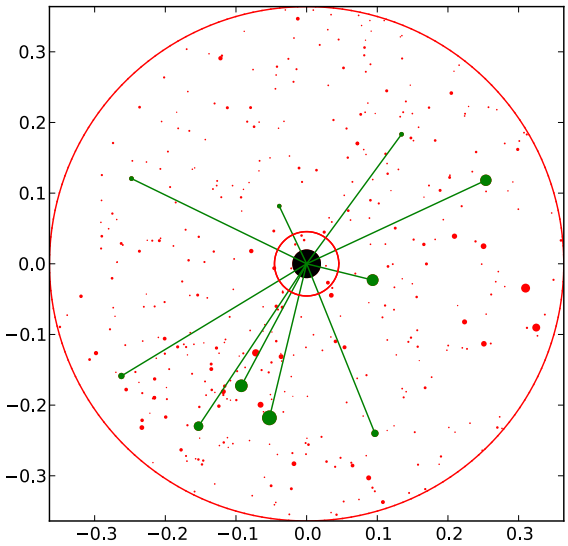


Figure 3.11: *Starprint* diagrams for Sirius. Circles were drawn at a distance $R = \pi/9$ radians (20^o) and $R/8$. The stars in P are connected to S (see **Algorithm 1**). Axes depict coordinates of the projected stars as seen from a pinhole camera with unity focal distance.

S	Neighbor 1	Neighbor 2	ϕ	$ratio$
2480	1995	2218	0.502	-0.090
2480	1995	2562	1.578	-0.491
2480	1995	2958	2.253	-0.007
...	...	...	...	...
2480	2218	2562	1.075	-0.400
2480	2218	2958	1.750	0.084
2480	2218	3033	2.748	0.130
...	...	...	...	...

Table 3.2: Part of the resulting starprint from **Figure 3.11**. The identifiers can be traced back to catalog stars: Alp CMa (2480), Kap Ori (1995), Gam Mon (2218), Bet CMa (2285), etc. The full table has 90 rows.

The choice of neighbors is performed in such a way that these star triads tend to be the same whether the calculations were performed on centroids extracted from an image or on simulated sky scenes from the catalog, and a star close to the image border would produce a partial fingerprint with some elements being the same as the full fingerprint. **Figure 3.11** displays a diagram of one such selection and **Table 3.2** displays the corresponding data generated for that starprint.

3.4 Star Catalogue processing

3.4.1 Catalogue cleanup

Two catalogs were used: The Bright Star Catalog (BSC) (21), with 9110 entries, and the Hipparcos Catalog (HYP) (53), with 118218 entries. In the case of the Bright Star Catalog, there are several items that were later found to not be stars, but galaxies or other objects. Some binary systems were split into two entries with the same position in the sky to denote two different stars. Binary stars were grouped into single entries and non-star entries were discarded. BSC was reduced from 9110 entries to 9082 after the preliminary cleanup. **Figure 3.12** gives an example of such entries in the BSC as a snipped of the plain text version of the catalog.

```
89          CD-31  138   1909192495                      001812.3-313526002312.6-310210358.89-82.70 6.55   0.00
90          BD+37   58   1967 53860    I:W       R And   001844.8+380125002402.0+383438117.07-23.98 7.39  +1.97
91          BD+51   62   1976 21366         328    155   001852.2+512757002415.6+520112118.68-10.63 5.57  -0.12
92 NOVA 1572                                    B Cas
93 12    CasBD+61   69   2011 11172                      001916.2+611637002447.5+614952119.79-00.88 5.40   0.00
94          BD-03   49   20231287432021                  001923.0-024620002429.7-021309107.27-64.27 6.07  +1.22
95 47    Tuc
```

Figure 3.12: First 100 characters of lines 89 to 95 of the Bright Star Catalog plain text version (21). Records number 92 and 95 were thought to be stars when the catalog was first made. Record 92 was later found to be a supernova and record 95 a globular cluster.

3.4.2 Calculate *starprints* of reference stars

This is "Step 1" from the diagram in **Figure 3.1**. The objective here is to generate a list of *starprints* (**Section 3.3**) of stars picked from a star catalog. "Step 2" generates the same *starprints* from an image and "Step 3" matches the *starprint* components from "Step 1" and "Step 2" to obtain a star identification.

Before calculating the *starprints*, a set of reference stars is chosen from the catalog to give coverage of all the sky and at the same time reduce the number of necessary calculations to find a match. A lower limit for the field of view is chosen to keep the number of stars to a minimum. If the chosen limit is 10° for example, for every picture with a circular field of view of radius 10° one of the stars in the scene is going to be a reference star and will have had it's starprint calculated from the catalog.

If one tries to identify stars in an image with a smaller field of view than the one used to choose the reference stars, there are no guarantees that a matching *starprint* component will exist in the scene. On the other hand, choosing a very narrow field of view to generate the reference set will increase the search space and will require more sensitive equipment to acquire the images. See **Figure 3.10**.

The reference stars are chosen by simulating a large number of views of the sky with a slightly smaller field of view ($\theta - \delta$) than the minimum field of view θ chosen. The sky is sampled at intervals of δ, with δ being much smaller than θ. For each of these simulated views, the brightest star in the scene is added to the reference set. This guarantees that any scene with a field of view equal or larger to θ will contain one of the simulated views and therefore at least one of the reference stars.

Starprints are then calculated for each of the reference stars and the components of all the calculated starprints are placed in a table with five columns: the catalog index of the primary star, the catalog index of the secondary star, the catalog index of the tertiary star, the angle between the lines connecting the primary with the secondary and the primary with the tertiary and the ratio of the distances between the tertiary and the primary and the secondary and the primary. The calculation of the *starprint* is described in **Section 3.3**.

Figure 3.13 displays a rendering of the stars present in the Bright Star Catalogue (21) overlayed with a

grid of the right ascension and declination coordinates. Right ascension grid lines are roughly parallel and are labeled in gray, declination lines are labeled in black and converge at the celestial north. Polaris, the North Star, is visible near the declination lines convergence point and the Ursa Major constellation is visible to the left. Brighter stars are rendered as larger dots. The same scene is also displayed on **Figure 3.14**, but without the grid lines. **Figure 3.15** displays only the stars from BSC selected as reference stars using a 4° radius circular window (θ=4°, σ=0.4°).

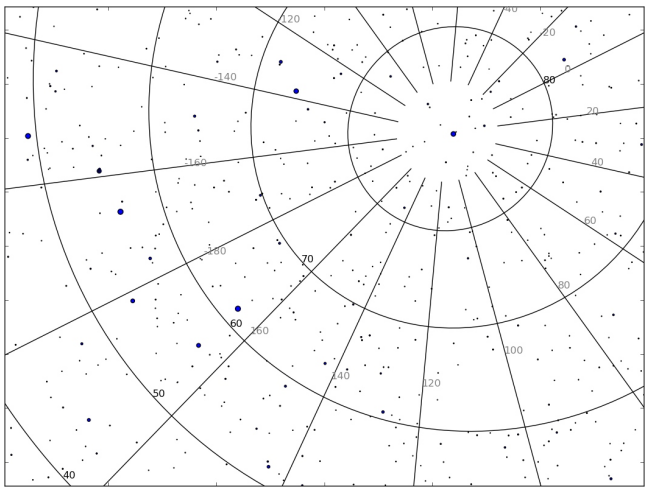


Figure 3.13: Stars in the BSC (21). Right ascension and declination lines shown.

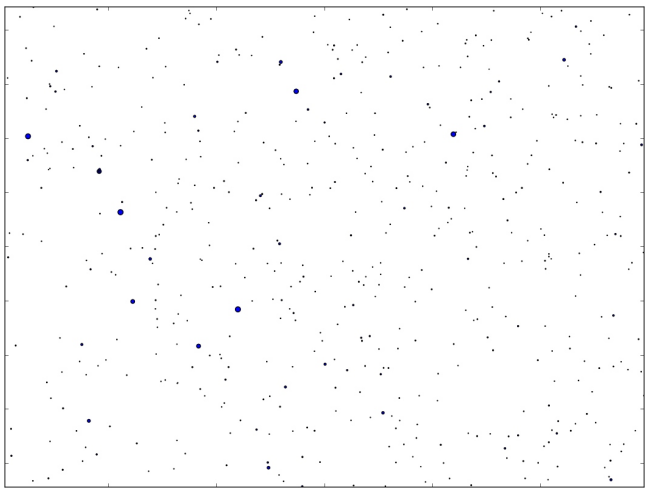


Figure 3.14: Stars in the BSC (21) of the area displayed in **Figure 3.13** without the right ascension and declination grid.

The reference star selection on the Hipparcos catalog was performed with a field of view window with radius $\theta = \frac{4\pi}{180}\text{rad} = 4^\circ$ with one star per window and $\delta = \frac{\theta}{10}$ (a tenth of the window size), which yielded 2768 reference stars

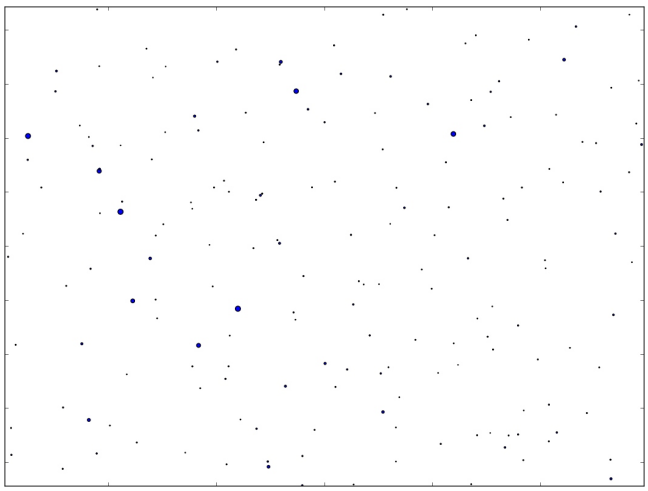


Figure 3.15: Reference stars chosen from the Bright Star Catalogue (21) using a $4°$ radius circular window (θ=$4°$, σ=$0.4°$)The region is the same as the one in **Figure 3.13** and **Figure 3.14**.

3.4.3 Coordinate change and epoch adjustment

The effect which causes the relative position of stars to change due to observer position is called "parallax", and the change in relative position due to the movement of the stars themselves is called "proper motion".

Parallax is not considered in this algorithm. For comparison, Proxima Centauri has the largest parallax in the Hipparcos Catalog at 0.772 arc-seconds. The mean parallax in the coverage set is two orders of magnitude less than the displacement due to proper motion in the 26 years since the Hipparcos catalog epoch (Hipparcos positions are given at year 1991.25 - year 1991 plus a quarter of a year (53)) and almost an order of magnitude less than the displacement due to proper motion in a year.

The catalog is then converted from Right Ascension and Declination angular positions to cartesian x, y and z coordinates on the surface of a unity sphere, and the proper motions are converted to vectors in cartesian x, y and z coordinates. **Equation set 3.1** shows the coordinate conversion and the position updates due to proper motion. Since the coordinates need to stay on the surface of the unit sphere, the last vector calculation needs to be normalized.

Equations for transforming the catalog coordinates and updating to the current date:

$$x_0 = np.cos(RA)np.cos(DE) \tag{3.1a}$$

$$y_0 = np.sin(RA)np.cos(DE) \tag{3.1b}$$

$$z_0 = np.sin(DE) \tag{3.1c}$$

$$\vec{p_0} = (x_0, y_0, z_0) \tag{3.1d}$$

$$v_x = -sin(RA)cos(DE)pmRA - cos(RA)sin(DE)pmDE \tag{3.1e}$$

$$v_y = +cos(RA)cos(DE)pmRA - sin(RA)sin(DE)pmDE \tag{3.1f}$$

$$v_z = 0pmRA + cos(DE)pmDE \tag{3.1g}$$

$$\vec{v} = (v_x, v_y, v_z) \tag{3.1h}$$

$$dt = (currentdateinyears - 1991.25) \tag{3.1i}$$

$$\vec{p} = \frac{\vec{p_0} + \vec{v}dt}{\|\vec{p_0} + \vec{v}dt\|} \tag{3.1j}$$

After processing the Hipparcos Catalog, a "coverage set" is generated, which is the set of reference stars, with the following characteristics:

- Number of years since catalog epoch: 26.28 (2017-07-11)
- Number of stars in the coverage set: 2768
- Magnitude of the faintest star in the coverage set: 6.86
- Magnitude of the brightest star in the coverage set: -1.09
- Mean proper motion of stars in the coverage set: 0.095 arcseconds/year.
- Highest proper motion found in the coverage set: 4.08 arcseconds/year.
- Mean displacement due to proper motion in the coverage set: 2.44 arcseconds.
- Highest displacement due to proper motion in the coverage set: 105 arcseconds.
- Mean parallax of stars in the coverage set: 0.018 arcseconds.
- Highest parallax found in the coverage set: 0.75 arcseconds.

3.5 Acquisition

Acquisition of sky images for triangulation was made using a Nikon D5000 camera and a Raspberry Pi NoIR camera module. Images for testing the star identification algorithm were also acquired using the Canon S1 IS (see **Section 3.1**) and others, sometimes unknown, were used to take the test pictures that were obtained from the internet.

3.5.1 Planning

The software Stellarium (7) was used to plan the observation sessions. It is important that at least two planets are visible at the same time in the sky and none of them is too low on the horizon so that atmospheric interference is minimized and the camera is able to capture a good star neighborhood around the planet. The planets can not be too near the moon too, in which case the atmospheric glow around the moon is likely to blind the sensor to fainter stars around the planet.

The planets should be visible at the same time, so that the time between the measurements is short and movement of the observer and the planets between measurements is minimized.

3.5.2 Software

The Python *picamera* library was used to control the NoIR camera and the *gphoto2* library was used to control the Nikon D5000. Two python scripts were written for the acquisition: one to control the raspberry pi camera and other to control the Nikon camera. The scripts set the parameters for the cameras and released the shutter the predetermined amount of times.

The clock of the Raspberry Pi had to be set every time it was turned on, since it does not have an internal

battery. This was done using the notebook's clock and the SSH command in **Listing 3.1**.

Listing 3.1: Code to set the time on the raspberry pi.

```
ssh pi@192.168.0.2 sudo date -s "\"$(date -u --rfc-3339=seconds)\""
```

The Nikon pictures were stored on the Nikon camera and the Raspberry Pi pictures were stored on the Raspberry Pi. When acquisition finished, images were retrieved from the Raspberry Pi using SFTP and the Nikon memory card was inserted directly in the notebook.

3.5.3 Parameter selection

The camera lens, exposure time, sensitivity and focus were fixed. The parameters were chosen so that the brightest bodies of interest in the sky did not saturate the sensors, with the exception of Venus. Venus is so much brighter than everything else that it was preferrable to handle the saturation algorithmically.

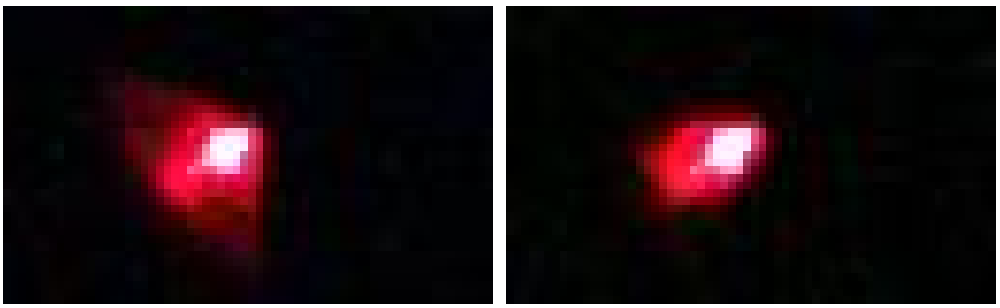
The chosen parameters for the Nikon were:

- autoiso=off; This disables the camera automatic iso setting.
- selftimerdelay="5 seconds"; The camera will wait 5 seconds to start the capture. It gives time for possible oscillations on the tripod to subside.
- longexpnr=1; Disables automatic long exposure noise reduction.
- f-number="2.8"; Reduce the lens aperture slightly to f/2.8.
- iso=400; Sets the iso at 400
- shutterspeed2="1/4"; Sets the exposure time to 250 milliseconds.
- capture interval: 2 seconds; Wait 2 seconds between pictures in a series.

The camera proved to be unable to take raw pictures using the gphoto interface at a speed higher than one frame every 2 seconds.

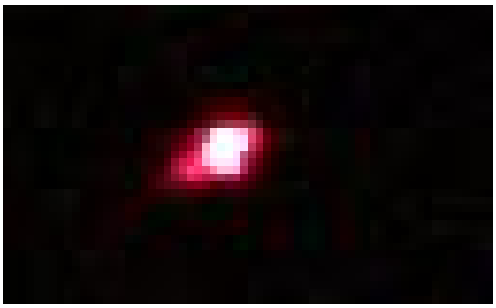
The aperture was chosen based on the quality of the image when stopping down the lens to smaller apertures. **Figure 3.16** displays pictures taken of a red led at the edge of the frame. The frame was cropped and only the small part of the picture that displays the led is shown. There is noticeable improvement from progressively stopping down from f/1.8 to f2.8, with f/1.8 displaying considerable coma - a type of optical aberration. There is no noticeable improvement stopping down further, so f/2.8 was chosen as the fixed aperture for performing the acquisitions.

Ideally the lowest possible iso setting should be chosen to reduce noise. In the case of the D5000, ISO 200 made it difficult to measure the noise in the dark frames because it was bordering the limits of the 12 bits A/D converter in the sensor. The ISO setting is usually the amount of analog gain applied to the signal coming from the sensor, so the gain was increased to 400 which provided a measurable noise floor. A measurable noise floor is important because it means that the quantization noise is not the dominant factor in the final noise. In that case, applying analog gain to the signal actually improves the signal to noise performance. For the Nikon

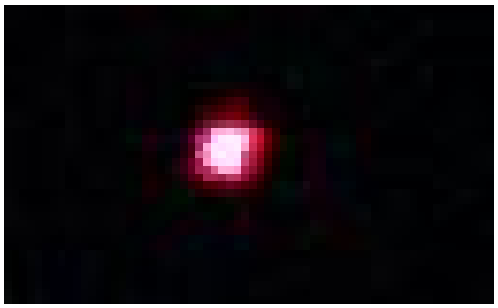


(a) f/1.8

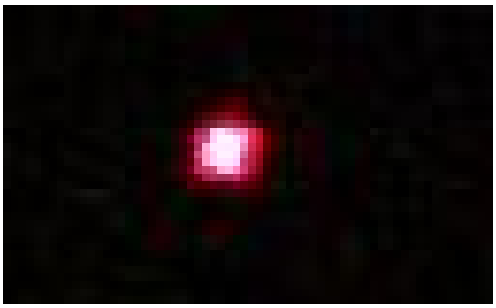
(b) f/2



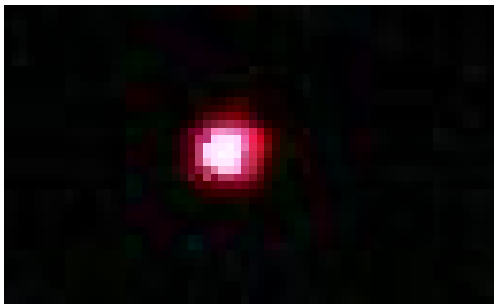
(c) f/2.2



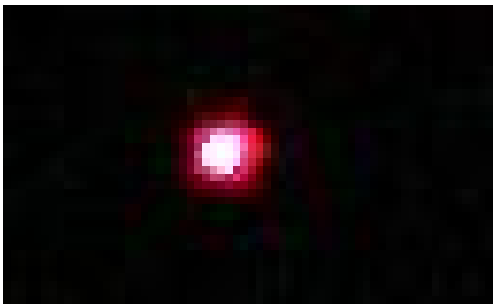
(d) f/2.5



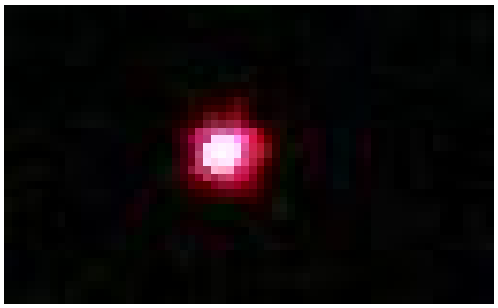
(e) f/2.8



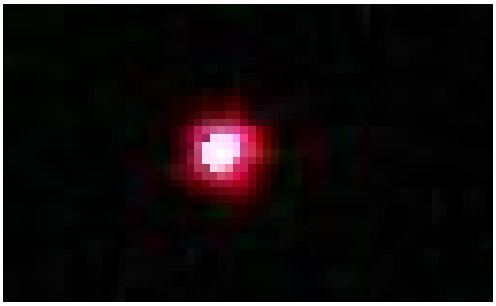
(f) f/3.2



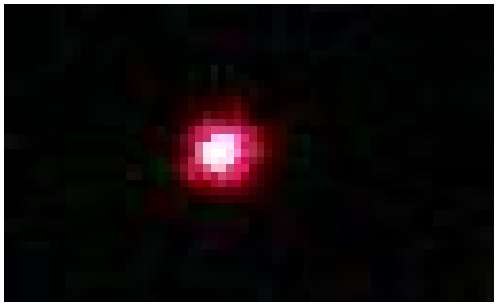
(g) f/3.5



(h) f/4.0



(i) f/4.5



(j) f/5.0

Figure 3.16: Red led imaged at the edge of the frame of the Nikon 50mm f/1.8 D lens on a D5000 at ISO 200 with different apertures.

sensor it is better to provide two images with ISO 400 and 1/4 s exposure time (1/2 s total exposure) than one image with ISO 200 and 1/2 s exposure time. The combination of the two exposures would theoretically be equivalent to the longer exposure using a 13 bits A/D.

The shutter speed chosen for the D5000 is 1/4 s (250 ms), which gives good exposure to most stars without saturating bright objects such as Sirius and Jupiter. Venus saturates slightly at that shutter speed, but it was subjectively found to be a good compromise between linearity and detection capacity. Two 1/8 s exposures could ideally be stacked to give twice the range, but the same problem with the A/D converter resolution is encountered.

The Raspberry Pi camera has a fixed aperture at f/2.9 and it was difficult to manually and independently set the digital gain (the "artificial" ISO) to 1 and the analog gain (the "real" ISO) to the desired value, so the capture script sets the target ISO and adjusts the exposure compensation and shutter speed until the camera stabilizes at a digital gain of 1. The script then locks the auto exposure mode, which freezes the digital gain at 1 and the analog gain at the desired value. After that, shutter speed is changed to the desired value without affecting the gain.

The chosen iso was 800, the Raspberry Pi camera maximum. The value was chosen to allow measuring the noise floor (the same situation with the D5000, but now with a 10 bits A/D instead of 12 bits), but in the case of the Raspberry Pi it was also a necessity due to the low light gathering capacity of the very small sensor. Setting the ISO at 200 or below would require 2 seconds for a correct exposure, the maximum allowed by the camera firmware. Increasing the ISO to 800 both decreases the exposure time to 1/2 s and allows measuring the noise floor.

All pictures for triangulation were taken in "raw" mode yielding the unprocessed values read at each pixel without any automatic noise reduction or debayering. The raw data used for processing is illustrated in part 2 of **Figure 3.22**.

3.5.4 Geometric Camera Calibration

Geometric calibration of the cameras was done automatically by the star identification algorithm. The star identification algorithm calculates a focal distance and up to three Brown distortion model (5) coefficients.

A few tuning parameters are available to penalize high higher order Brown coefficients that are too high. Trying to calibrate raspberry pi camera with different penalties for higher order distortion terms yields slightly different results. A few results for the raspberry pi camera are:

Penalty exponents for K_1, K_2 and K_3: 10,12,14
Number of star matches: 17
Final cost function value: 61.4
Focal distance (pixels): 2571.6
Pointing vector: [-0.978790, 0.204833, 0.003558]
Up vector: [0.190133, 0.914741, -0.356508]
K_1, K_2 and K_3: [-0.002038, -0.003773, -0.000354]

Penalty exponents for K_1, K_2 and K_3: 6,8,10
Number of star matches: 17
Final cost function value: 53.6
Focal distance (pixels): 2568.8
Pointing vector: [-0.978772, 0.204919, 0.003590]
Up vector: [0.190190, 0.914671 , -0.356657]
K_1, K_2 and K_3: [0.016, -0.061, -0.0062]

Penalty exponents for K_1, K_2 and K_3: 4,6,8
Number of star matches: 16
Final cost function value: 52.2
Focal distance (pixels): 2565.4
Pointing vector: [-0.978767, 0.204947, 0.003595]
Up vector: [0.190213, 0.914661, -0.356671]
K_1, K_2 and K_3: [0.035, -0.12, -0.011]

Penalty exponents for K_1, K_2 and K_3: 2,4,6
Number of star matches: 17
Final cost function value: 51.3
Focal distance (pixels): 2563.3
Pointing vector: [-0.978763, 0.204964, 0.003598]
Up vector: [0.190228, 0.914654, -0.356680]
K_1, K_2 and K_3: [0.047, -0.16, -0.0070]

Penalty exponents for K_1, K_2 and K_3: no penalty
Number of star matches: 16
Final cost function value: 57.4
Focal distance (pixels): 2560.2
Pointing vector: [-0.978743, 0.205058, 0.003666]
Up vector: [0.190281, 0.914593, -0.356810]
K_1, K_2 and K_3: [0.056, -0.092, -0.41]

The original nominal results were rounded for better visualization. Focal distances and cost functions are rounded to one decimal place, vectors are rounded to six decimal places and constants are rounded to two significant digits.

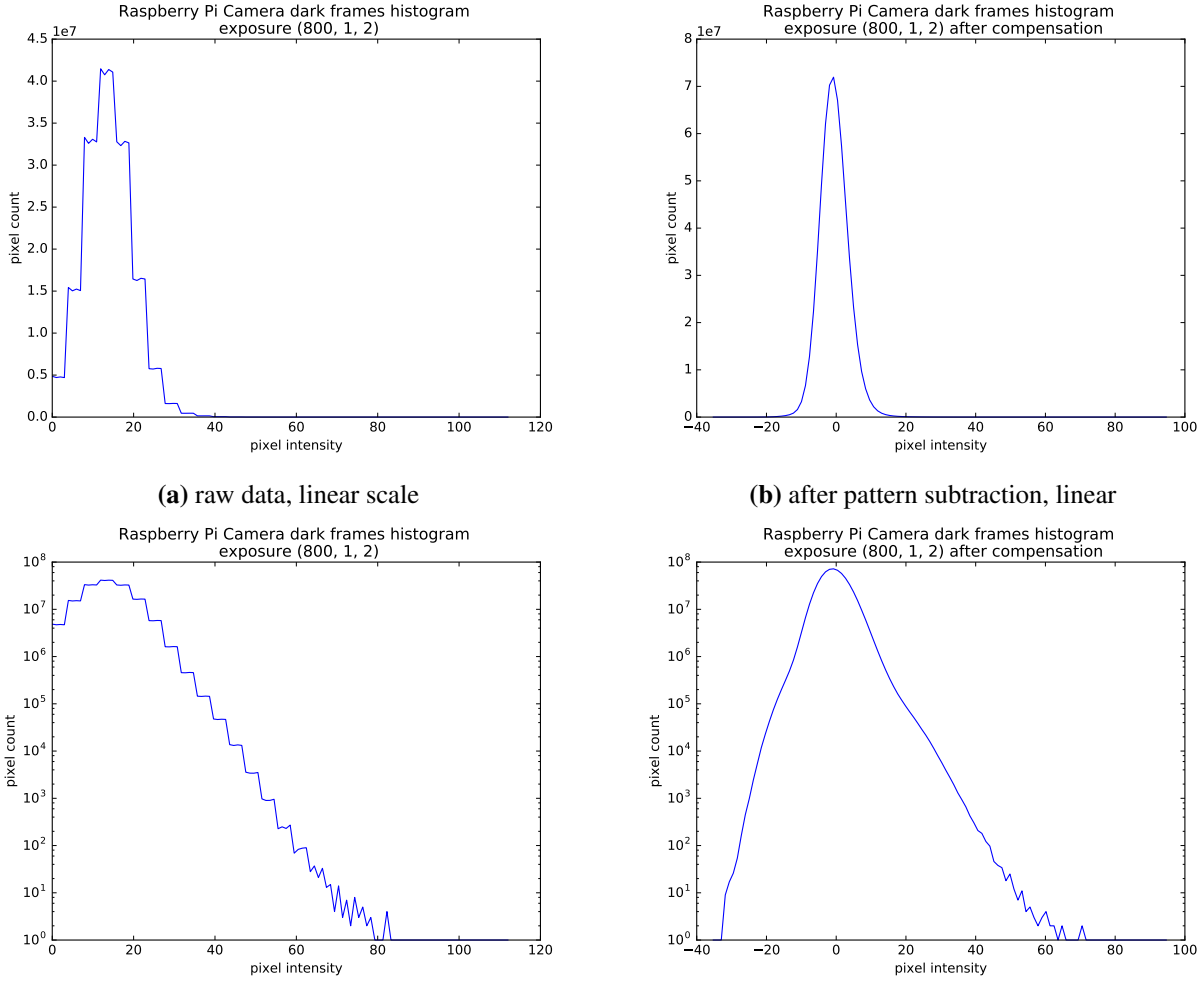
3.5.5 Radiometric Camera Calibration

The radiometric calibration of dark frames (fixed pattern noise) and individual pixel noise of the image is crucial to detect faint stars amongst the noise with confidence.

The calibration does not include individual pixel gain and linearity. Such a calibration is possible by flat fielding, and would yield more consistent star brightnesses across the frame, but the identification algorithm worked well without it.

Every picture session includes at least 10 pictures (usually 30 or more) taken with the Nikon lens cap in place or the shroud over the Raspberry Pi camera. These are called dark frames (pictures taken with the camera lens covered) and have to be taken just before and/or just after the main acquisition with the same camera settings.

Ideally, dark frame should be black, with all pixels reading a signal equal to 0. What happens in practice is that the dark frame registers at least two kinds of noise: a fixed-pattern noise and a variable noise. The fixed-pattern noise can be measured by averaging the dark frames.



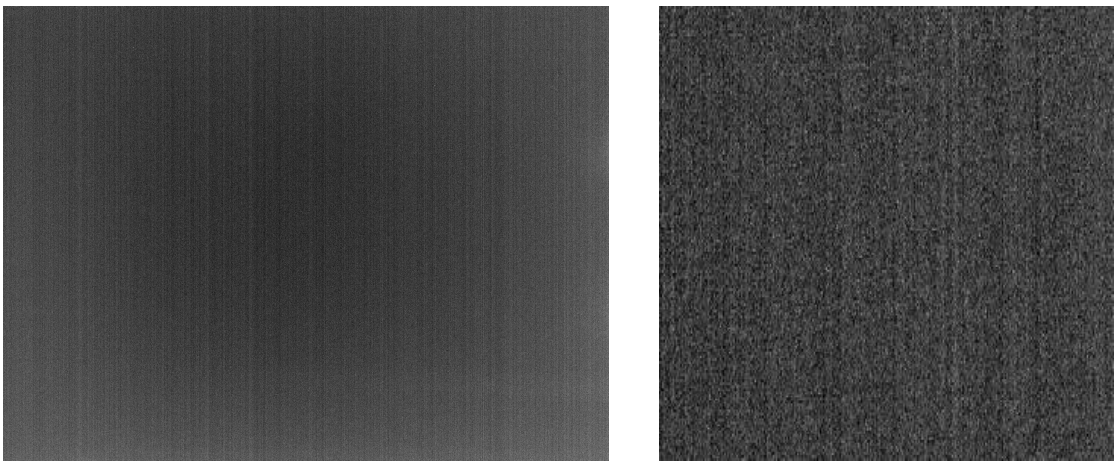
(a) raw data, linear scale

(b) after pattern subtraction, linear

(c) raw data, log10 scale

(d) after pattern subtraction, log10

Figure 3.17: Histogram sum of 120 dark frames taken with the Raspberry Pi camera at ISO 800 and 1/2 s exposure time. The compensated histograms are the sum of the histogram of the 120 frames after subtracting the dark frame (fixed pattern noise).



(a) full scene **(b)** a 300×300 pixels crop of 3.18a

Figure 3.18: Mean dark frame (fixed pattern noise) of the Raspberry Pi camera averaged over 120 dark frames.

3.6 Image processing

3.6.1 Preprocessing

Dcraw (8) was used to extract the raw values embedded in the .NEF files. A customized python script was written to extract the raw values from the raspberry Pi .jpg files.

Figure 3.19 displays a raw image obtained from the Raspberry Pi camera which displays Jupiter. At the top right corner of the image is possible to see part of a building wall.

The fixed pattern obtained through dark frame calibration is subtracted from the raw image. **Figure 3.21a** exhibits a crop straight from the camera, without bayer demosaicing, and **Figure 3.21b** shows the result after the initial dark frame subtraction.

All the preprocessing is done on the raw bayer-sampled data as illustrated in **Figure 3.22**. Part 2 of the image represents the raw data used, no reconstruction is attempted at the original data so no interpolated pixels are introduced in the calculation. All the image processing steps operate on the raw data.

The next step is to apply a high pass filter to remove the atmospheric glow from the background. **Figure 3.23a** displays the result of applying the high pass filter to **Figure 3.21b**. This step would not be necessary if the pictures were taken from space.

The resulting image has the interesting property that it may have pixels of negative value due to the filtering steps so far. The next filter smooths the image with a low pass filter producing **Figure 3.23b**, which is now suitable for ingestion by the segmentation algorithm.

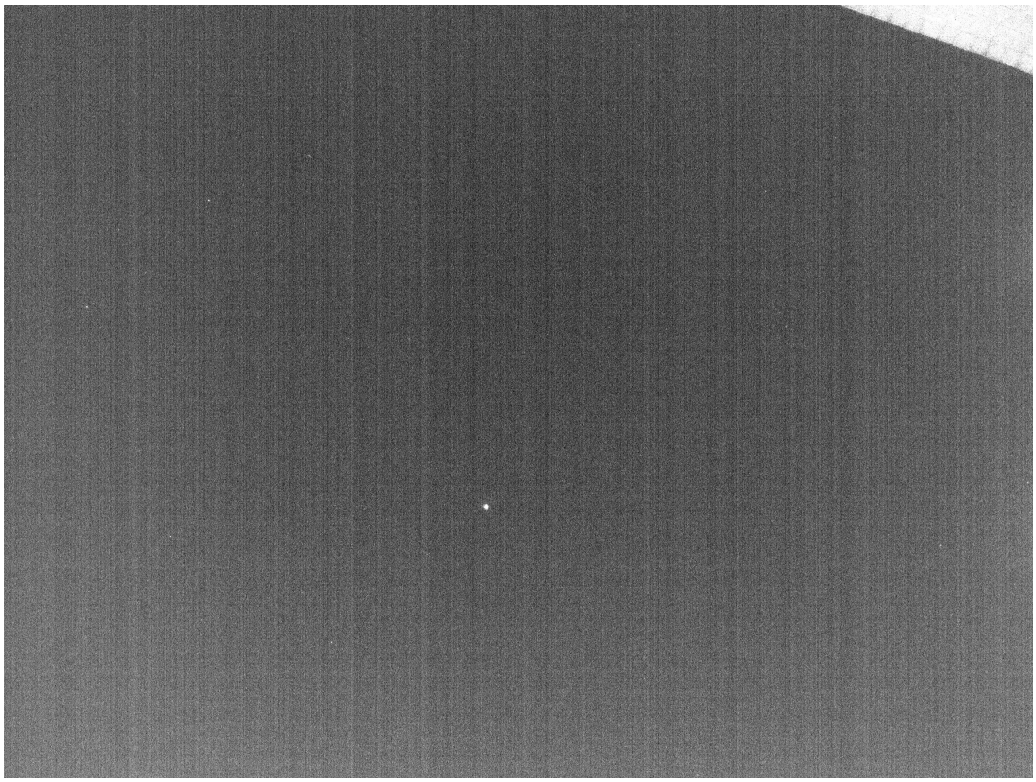


Figure 3.19: Raw image obtained from the Raspberry Pi camera

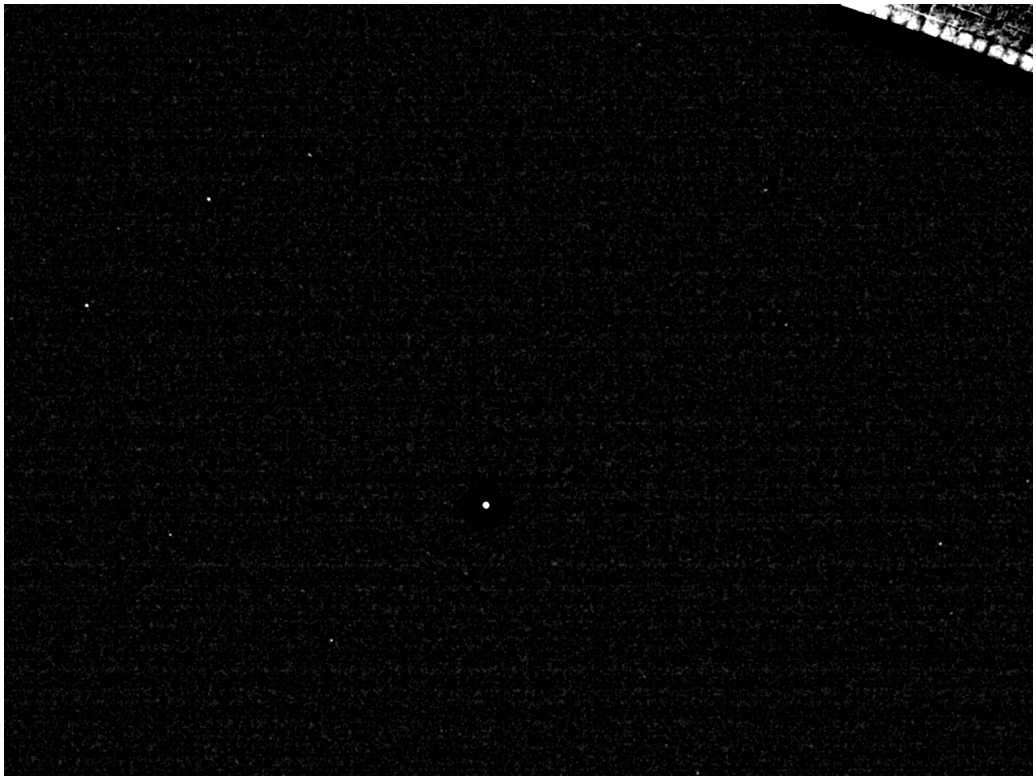
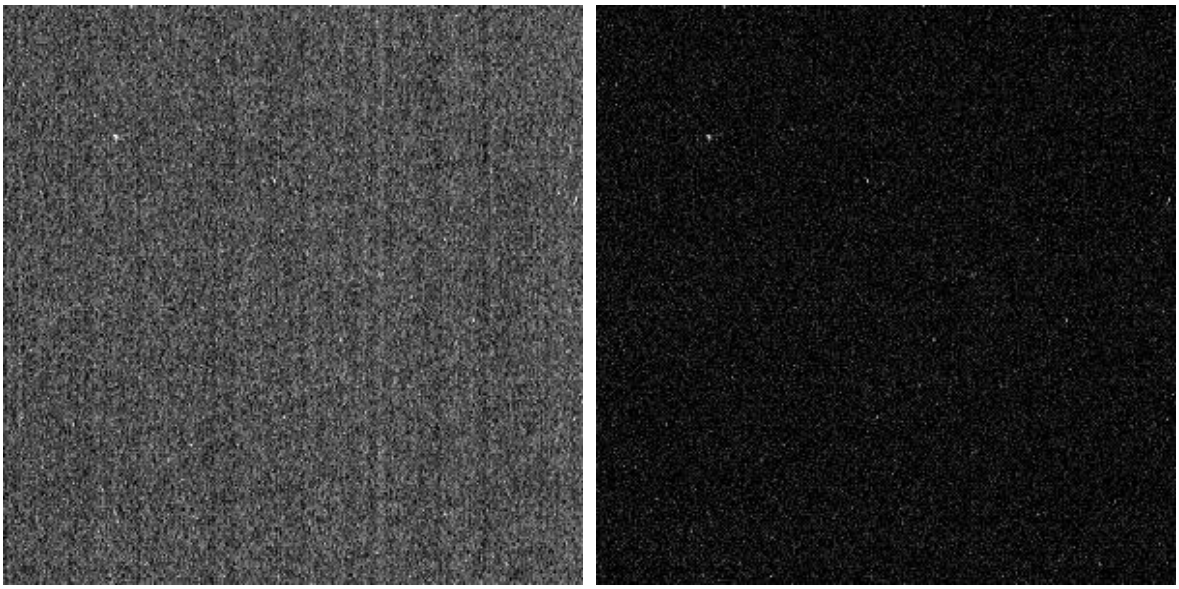


Figure 3.20: Final image after all preprocessing steps.



(a) raw image from the camera **(b)** after dark frame subtraction

Figure 3.21: 300×300 crop of a Raspberry Pi camera image.

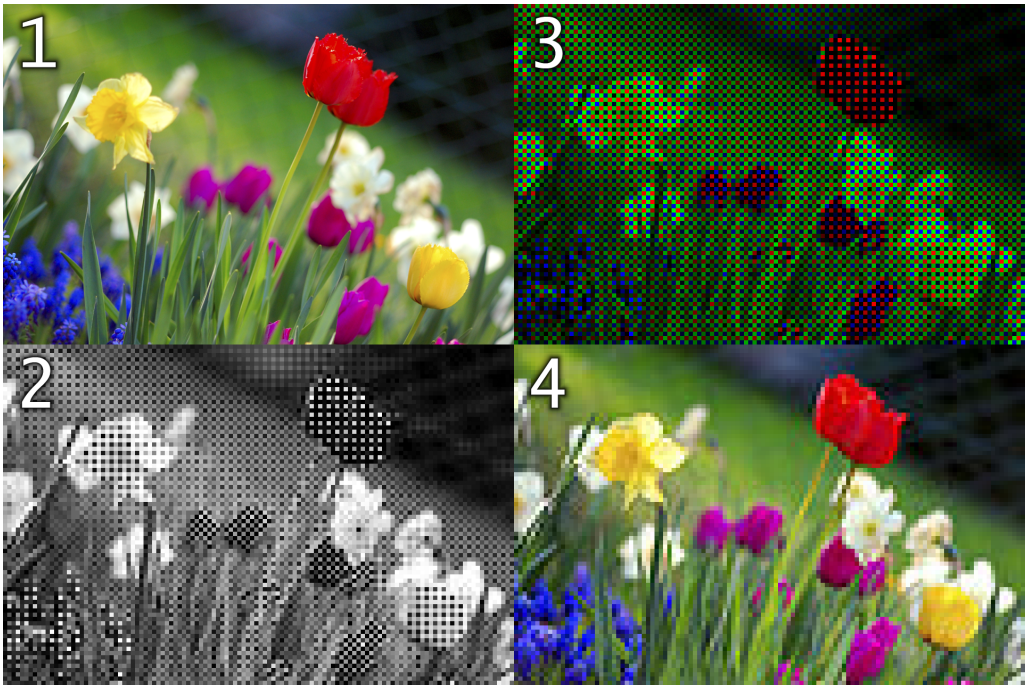
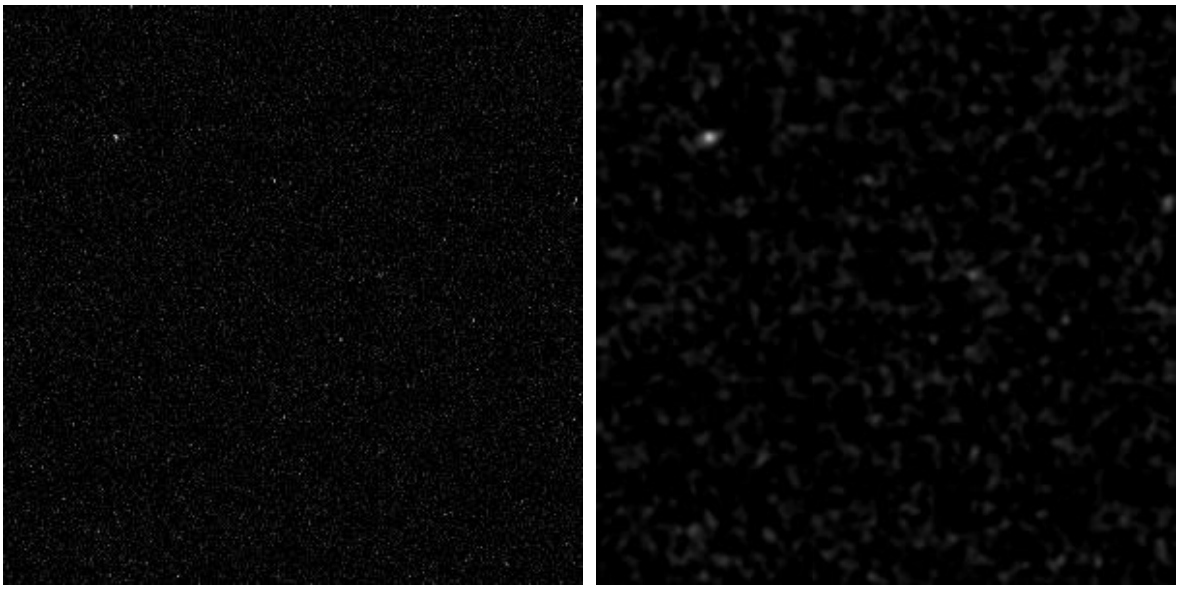


Figure 3.22: Bayer pattern illustration. Part 1 illustrates the ideal signal, part 2 shows the grayscale data acquired by the sensor at each pixel, part 3 shows the raw data with the sampled colors and part 4 shows a reconstruction of the original signal from the sampled data. Copyryght Anita Martinz (39)



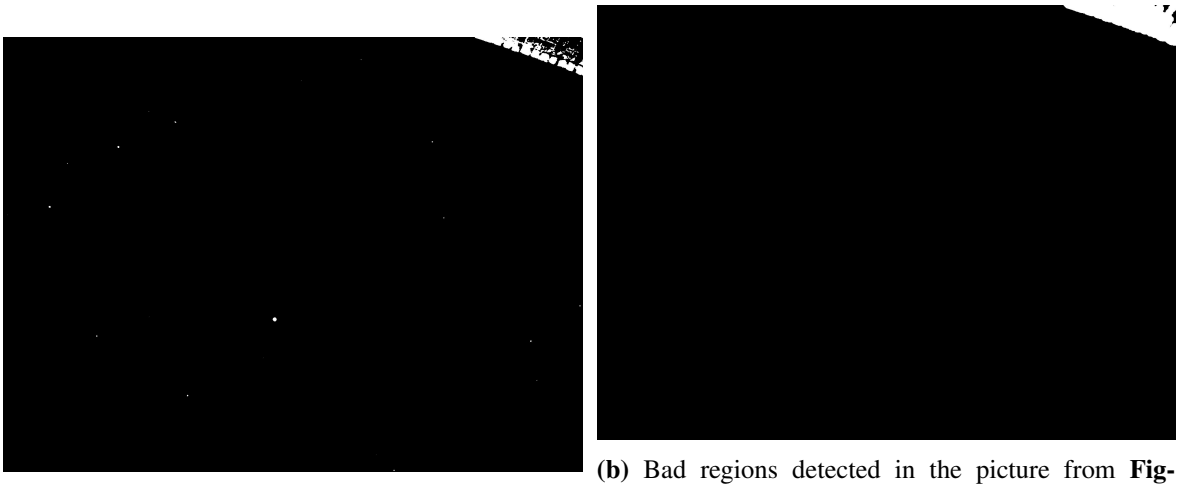
(a) Figure 3.21b after appling a high pass filter **(b) Figure 3.23a** after applying a low pass filter

Figure 3.23: Filters applied to the raw image after dark frame subtraction.

3.6.2 Segmentation

After preprocessing, a threshold is applied to the filtered image which is different in each pixel according to the individual pixel variance detected in the dark frames. The threshold for each pixel is three times the sum of the average standard deviation of all the pixels in the dark frames and the standard deviation of the specific pixel.

The application of the threshold produces a binary mask. The binary mask dilated and is labeled according to the image processing labeling and dilation algorithms in (19). The labeled groups that fit into a 50x50 square are considered star areas, and the larger ones are flagged as bad areas, which likely contains a bright obstruction, the moon or other large non-starlike object.



(a) Stars detected in the full picture from **Figure 3.20**

(b) Bad regions detected in the picture from **Figure 3.20**

Figure 3.24: Segmentation of the image in **Figure 3.20** showing pixel areas above the detection threshold and the bad areas for star detection. Part of the building wall was automatically flagged as a bad region.

On the Raspberry Pi the centroid detection can be challenging, since there are many false positives. The adopted approach is to cross-reference centroids between frames.

A centroid has to be confirmed in two frames to be selected for calculating an attitude solution. The criterion is that the position of the centroids in the two frames deviates less than two pixels from the mean centroid motion between the two frames.

Demanding that the centroid is confirmed in three consecutive frames reduces the number of confirmed features to approximately half in a specific sequence.

Once the centroids are found, they are measured by a sub-pixel algorithm that fits a gaussian curve with separated R, G and B channels to the Bayer pattern profile of the detected centroid. The cost function chosen does not penalize the solution when the fitted curve has higher values than the image in locations where the pixels are saturated.

An advantage of this method is that it can be used directly on the Bayer pattern of the image, without need to demosaic the three channels, and uses all available information on the original image for the fit, including the original variance of the pixels in the dark frames to assign weights.

Another advantage of this method is that it is possible to map the point spread function of the camera for every different location of the frame and use it instead of the gaussian. This has not been done for this work, though.

The result of this measurement is an array of measured centroids with their respective positions and brightness on each channel.

3.6.3 *Starprint* calculation

The procedure described in **Section 3.3** is now applied to the centroids obtained after segmentation, with the difference that it is now desirable to generate starprints for all the brightest stars in the image (not only of stars chosen as reference stars).

Notice that now some bright stars may be near the border of the image. The *starprint* strategy of choosing the brighter star from each slice will be an advantage here - even if the neighborhood of a star is partially outside the image, the part that is inside the image will likely generate coinciding elements with the star catalog since the angles will be calculated from and to the brightest star in each "slice".

3.7 Attitude calculation - star tracker algorithm

This is "Step 3" from the diagram in **Figure 3.1**. In this step the "lost in space" problem is solved by comparing the *starprints* calculated from the image and the ones calculated from the catalog (55). A rough solution for each matching *starprint* element is computed and a vote is cast for a region in the sky that the camera is pointing to ("pointing vector") and the camera's rotation ("up vector"). Votes going to approximately the same location are grouped, and the region with the most votes is elected as a candidate solution.

The initial attitude calculation is the "lost in space" problem, which is the first attempt of a star tracker to infer its orientation in space. The main difference between the problem approached here is that the camera in a conventional star tracker is calibrated and its parameters are known, and the cameras used in this work are uncalibrated and no parameters are known a priori. The proposed algorithm not only calculates the direction to where the camera is pointing, it also provides several calibration parameters such as field of view, focal distance (in pixels) and distortions.

To identify the stars in the picture the database of catalog *starprints* is searched for elements which coincide or are very similar to *starprint* elements extracted from the figure. The components are defined by an angle, a distance and three stars (see **Section 3.3**). A match between a component from the catalog and a component from the image implies a correspondence between three catalog stars, and therefore, a possible solution to the camera's pointing vector, up vector and focal distance. The star matches and the solutions corresponding to them are stored.

Results that point to roughly the same solution are grouped even if there are no stars in common between them. The groups are sorted in descending order of number of matches of *starprint* components. The groups with a larger amount of matches stay at the beginning of the list.

The groups are tested one by one using simulated views of the sky as seen from a camera with the solution parameters. At this point all detected stars and all catalog stars corresponding to that region are used, and the correspondence is based on the position of stars in the frame and their projected positions using the catalog and the solution provided.

The initial group estimate is iteratively refined by optimizing the pointing vector, up vector and focal distance to try to minimize the sum of squared errors between the detected and the projected position of stars.

When the error can be reduced no further, the brightness of the matching stars in the catalog and the measured brightness in the image are submitted to a linear fit and a magnitude detection threshold is estimated. All matches that fall too far from the fit and all matches below the estimated brightness detection threshold are disregarded. This will exclude stars which apparently match in location but have a much higher or much lower brightness than expected according to the fit.

Figure 3.25 is an example of such fit in a jpg picture which is a processed image and is not expected to perform linear photometry. The y axis is the logarithm in base $\sqrt[5]{100}$ of the sum of the pixel values of the stars detected in the image. The x axis is the visual magnitude of the corresponding catalog star. The blue line is the linear best fit and the red markers show stars with a difference of 0.75 (100% more or 50% less) to the blue line.

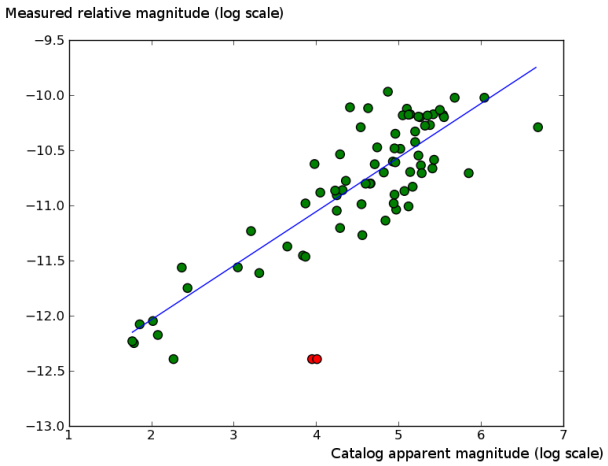


Figure 3.25: Catalog brightness versus measured brightness of stars identified in a jpeg image. Axes are logarithmic and have no units. An increment of 5 along each axis corresponds to a $100\times$ increment in brightness.

For every group of matches after the brightness filtering, a result is generated containing: mean quadratic error, number of matching stars, number of stars in the image without catalog counterparts ("extra stars") and stars in the catalog without and image counterpart ("missing stars").

The chosen result is the one that has the largest number of matching stars but keeps the quadratic error, the number of "missing stars" and the number of "extra stars" under a predetermined limits. If there is a candidate that satisfies all requirements, the identification is considered to be successful.

When a suitable candidate solution is found, the result is then refined, with the added complexity of a distortion model. The final result is the camera attitude, it's calibration (distortion parameters and focal length), the star identifications and the position of unknown bodies in the image.

The refining process is done iteratively by using optimization algorithms with a cost function modeling the error between the estimated and measured centroid positions in the image. Star identification is reassessed between runs of the variable optimization, in a RANSAC approach like Klaus et al. (31), so that the criterion for a match grows more stringent as the accuracy of the solution improves.

In a nutshell, the steps are:

- Search for (r, θ) *starprint* tuples from the catalog that correspond to tuples obtained from the image.
- For every corresponding tuple, estimate which would be the camera solution (pointing vector, up vector and focal distance) that would provide the correspondence.
- Group all the matches that correspond to a similar solution.
- Test each of the groups' solution with all the centroids against the full star catalog.
- Select the best match based on the test results.
- Refine the focal distance, attitude and distortion parameters of the best match found.

3.8 Position calculation

Calculating the position of the observer is a problem of triangulation. Once the stars are identified, there are "star like" objects left without matches from the catalog. If the object is known to be a specific planet, and if the coordinates of the planet are known, it is possible to determine constraints to the possible position of the observer.

All coordinates are cartesian (x, y, z) and refer to the International Coordinate Reference System - ICRF (17) and (23). In this work we adopt the barycenter of the Solar Sytem as the origin of the asis. The ICRF is the same reference system utilized in the Hipparcos (34) and Bright Star Catalogue (21).

To make a full position calculation, the navigation system needs the following data for at least **two** bodies:

1. The location planets in the Cartesian ICRF coordinates as a function of time.
2. A positive identification of what planet, if any, corresponds to each unmatched star-like object in the scene.
3. An accurate measurement of the apparent angular position (Right Ascension and Declination) of the body in question from the observed point as per the ICRF.

If the observer is moving according to the chosen reference coordinates and there is a time difference between the measurements taken of the two bodies, the speed of the observer has to be taken into account and multiple observations with a time interval between them are required to also solve for the speed of the observer.

Having the above information for a single body instead of two will constrain the possible observer's position to a very thin cone-like volume with the vertex at the planet. The larger the estimated error in the planet's angular position, the wider the cone. A second body will produce a second volume, and the intersection of the two will determine the position of the observer. More bodies will reduce the error margin.

The algorithm needs a list of dark frame files and a list of image files and the Horizons data of the correct point in time. At this stage in development an internet connection to the Horizons database is needed to get the planet positions. The processing steps are the following:

1. Calculate the average dark frame and dark frame variance.
2. Read the acquisition date of all the input files.
3. Download the position and speed of planets at the time of the pictures.
4. Perform the star identification steps for each of the input images.
5. Calculate the position on the sky for all the unidentified bodies on the input images.
6. Group the identified bodies by sky position and make a list of candidate bodies for matching based on the mean calculated position.
7. Perform the position calculation for all planet combinations for the three unidentified bodies with the most confirmations in input images.
8. Output the solution that is most consistent.

3.8.1 Planet absolute positions

The location of the planets as a function of time is provided by the Nasa Horizons database (47). In this work a Python script was developed to operate the Horizons database using it's telnet text interface to obtain standardized tables of times, positions and speeds for planets in the solar system. Care was taken to download the minimum necessary amount of data and avoid overloading the system.

In the experiments performed, the position of the observer is compared to the position provided by Horizons for Earth. The position of Earth, however, is not fed into the algorithm, and is used only to evaluate the result.

3.8.2 Planet identification and position calculation

As in the attitude calculation, identification of the planets and position calculation happens at the same time because it is necessary to find a solution that provides a consistent result. It is possible, however, that an incorrect solution may also be geometrically consistent; such ambiguities can be resolved by imaging a third body, performing more observations along a few weeks, or having at least a vague notion of the initial position of the observer.

It is assumed that an autonomous spacecraft has at least a crude estimate of it's current position that would be enough to make the identification and then refine the measurement, but for the purposes of this work, the location of the observer (Earth) is not fed into the algorithm, but is used to measure the error of the solution and servers as a ground truth value.

All the planets except for Neptune are considered to be possible matches. We also do not expect to use the Earth's moon position nor the position of the Sun. The positions and velocities of the candidate planets at the time of the pictures are downloaded from the Horizons database.

The problem is not as simple as determining the coordinates in space that are at the intersection of two straight lines. The trajectory of the planet as a function of time - or ephemeris - has to be known, since light

takes several minutes or sometimes several hours to traverse the distance between planets. If one determines the position of the Earth with one hour of error that position would be incorrect by roughly $30km/s \times 3600s = 1.08 \times 10^5 m$, which is about 8.5 times the diameter of the planet.

Based on the Horizons information, the position of the planet is interpolated from the first data point, which has a time, a (x, y, z) position at that time and a (v_x, v_y, v_z) velocity. This linear approximation will be worse the greater the time difference between the horizons data point and the time in which the position is desired.

All the planet combinations are tried with the unidentified bodies that are shown in the images and a solution is iterated. The solution consists of the (x, y, z) observer position. For every combination, the cost function takes the position and estimates the angular position on the sky where the planets in the combination would appear if observed from the given position and at the time each picture was taken. The algorithm estimates where the planet would be at the moment the light being observed at the time of the picture departed from the observed planet.

To include the observer speed in the solution it would be necessary to space the observations along several days.

Chapter 4

Results

4.1 Star identification algorithm

The proposed algorithm was tested on a variety of conditions. The tests are described in this section. Some of the images used in the tests were pictures taken by the author and other were obtained from the Internet.

4.1.1 Test under standard conditions

In this section, the algorithm is tested with pictures taken with a regular camera, without assistance of specialized equipment, in conditions that may be reproduced by amateur photographers in a domestic environment. Those are the conditions for which the star identification algorithm was originally designed for.

Those conditions include distorted pictures, noise, possible obstructions such as clouds, planets and unknown camera parameters.

The picture in **Figure 4.1**, was taken by the author of this work, from the rooftop of a 17 floor building at São José do Campos, SP, Brazil in 2005-11-21 at 21h42. The sky was clean, with few clouds. The camera used was a Canon PowerShot S1 IS (see **Section 3.1**) mounted on a mini tripod.

The pictures are 2048 pixels wide by 1536 pixels tall and the focal distance recorded on EXIF metadata is 5.8 mm at the minimum zoom.

The measured barrel distortion in Dpreview's assessment (15) was around 1.2% with the zoom set to minimum. The distortion model used is not mentioned, but the reviewer provides an image of a grid that visually shows the camera distortion (**Figure 4.2a**).

Running the algorithm as described in **Section 3.7** yielded the focal distance of 2242.12 pixels. The formulas to convert the focal distance in millimeters are:

$$
\begin{aligned}
focal\_distance\_pixels &= focal\_distance\_mm \frac{sensor\_width\_pixels}{sensor\_width\_mm} \\
focal\_distance\_pixels &= 5.8 \frac{2048}{5.27} = 2253.96 \\
&\textit{or} \\
focal\_distance\_pixels &= focal\_distance\_mm \frac{sensor\_height\_pixels}{sensor\_height\_mm} \\
focal\_distance\_pixels &= 5.8 \frac{1536}{3.96} = 2249.69
\end{aligned}
$$

Those formulas should produce the same result in case the ratio between the images height and width are

Figure 4.1: Picture taken from the rooftop of a building in 2005-11-21 at 21h42 local time. An antenna, a lightning rod and part of a wall can be seen.

the same as the ratio between the sensors height and width. The camera presented a difference of less than 0.2% between the ratios:

$$\frac{\frac{5.27}{3.96}}{\frac{2048}{1536}} = 1.0019$$

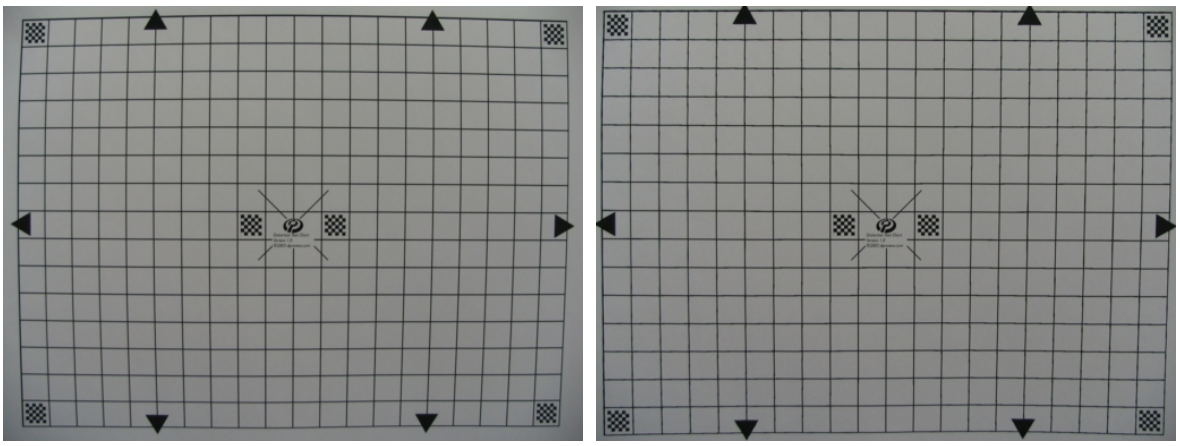
In relation to the 5.8 mm focal distance informed in the metadata, the results obtained were within 0.5% of the expected. For this test, it's considered that the algorithm calculated the correct value for the focal distance.

On the first iteration, the algorithm identified the camera parameters without taking into account possible distortions, which led to 21 stars being identified. On the second iteration, the results of the first one were used as input and the camera distortions were estimated. A better model of the lens was obtained leading to the identification of 31 stars.

To rectify an image produced by the camera using the distortion coefficients (in other words, to "undistort" it), the rectified image is rebuilt pixel by pixel according to the distortion model, so it's possible to convert the coordinates from the adjusted image to the distorted one. The image in **Figure 4.2a** was rectified with the distortion coefficients $(-0.0308475, -0.15502425, 0.1329343)$ to produce the image in **Figure 4.2b**.

The lines closer to the image edges are noticeably curved on the original image and their curvature is reduced in the corrected image. The fact that the distortion was not fully corrected can mean that the algorithm did not converge to the optimal solution or it can also mean that there is an optical difference between cameras of that model, or it could also mean that the distortion model does not fully model that specific optical system.

Star identification is performed in two steps. The first one does not take distortions into account and the second one estimates the distortion parameters and is able to identify more stars. **Figure 4.3** display the results



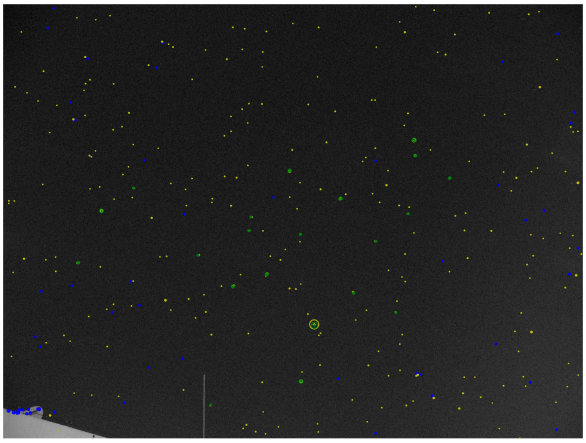
(a) Original grid from (14) **(b)** After rectification

Figure 4.2: Grid photographed by a camera of the same model from (14) before and after correcting with the distortion coefficients found from star identification.

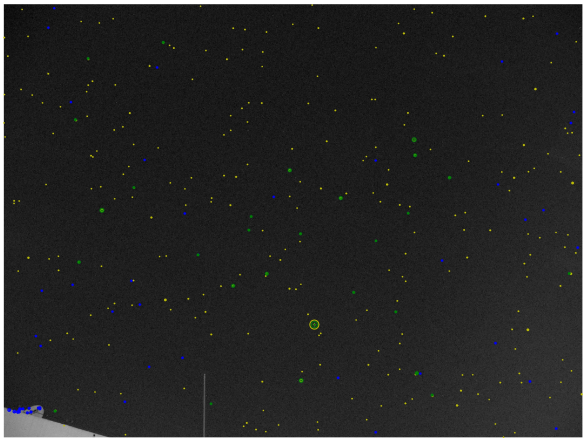
of the first and second steps mentioned before.

On **Figure 4.3**, images to the left are a visual representation of the solution found by the algorithm using a lens model without distortions. Images to the right represent the solution when using the model with distortions. Images on the second line are close ups of the images above them. Detected stars are represented as green and blue markers. Yellow markers represent "missing" stars, which the algorithm expected to find but did not. Green markers represent correct matches. Larger circles represent brighter images.

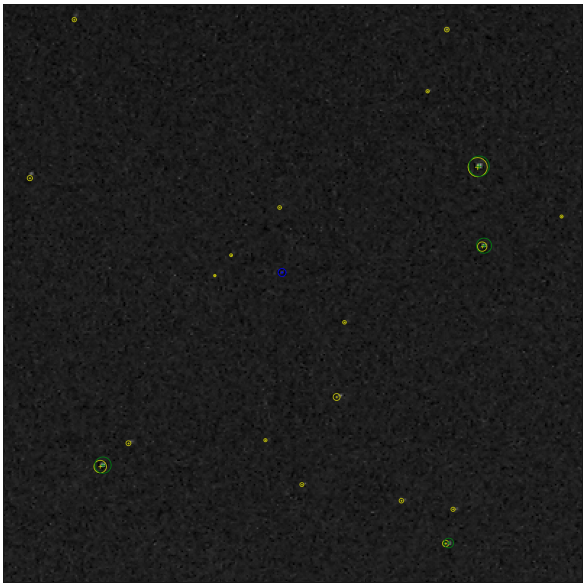
Figure 4.4 displays the identified star names as described in the BSC (21). The solution without distortions is displayed to the left and the refined solution with distortions is displayed to the right.



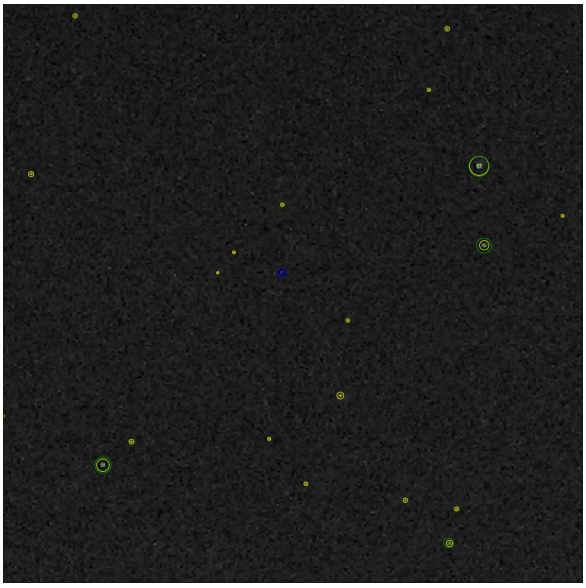
(a) Solution using an ideal camera model



(b) Solution using model with distortions

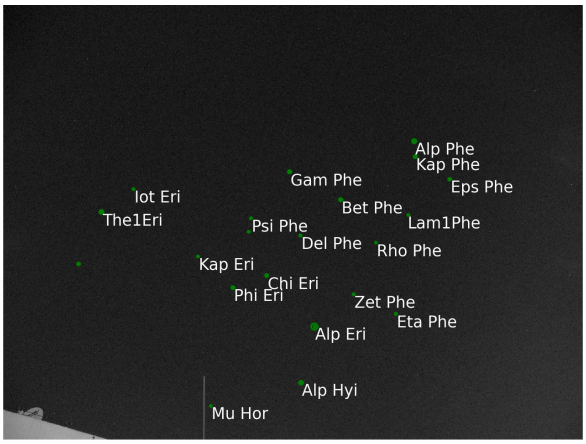


(c) Close-up of the ideal camera solution

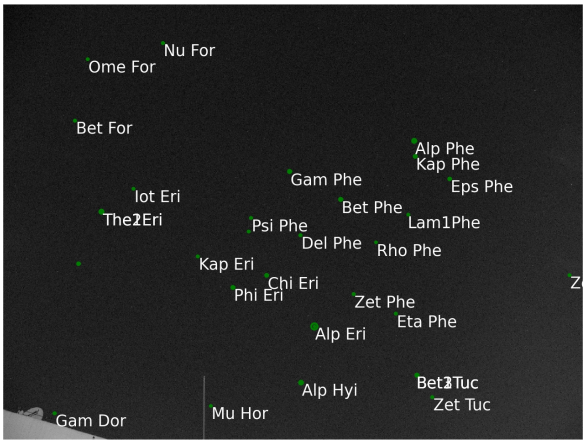


(d) Close-up of the solution using distortions

Figure 4.3: Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars.



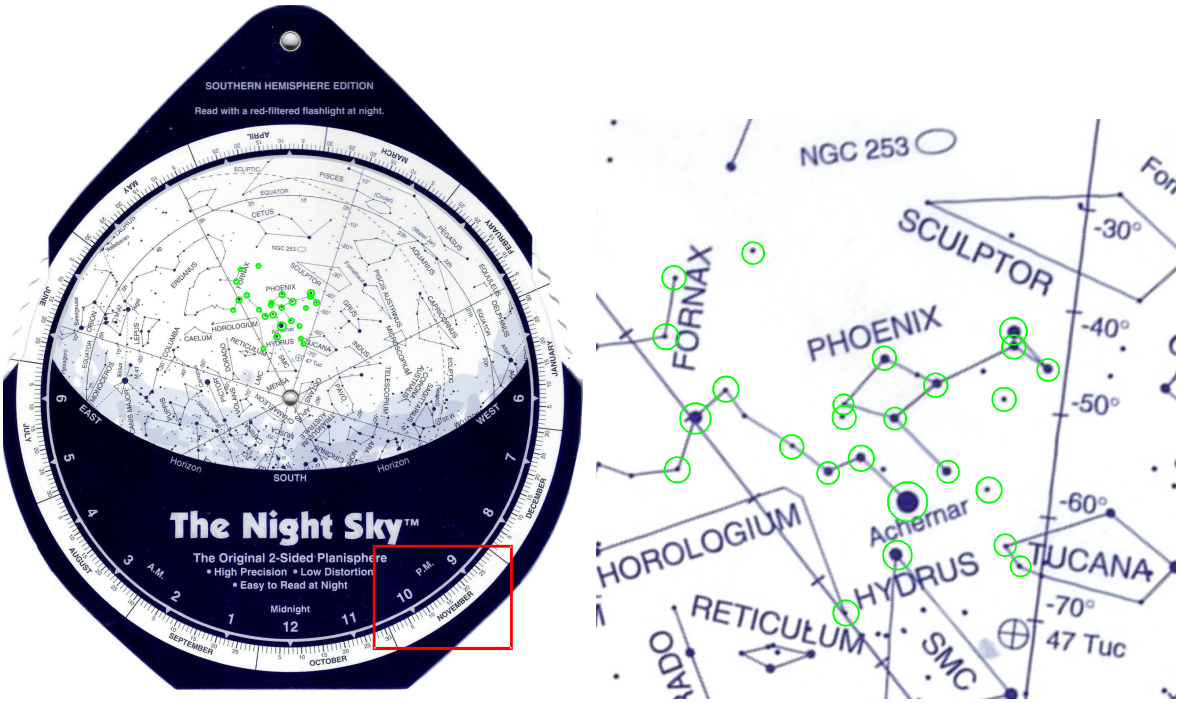
(a) Solution using an ideal camera model



(b) Solution using model with distortions

Figure 4.4: Star names as described in the BSC (21) for the stars identified in **Figure 4.1**.

The consistency of the results was checked using a paper map "The Night Sky" (6). The map was set to display the Southern Hemisphere around the time of the year the picture was taken. There weren't identified any relevant divergences, like important stars not apperaring in the image nor bright spots on the image which do not correspond to stars. **Figure 4.5** shows the "The Night Sky" (6) paper map which displays several stars that were not detected by the segmentation algorithm but appear faintly on the image.



(a) "The Night Sky" paper map **(b)** region of interest

Figure 4.5: Paper map "The Night Sky" (6), the red square shows the alignment at 2005-11-11 at 21:45, the green circles show identified stars on **Figure 4.4**

Table 4.1 and **Table 4.2** display the results of the first and second identifications.

21 stars were identified while using the model without distortions and 31 stars were identified while using the model with distortions. That can be observed in **Figure 4.3** with the increase in the number of green markers and in **Figure 4.4**. Using the model with distortions increased the amount of identified stars and reduced the difference between the expected position and the position in which the stars were detected as shown in **Table 4.1** and in **Table 4.2**.

The paper map in **Figure 4.5** was used to verify whether the region of the sky indicated by the algorithm was in fact visible on the sky at that time, date and geographic location (southern hemisphere).

Even though the original image in **Figure 4.1** contains noise, foreign bodies and distortions, it was possible to:

- estimate the camera focal distance
- estimate the camera distortion parameters
- identify individual stars
- identify a region of the sky

Table 4.1: Matches found for **Figure 4.1** using the model **without distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0.46	472	Alp Eri	(75.0,-364.3)	(75.7,-367.5)	3.4
2.39	99	Alp Phe	(427.6,287.4)	(428.5,288.1)	1.2
2.86	591	Alp Hyi	(28.2,-565.2)	(28.6,-566.1)	0.9
3.24	897	The1Eri	(-676.8,37.7)	(-676.7,37.8)	0.1
3.31	322	Bet Phe	(166.8,80.2)	(168.8,81.2)	2.2
3.41	429	Gam Phe	(-11.4,177.0)	(-11.9,179.8)	2.8
3.56	674	Phi Eri	(-209.7,-227.6)	(-212.1,-230.1)	3.4
3.70	566	Chi Eri	(-91.6,-184.8)	(-93.0,-187.1)	2.8
3.88	25	Eps Phe	(554.3,153.8)	(553.9,153.6)	0.4
3.92	338	Zet Phe	(213.0,-251.0)	(214.6,-253.4)	2.9
3.94	100	Kap Phe	(430.7,232.4)	(432.0,233.1)	1.4
3.95	440	Del Phe	(26.1,-44.9)	(26.4,-45.5)	0.7
4.11	794	Iot Eri	(-562.1,118.1)	(-563.7,118.6)	1.6
4.25	721	Kap Eri	(-332.5,-118.9)	(-335.7,-119.7)	3.2
4.27	1008		(-761.2,-146.5)	(-757.9,-146.1)	3.3
4.36	191	Eta Phe	(362.5,-321.0)	(363.6,-322.4)	1.7
4.41	555	Psi Phe	(-145.6,15.6)	(-148.4,15.7)	2.8
4.77	125	Lam1Phe	(405.5,27.0)	(407.6,27.4)	2.2
4.83	574		(-153.9,-31.9)	(-156.5,-32.4)	2.7
5.11	934	Mu Hor	(-291.9,-650.9)	(-290.4,-648.5)	2.8
5.22	242	Rho Phem	(290.5,-69.9)	(293.5,-70.5)	3.0

- validate the focal distance from the camera specs at (15)
- partially validate the distortion calculation by applying an inverse correction to a picture of a grid taken with another camera of the same model.
- validate names and relative positions of identified stars using a paper map.
- validate the distortion calculation effectiveness in reducing the error and increasing the number of identified stars
- validate the final result by visually inspecting the image
- validate the identification of the region of the sky by using a paper map.

Table 4.2: Matches found for **Figure 4.1** using the model **with distortions**.

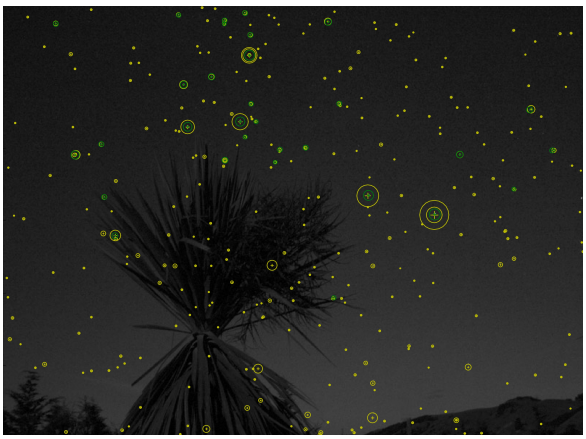
Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0.46	472	Alp Eri	(75.3,-367.6)	(75.7,-367.5)	0.4
2.39	99	Alp Phe	(428.6,288.3)	(428.5,288.1)	0.2
2.86	591	Alp Hyi	(28.0,-566.4)	(28.6,-566.1)	0.6
3.24	897	The1Eri	(-675.3,37.2)	(-676.7,37.8)	1.6
3.31	322	Bet Phe	(168.7,81.1)	(168.8,81.2)	0.1
3.41	429	Gam Phe	(-12.0,179.3)	(-11.9,179.8)	0.5
3.56	674	Phi Eri	(-212.2,-230.3)	(-212.1,-230.1)	0.3
3.70	566	Chi Eri	(-93.1,-187.5)	(-93.0,-187.1)	0.4
3.88	25	Eps Phe	(554.5,153.9)	(553.9,153.6)	0.7
3.92	338	Zet Phe	(214.8,-253.7)	(214.6,-253.4)	0.3
3.94	100	Kap Phe	(432.2,233.4)	(432.0,233.1)	0.3
3.95	440	Del Phe	(26.2,-45.8)	(26.4,-45.5)	0.4
4.11	794	Iot Eri	(-563.3,117.8)	(-563.7,118.6)	0.8
4.12	8747	Zet Gru	(983.3,-187.3)	(977.4,-186.4)	6.0
4.23	77	Zet Tuc	(493.3,-620.0)	(492.4,-617.5)	2.7
4.25	1338	Gam Dor	(-841.3,-673.2)	(-842.5,-674.5)	1.8
4.25	721	Kap Eri	(-335.9,-120.3)	(-336.4,-120.4)	0.5
4.27	1008	Lam1Phe	(-756.4,-145.9)	(-757.9,-146.1)	1.5
4.35	898	The2Eri	(-675.4,37.1)	(-676.7,37.8)	1.5
4.36	191	Eta Phe	(363.9,-322.7)	(363.6,-322.4)	0.4
4.37	126	Bet1Tuc	(439.1,-541.2)	(438.2,-539.9)	1.5
4.41	555	Psi Phe	(-148.2,15.6)	(-148.4,15.7)	0.2
4.46	841	Bet For	(-768.7,357.3)	(-770.5,359.3)	2.8
4.54	127	Bet2Tuc	(439.0,-541.5)	(438.2,-539.9)	1.8
4.69	612	Nu For	(-459.5,631.4)	(-459.5,633.4)	2.0
4.77	125	Lam1Phe	(408.1,27.1)	(407.6,27.4)	0.5
4.83	574		(-156.5,-32.7)	(-156.5,-32.4)	0.3
4.90	749	Ome For	(-724.8,573.9)	(-725.6,576.4)	2.7
5.09	136	Bet3Tuc	(433.7,-544.4)	(438.2,-539.9)	6.3
5.11	934	Mu Hor	(-290.9,-648.5)	(-290.4,-648.5)	0.4
5.22	242	Rho Phe	(293.3,-70.8)	(293.5,-70.5)	0.3

4.1.2 Test with dark field of view obstruction

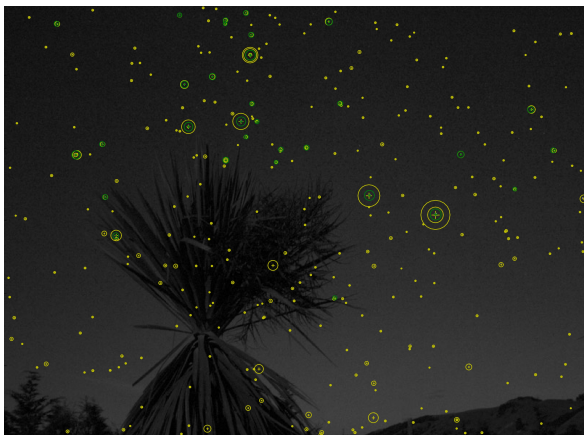
The lower part of **Figure 4.6** contains vegetation and terrain, the upper part consists of a starry sky and a plant occupies the central lower part, obstructing part of the night sky. According to the legend in (54), the Southern Cross appears in the image and we expect it to be identified by the algorithm.

Figure 4.6: "Southern Cross by Moonlight". New Zealand landscape photographed with a Canon 10D in January 2005, in Kaituna, Wairarapa, New Zealand. Exposure time was 10 s, with a 20mm lens at f/4 (54). A plant obstructs the view of the sky.

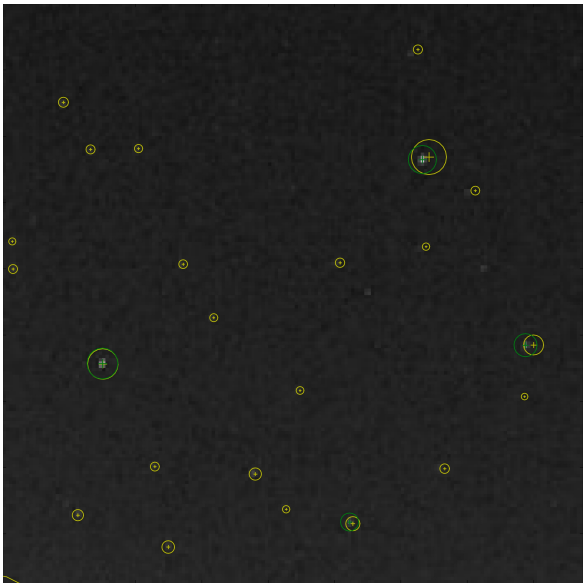
Figure 4.7d display the results of processing **Figure 4.6** using the lens model without distortions and with distortions.



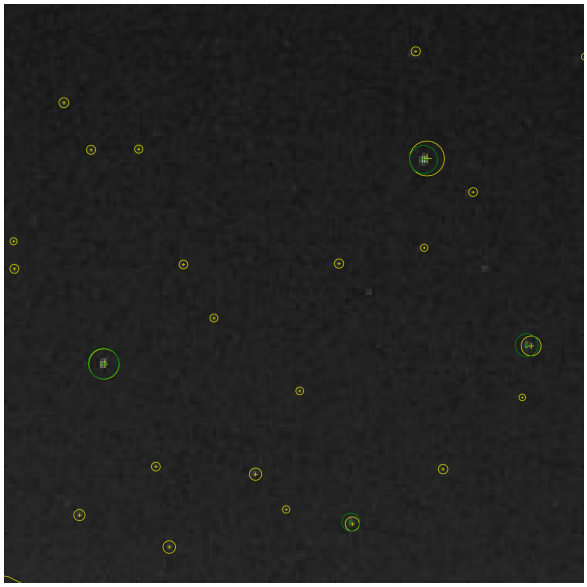
(a) Solution using an ideal camera model



(b) Solution using model with distortions

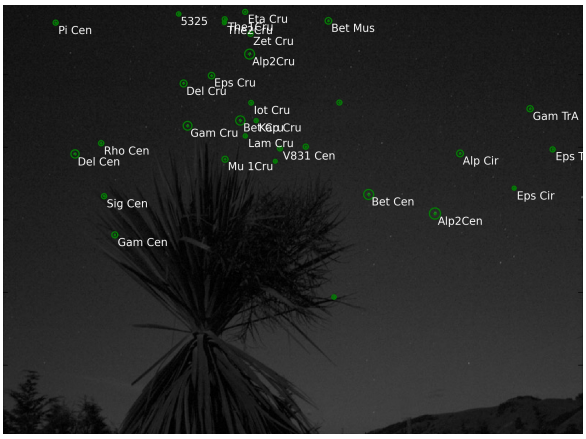


(c) Close-up of the ideal camera solution

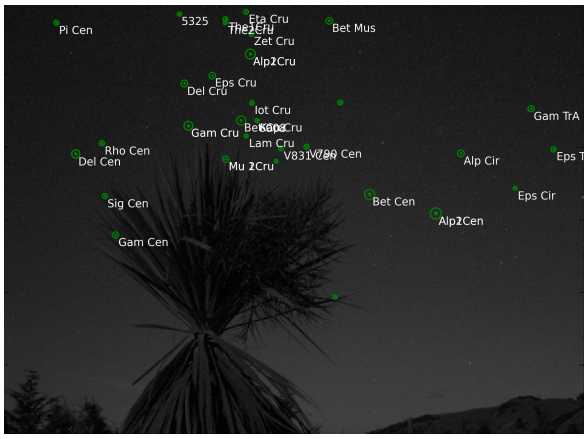


(d) Close-up of the solution using distortions

Figure 4.7: Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars.



(a) Solution using an ideal camera model



(b) Solution using model with distortions

Figure 4.8: Star names as described in the BSC (21) for the stars identified in **Figure 4.6**.

Table 4.3 and **Table 4.4** display the results of the first and second identifications.

Table 4.3: Matches found for **Figure 4.6** using the model **without distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0.61	5267	Bet Cen	(103.3.34.3)	(104.3.34.8)	1.1
1.25	4853	Bet Cru	(-72.7.136.5)	(-72.5.136.8)	0.3
1.33	5460	Alp2Cen	(195.0.7.9)	(195.6.8.6)	0.9
1.63	4763	Gam Cru	(-145.0.129.3)	(-145.1.129.5)	0.2
1.73	4731	Alp2Cru	(-60.1.228.2)	(-59.7.228.2)	0.4
2.17	4819	Gam Cen	(-244.7.-20.5)	(-245.3.-20.8)	0.6
2.6	4621	Del Cen	(-299.3.90.8)	(-300.1.90.7)	0.8
2.8	4656	Del Cru	(-150.7.187.7)	(-150.6.187.8)	0.1
2.89	5671	Gam TrA	(328.5.153.7)	(326.5.152.9)	2.1
3.05	4844	Bet Mus	(48.3.274.7)	(48.8.274.0)	0.9
3,19	5463	Alp Cir	(230.2,91.2)	(230.1,91.4)	0,2
3,59	4700	Eps Cru	(-112.3,198.4)	(-112.1,198.6)	0,2
3,89	4390	Pi Cen	(-326.9,272.1)	(-326.8,271.5)	0,6
3,91	4743	Sig Cen	(-259.8,32.6)	(-260.1,32.5)	0,3
3,96	4638	Rho Cen	(-263.7,105.5)	(-264.1,105.5)	0,4
4,03	4898	Mu 1Cru	(-93.8,83.2)	(-93.6,83.4)	0,3
4,04	4679	Zet Cru	(-58.9,256.6)	(-58.5,256.2)	0,6
4,11	4522		(-157.7,284.0)	(-157.4,283.5)	0,6
4,11	5771	Eps TrA	(360.0,97.0)	(357.6,96.9)	2,4
4,15	4616	Eta Cru	(-66.3,286.9)	(-65.8,286.4)	0,7
4,33	4599	The1Cru	(-94.5,276.6)	(-94.0,276.3)	0,5
4,53	5035		(17.0,100.3)	(17.7,100.7)	0,8
4,53	5041		(63.4,161.3)	(64.0,161.5)	0,6
4,6	4975		(-18.4,97.8)	(-17.8,98.1)	0,7
4,62	4897	Lam Cru	(-66.2,115.1)	(-65.6,115.5)	0,6
4,69	4842	Iot Cru	(-58.0,160.9)	(-57.6,161.2)	0,5
4,72	4603	The2Cru	(-95.0,271.9)	(-94.5,271.5)	0,7
4,75	5297		(55.4,-107.3)	(56.5,-106.5)	1,4
4,86	5666	Eps Cir	(305.6,43.0)	(304.6,43.5)	1,1
4,92	4989		(-24.7,80.6)	(-24.4,80.6)	0,3
5,9	4890	Kap Cru	(-50.6,136.6)	(-50.6,136.6)	0

31 stars were identified while using the model without distortions and 40 stars were identified while using the model with distortions. Using the model with distortions increased the amount of identified stars and reduced the difference between the expected position and the position in which the stars were detected as shown in **Table 4.3** and in **Table 4.4**.

Among the identified stars are the Southern Cross stars (abbreviated as "Cru") and the stars of the neighboring constellations Centaurus "Cen", Circinus "Cir", Austral Triangle "TrA" and Musca "Mus".

Even though **Figure 4.6** has a partially obstructed view of the sky, it was possible to:

- estimate the camera distortion parameters
- individually identify stars
- identify a region of the sky

Table 4.4: Matches found for **Figure 4.6** using the model **with distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
-0.01	5459	Alp1Cen	(195.3.7.9)	(195.6.8.6)	0.7
0.61	5267	Bet Cen	(103.6.34.4)	(104.3.34.8)	0.8
1.25	4853	Bet Cru	(-72.9.136.8)	(-72.5.136.8)	0.4
1.33	4730	Alp1Cru	(-60.1.228.3)	(-59.7.228.2)	0.4
1.33	5460	Alp2Cen	(195.3.7.9)	(195.6.8.6)	0.7
1.63	4763	Gam Cru	(-145.3.129.5)	(-145.1.129.5)	0.2
1.73	4731	Alp2Cru	(-60.1.228.3)	(-59.7.228.2)	0.4
2.17	4819	Gam Cen	(-244.8.-20.5)	(-245.3.-20.8)	0.6
2.6	4621	Del Cen	(-298.7.90.7)	(-300.1.90.7)	1.4
2.8	4656	Del Cru	(-150.7.187.7)	(-150.6.187.8)	0.1
2.89	5671	Gam TrA	(327.6.153.3)	(326.5.152.9)	1.1
3.05	4844	Bet Mus	(48.3.274.5)	(48.8.274.0)	0.7
3.19	5463	Alp Cir	(230.2.91.3)	(230.1.91.4)	0.2
3.59	4700	Eps Cru	(-112.4.198.6)	(-112.1.198.6)	0.3
3.89	4390	Pi Cen	(-325.9.271.3)	(-326.8.271.5)	1
3.91	4743	Sig Cen	(-259.7.32.5)	(-260.1.32.5)	0.4
3.96	4638	Rho Cen	(-263.5.105.4)	(-264.1.105.5)	0.6
4.03	4898	Mu 1Cru	(-94.0.83.4)	(-93.6.83.4)	0.5
4.04	4679	Zet Cru	(-58.8.256.5)	(-58.5.256.2)	0.5
4.11	4522	Eta Cru	(-157.3.283.4)	(-157.4.283.5)	0.1
4.11	5771	Eps TrA	(358.9.96.7)	(357.6.96.9)	1.4
4.15	4616	Eta Cru	(-66.2.286.5)	(-65.8.286.4)	0.4
4.33	4599	The1Cru	(-94.3.276.2)	(-94.0.276.3)	0.4
4.47	4618		(-300.4.90.5)	(-300.1.90.7)	0.3
4.53	5035	Lam Cru	(17.1.100.5)	(17.7.100.7)	0.6
4.53	5041	Iot Cru	(63.5.161.6)	(64.0.161.5)	0.5
4.6	4975	The2Cru	(-18.5.98.0)	(-17.8.98.1)	0.7
4.62	4897	Lam Cru	(-66.3.115.4)	(-65.6.115.5)	0.7
4.69	4842	Iot Cru	(-58.2.161.2)	(-57.6.161.2)	0.6
4.72	4603	The2Cru	(-94.9.271.6)	(-94.5.271.5)	0.4
4.75	5297	Kap Cru	(55.5.-107.5)	(56.5.-106.5)	1.5
4.86	5666	Eps Cir	(305.0.43.0)	(304.6.43.5)	0.7
4.92	4989		(-24.7.80.8)	(-24.4.80.6)	0.4
5.17	4899	Mu 2Cru	(-94.1.83.2)	(-93.6.83.4)	0.6
5.31	5048		(65.2.159.1)	(64.0.161.5)	2.7
5.76	4887		(-52.3.137.0)	(-50.6.136.6)	1.7
5.9	4890	Kap Cru	(-50.7.137.0)	(-50.6.136.6)	0.4
6.18	5034		(16.8.100.3)	(17.7.100.7)	1
6.37	4619		(-298.7.92.0)	(-300.1.90.7)	1.9
6.42	4764	Gam Cru	(-145.5.128.8)	(-145.1.129.5)	0.8

- validate the distortions calculation by the decrease of the error and increase of the number of identified stars.
- validate the identification result by visual inspection of the image.
- validate the names and relative position of stars using information available in (54).
- validate the identified region of the sky using the information available in (54).

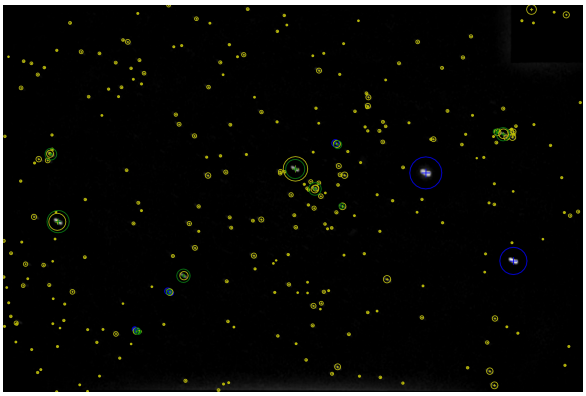
Figure 4.9: Taurus constellation, long exposure, obtained from Starnamer's blog (63). The cluster of stars to the right of the image is the Pleiades globular cluster.

4.1.3 Test with blurred stars

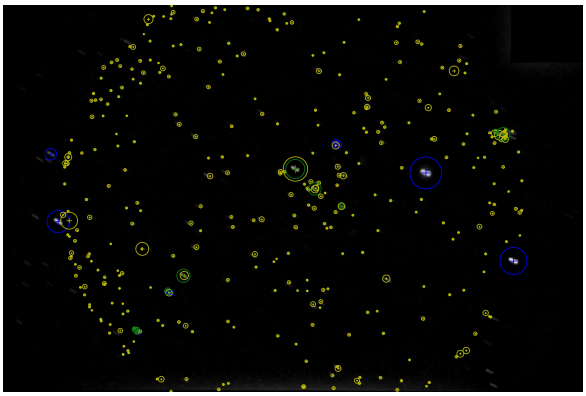
If the camera rotates in relation to the sky frame of reference, which can happen when using long exposure times, the stars in the image will be displayed as streaks instead of dots. **Figure 4.9** is an example taken from (63) which illustrates the constellation of Taurus.

Figure 4.11 display the results of processing **Figure 4.10** using the lens model without distortions and with distortions.

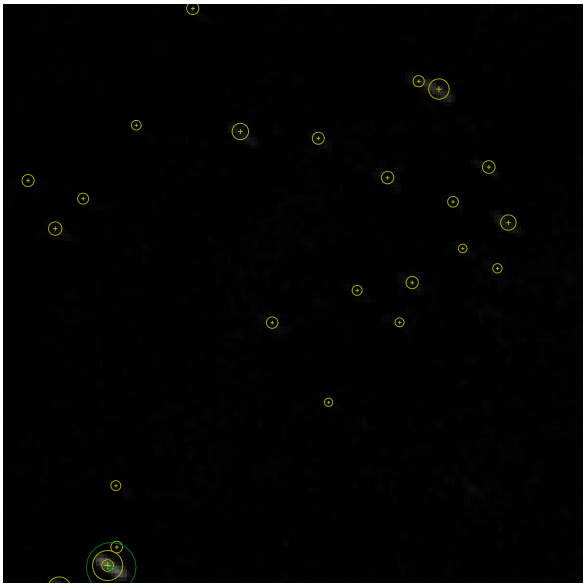
Figure 4.10: Right upper corner after application of Gimp's (30) difference of gaussians filter on **Figure 4.9**



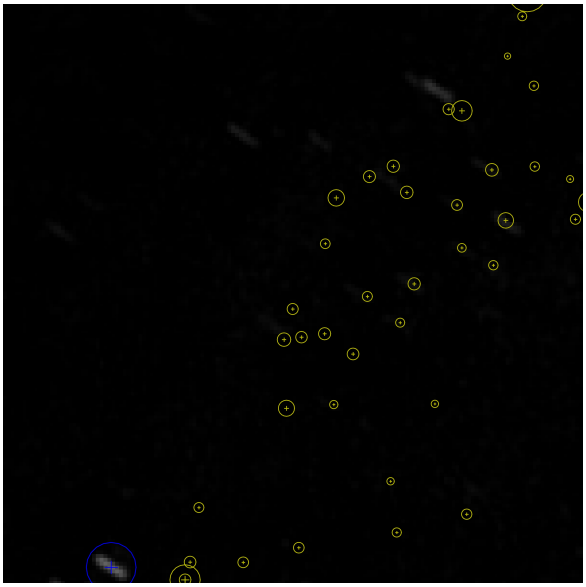
(a) Solution using an ideal camera model



(b) Solution using model with distortions

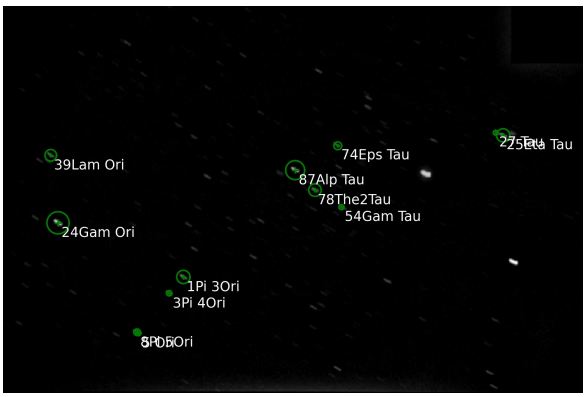


(c) Close-up of the ideal camera solution



(d) Close-up of the solution using distortions

Figure 4.11: Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars.



(a) Solution using an ideal camera model

(b) Solution using model with distortions

Figure 4.12: Star names as described in the BSC (21) for the stars identified in **Figure 4.11**.

Table 4.5 and **Table 4.6** display the results of the first and second identifications.

Table 4.5: Matches found for **Figure 4.10** using the model **without distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0.85	1457	87Alp Tau	(2.7.32.8)	(2.5.32.7)	0.3
1.64	1790	24Gam Ori	(-260.0.-25.6)	(-259.0.-25.3)	0.9
2.87	1165	25Eta Tau	(232.3.71.5)	(231.8.71.1)	0.6
3.19	1543	1Pi 3Ori	(-120.2.-85.0)	(-120.8.-85.0)	0.6
3.4	1412	78The2Tau	(24.2.10.5)	(24.3.10.7)	0.2
3.53	1409	74Eps Tau	(48.4.60.3)	(49.4.59.6)	1.3
3.54	1879	39Lam Ori	(-267.9.49.5)	(-267.0.49.1)	1
3.63	1178	27 Tau	(225.7.73.2)	(223.6.74.0)	2.2
3.65	1346	54Gam Tau	(54.7.-8.4)	(53.5.-8.1)	1.3
3.69	1552	3Pi 4Ori	(-136.0.-103.0)	(-136.5.-103.0)	0.5
3.72	1567	8Pi 5Ori	(-172.7.-146.1)	(-171.9.-145.9)	0.8
5.33	1562	5 Ori	(-168.8.-146.7)	(-170.5.-146.9)	1.7

Table 4.6: Matches found for **Figure 4.10** using the model **with distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0,85	1457	87Alp Tau	(2.7,32.7)	(2.5,32.7)	0,2
3,19	1543	1Pi 3Ori	(-119.9,-84.7)	(-120.8,-85.0)	0,9
3,4	1412	78The2Tau	(24.3,10.6)	(24.3,10.7)	0,2
3,63	1178	27 Tau	(222.6,72.2)	(223.6,74.0)	2,1
3,65	1346	54Gam Tau	(53.5,-8.3)	(53.5,-8.1)	0,2
3,69	1552	3Pi 4Ori	(-136.8,-103.7)	(-138.0,-102.0)	2
3,72	1567	8Pi 5Ori	(-172.2,-145.6)	(-173.5,-145.0)	1,4
5,09	1180	28 Tau	(222.8,73.4)	(223.6,74.0)	1
5,33	1562	5 Ori	(-168.6,-146.5)	(-171.9,-145.9)	3,4
6,05	1646		(-174.8,-142.9)	(-174.5,-144.1)	1,2
6,43	1152	22 Tau	(232.3,73.6)	(231.8,71.1)	2,6

12 stars were identified while using the model without distortions and 11 stars were identified while using the model with distortions. The model without distortions produced a consistent result in all the field of view, predicting accurately the position of stars that were not detected by the segmentation algorithm. The model with distortions produced a slightly better result regarding the errors, but the estimated distortion was so high that made a few stars that should be projected further from the center than other stars to be projected nearer to the center instead, as if the edges of the star field had folded upon themselves. Such distortion made all stars from the catalog be projected inside the circle clearly seen on **Figure 4.11**.

This result may have been caused by the low accuracy in centroid measurement of the blurred stars and the reduced number of stars. These factors make it easier for the algorithm to converge to a local minimum of the cost function which has no real world meaning in the attempt to reduce the error. That tendency is smaller when centroids are detected with high accuracy because a good global minimum for the cost function exists, making it more unlikely that the algorithm converges to a local minimum with a worse result.

Among the identified stars in the model without distortion are the Taurus constellation stars, which matches the image description.

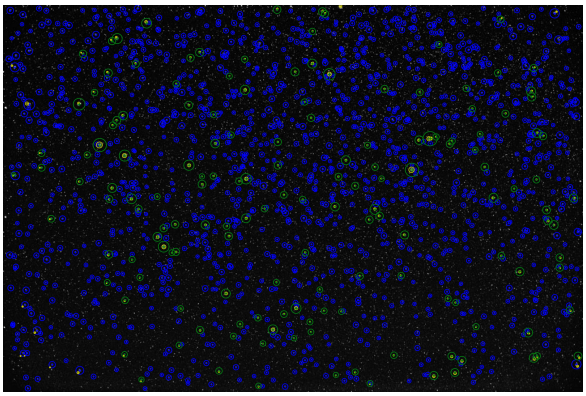
Even though the original image in **Figure 4.9** displays foreign objects, blurred stars and a strong vignetting effect and does not have enough contrast for star detection, the processed version in **Figure 4.10** was enough to perform identification, when using the model without distortions.

4.1.4 Test with high quality image

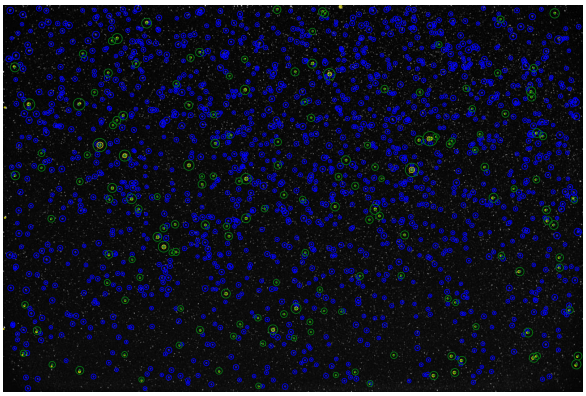
Previous tests involve situations that make it difficult to identify stars. This test explores how the algorithm works when presented with very high quality images, with low noise, low distortion and a large amount of stars. The example on **Figure 4.13** taken from (9) illustrates the Capricornus constellation.

Figure 4.14 display the results of processing **Figure 4.13** using the lens model without distortions and with distortions.

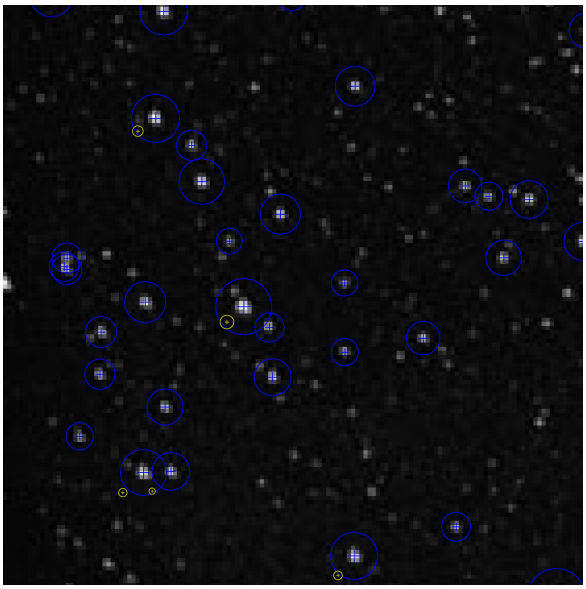
Figure 4.13: High quality picture of the Capricornus constellation obtained from (9).



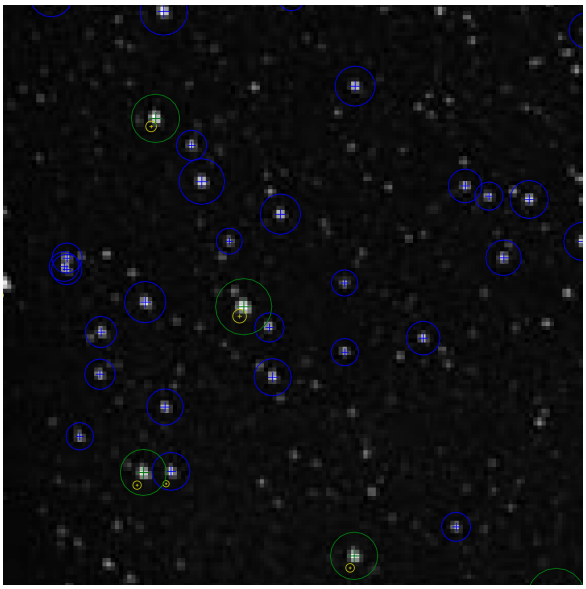
(a) Solution using an ideal camera model



(b) Solution using model with distortions

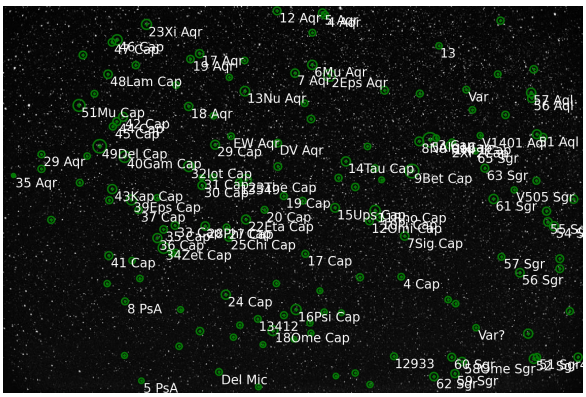


(c) Close-up of the ideal camera solution

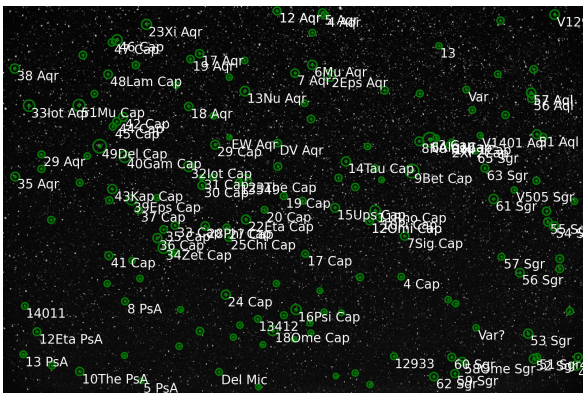


(d) Close-up of the solution using distortions

Figure 4.14: Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars.



(a) Solution using an ideal camera model



(b) Solution using model with distortions

Figure 4.15: Star names as described in the BSC (21) for the stars identified in **Figure 4.14**.

Table 4.7 and **Table 4.8** display the results of the first and second identifications.

Table 4.7: Matches found for **Figure 4.13** using the model **without distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
2,87	8322	49Del Cap	(-341.9,94.7)	(-341.1,94.6)	0,8
3,08	7776	9Bet Cap	(209.4,51.0)	(210.2,50.7)	0,9
3,68	8278	40Gam Cap	(-297.5,76.0)	(-297.1,76.0)	0,5
3,74	8204	34Zet Cap	(-228.0,-85.0)	(-228.3,-84.7)	0,5
3,77	7950	2Eps Aqr	(64.4,219.8)	(64.3,219.5)	0,3
4,07	8075	23The Cap	(-82.4,35.0)	(-82.8,35.1)	0,4
4,11	7980	18Ome Cap	(-35.3,-231.1)	(-35.5,-231.1)	0,2
4,14	7936	16Psi Cap	(5.5,-193.2)	(5.4,-193.7)	0,5
4,24	7747	5Alp1Cap	(243.1,106.2)	(242.0,105.8)	1,2
4,28	8167	32Iot Cap	(-183.4,58.9)	(-183.8,59.0)	0,4
4,5	8080	24 Cap	(-118.0,-167.3)	(-118.5,-167.5)	0,5
4,51	8093	13Nu Aqr	(-84.3,191.9)	(-84.8,191.8)	0,5
4,51	8213	36 Cap	(-238.9,-67.7)	(-239.3,-67.5)	0,5
4,52	7604	59 Sgr	(286.2,-308.1)	(286.0,-306.8)	1,3
4,58	7650	62 Sgr	(248.7,-313.3)	(248.8,-312.1)	1,3
4,6	7440	52 Sgr	(426.4,-282.9)	(425.3,-280.8)	2,4
4,68	8260	39Eps Cap	(-285.2,-0.5)	(-285.1,-0.3)	0,3
4,69	8264	23Xi Aqr	(-259.5,312.1)	(-258.5,309.7)	2,6
4,7	7597	58Ome Sgr	(298.4,-286.6)	(298.5,-285.4)	1,2
4,73	7990	6Mu Aqr	(34.4,238.6)	(34.2,238.2)	0,4
4,73	8288	43Kap Cap	(-319.5,18.7)	(-319.0,18.9)	0,5
4,76	7773	8Nu Cap	(222.4,103.3)	(223.0,103.2)	0,7
4,78	7822	11Rho Cap	(144.6,-18.2)	(145.0,-18.9)	0,8
4,83	7618	60 Sgr	(280.3,-278.8)	(280.3,-277.8)	0,9
4,84	8060	22Eta Cap	(-82.6,-34.6)	(-82.9,-34.7)	0,4
4,86	7515	56 Sgr	(400.6,-129.2)	(400.6,-128.7)	0,5
5,02	7614	61 Sgr	(353.2,1.4)	(354.2,1.4)	1
5,06	7489	55 Sgr	(450.6,-39.3)	(450.0,-39.0)	0,6
5,08	8351	51Mu Cap	(-378.4,168.4)	(-377.6,166.7)	2
5,09	8311	46 Cap	(-312.0,284.6)	(-310.4,282.4)	2,7
5,1	7900	15Ups Cap	(74.0,-13.9)	(74.3,-14.1)	0,5
5,18	8283	42 Cap	(-300.7,147.0)	(-299.8,146.5)	1
5,22	7889	14Tau Cap	(93.4,67.8)	(93.5,67.8)	0,2
5,24	8127	28Phi Cap	(-154.4,-46.3)	(-155.0,-46.2)	0,6
5,24	8285	41 Cap	(-326.0,-99.4)	(-325.0,-98.6)	1,3
5,25	7814	10Pi Cap	(152.1,-30.3)	(153.0,-30.6)	0,9
5,28	7761	7Sig Cap	(196.5,-63.7)	(197.2,-64.0)	0,8
5,28	8128	29 Cap	(-136.8,97.3)	(-137.4,97.2)	0,5

Table 4.7 Continued:

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
5,3	8087	25Chi Cap	(-112.9,-66.0)	(-113.4,-66.2)	0,5
5,39	7553	51 Aql	(431.3,116.4)	(430.6,115.8)	0,9
5,41	8183	33 Cap	(-207.8,-45.4)	(-208.2,-45.3)	0,5
5,42	8110		(-164.0,-232.4)	(-164.2,-231.8)	0,6
5,43	7363	49Chi3Sgr	(505.4,-280.5)	(503.3,-277.7)	3,5
5,43	8137	30 Cap	(-160.0,25.4)	(-160.5,25.5)	0,5
5,47	7454		(496.9,1.2)	(495.4,1.2)	1,5
5,49	7496		(448.2,-20.2)	(447.7,-20.1)	0,5
5,49	8187	18 Aqr	(-183.7,164.9)	(-184.1,164.8)	0,4
5,51	8015	7 Aqr	(4.0,223.9)	(3.6,223.7)	0,4
5,55	7985	5 Aqr	(52.6,331.5)	(52.1,329.9)	1,7
5,58	8319	48Lam Cap	(-327.8,223.2)	(-326.4,221.7)	2,1
5,6	8172		(-206.2,-94.2)	(-206.6,-93.9)	0,5
5,64	7443		(472.1,-104.6)	(471.1,-104.0)	1,2
5,65	7431	51 Sgr	(431.9,-279.8)	(430.9,-277.6)	2,4
5,65	7845		(161.6,193.6)	(161.7,193.3)	0,3
5,68	8070	Del Mic	(-130.7,-305.3)	(-130.6,-304.0)	1,4
5,69	8245	37 Cap	(-272.6,-18.3)	(-272.5,-18.0)	0,3
5,7	8013		(-61.8,-210.1)	(-62.0,-210.3)	0,3
5,7	8195	19 Aqr	(-180.9,249.4)	(-180.9,248.5)	0,9
5,71	7649	63 Sgr	(338.4,55.9)	(338.7,55.8)	0,4
5,73	7722		(178.0,-276.5)	(178.2,-276.3)	0,4
5,73	8253	8 PsA	(-297.1,-180.5)	(-296.5,-179.3)	1,4
5,78	8000	19 Cap	(-16.0,6.8)	(-16.1,6.6)	0,3
5,78	8207	35 Cap	(-228.0,-52.5)	(-228.4,-52.3)	0,4
5,79	7584	56 Aql	(422.9,180.1)	(421.5,179.3)	1,6
5,8	7905		(80.1,38.8)	(80.5,38.7)	0,4
5,81	8439	35 Aqr	(-494.3,40.5)	(-493.4,43.0)	2,6
5,85	7715	2Xi 2Cap	(276.3,96.7)	(276.8,96.5)	0,6
5,85	7801		(109.7,-306.6)	(109.9,-305.8)	0,7
5,86	7961		(-15.6,-203.4)	(-15.9,-203.7)	0,4
5,87	7748	4 Cap	(190.4,-135.5)	(191.1,-135.8)	0,8
5,87	8018		(-26.3,58.7)	(-26.6,58.7)	0,3
5,88	7637		(375.6,150.5)	(375.2,149.9)	0,7
5,88	7976		(31.9,142.7)	(31.9,142.6)	0,1
5,88	8295	44 Cap	(-311.6,138.6)	(-311.0,138.1)	0,8
5,92	7561	57 Sgr	(367.8,-100.8)	(368.1,-100.6)	0,3
5,93	7937	17 Cap	(21.6,-94.9)	(21.8,-95.4)	0,5
5,97	7473		(415.2,-237.5)	(415.3,-236.3)	1,2
5,99	7982	4 Aqr	(56.9,327.5)	(56.4,326.0)	1,5

Table 4.7 Continued:

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
5,99	8175	17 Aqr	(-164.5,258.9)	(-164.8,258.0)	1
5,99	8302	45 Cap	(-318.7,129.9)	(-318.0,129.4)	0,8
6	8318	47 Cap	(-321.2,280.3)	(-319.5,278.0)	2,9
6,01	7643		(286.8,-183.4)	(287.3,-183.3)	0,5
6,01	8024		(-26.6,100.0)	(-26.9,100.0)	0,3
6,04	7541		(442.5,111.0)	(441.5,110.5)	1,1
6,04	8122		(-140.1,39.9)	(-140.7,39.8)	0,7
6,05	8045		(-93.3,-221.4)	(-93.7,-221.3)	0,4
6,08	8273		(-278.1,239.1)	(-277.4,237.7)	1,6
6,11	7439		(471.8,-122.2)	(471.0,-121.7)	0,9
6,11	7532		(430.1,33.8)	(429.8,33.6)	0,4
6,12	7837		(144.4,55.9)	(145.0,55.7)	0,6
6,12	8378		(-429.4,-36.3)	(-426.8,-35.5)	2,7
6,13	7410		(490.2,-198.2)	(488.8,-196.6)	2,2
6,13	7855		(131.2,94.2)	(131.7,94.2)	0,5
6,16	7825		(117.8,-136.2)	(118.3,-136.7)	0,7
6,16	8346		(-378.0,30.1)	(-376.4,30.2)	1,6
6,17	8083		(-94.4,30.8)	(-95.0,30.7)	0,6
6,18	7578		(323.0,-226.5)	(323.2,-225.8)	0,7
6,18	7694		(316.7,159.2)	(316.8,158.7)	0,5
6,19	7865		(109.4,22.9)	(110.0,22.7)	0,6
6,2	7476	54 Sgr	(461.4,-46.7)	(460.6,-46.5)	0,9

Table 4.8: Matches found for **Figure 4.13** using the model **with distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
2,87	8322	49Del Cap	(-340.8,94.4)	(-341.1,94.6)	0,3
3,08	7776	9Bet Cap	(209.9,50.9)	(210.2,50.7)	0,3
3,57	7754	6Alp2Cap	(240.6,105.6)	(242.0,105.8)	1,5
3,68	8278	40Gam Cap	(-297.0,75.8)	(-297.1,76.0)	0,2
3,74	8204	34Zet Cap	(-227.9,-85.2)	(-228.3,-84.7)	0,6
3,77	7950	2Eps Aqr	(64.8,219.8)	(64.3,219.5)	0,6
4,07	8075	23The Cap	(-82.5,35.0)	(-82.8,35.1)	0,3
4,11	7980	18Ome Cap	(-35.1,-231.5)	(-35.5,-231.1)	0,5
4,14	7936	16Psi Cap	(5.7,-193.7)	(5.4,-193.7)	0,3
4,24	7747	5Alp1Cap	(243.4,106.0)	(242.0,105.8)	1,4
4,27	8418	33Iot Aqr	(-466.7,166.7)	(-465.7,166.2)	1,1
4,28	8167	32Iot Cap	(-183.6,58.9)	(-183.8,59.0)	0,3

Table 4.8 Continued:

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
4,5	8080	24 Cap	(-118.1,-167.8)	(-118.5,-167.5)	0,5
4,51	8093	13Nu Aqr	(-84.2,192.1)	(-84.8,191.8)	0,6
4,51	8213	36 Cap	(-238.9,-67.9)	(-239.3,-67.5)	0,6
4,52	7604	59 Sgr	(285.1,-307.0)	(286.0,-306.8)	0,8
4,58	7650	62 Sgr	(247.9,-312.4)	(248.8,-312.1)	0,9
4,6	7440	52 Sgr	(423.9,-281.4)	(425.3,-280.8)	1,5
4,68	8260	39Eps Cap	(-284.9,-0.6)	(-285.1,-0.3)	0,4
4,69	8264	23Xi Aqr	(-258.2,310.8)	(-258.5,309.7)	1,1
4,7	7597	58Ome Sgr	(297.4,-285.7)	(298.5,-285.4)	1,1
4,73	7990	6Mu Aqr	(34.7,238.6)	(34.2,238.2)	0,6
4,73	8288	43Kap Cap	(-318.9,18.6)	(-319.0,18.9)	0,4
4,76	7773	8Nu Cap	(222.8,103.2)	(223.0,103.2)	0,3
4,78	7822	11Rho Cap	(145.2,-18.4)	(145.0,-18.9)	0,5
4,83	7618	60 Sgr	(279.5,-278.0)	(280.3,-277.8)	0,9
4,84	8060	22Eta Cap	(-82.6,-34.9)	(-82.9,-34.7)	0,3
4,86	7515	56 Sgr	(399.1,-128.8)	(400.6,-128.7)	1,6
5,01	8326	10The PsA	(-377.1,-305.4)	(-376.4,-303.0)	2,5
5,02	7614	61 Sgr	(352.6,1.1)	(354.2,1.4)	1,6
5,03	7362	47Chi1Sgr	(499.3,-294.0)	(500.5,-292.4)	2,1
5,06	7489	55 Sgr	(448.5,-39.3)	(450.0,-39.0)	1,5
5,08	8351	51Mu Cap	(-376.6,167.6)	(-377.6,166.7)	1,4
5,09	8311	46 Cap	(-310.4,283.2)	(-310.4,282.4)	0,8
5,1	7900	15Ups Cap	(74.4,-14.1)	(74.3,-14.1)	0,1
5,18	8283	42 Cap	(-299.9,146.6)	(-299.8,146.5)	0,1
5,22	7889	14Tau Cap	(93.9,67.8)	(93.5,67.8)	0,4
5,24	8127	28Phi Cap	(-154.6,-46.6)	(-155.0,-46.2)	0,5
5,24	8285	41 Cap	(-325.1,-99.3)	(-325.0,-98.6)	0,8
5,25	7814	10Pi Cap	(152.8,-30.6)	(153.0,-30.6)	0,2
5,28	7761	7Sig Cap	(197.0,-64.0)	(197.2,-64.0)	0,2
5,28	8128	29 Cap	(-137.0,97.4)	(-137.4,97.2)	0,5
5,3	8087	25Chi Cap	(-113.0,-66.4)	(-113.4,-66.2)	0,4
5,39	7553	51 Aql	(429.5,115.6)	(430.6,115.8)	1,2
5,41	8183	33 Cap	(-207.9,-45.6)	(-208.2,-45.3)	0,5
5,42	8110		(-163.7,-232.5)	(-164.2,-231.8)	0,8
5,42	8386	12Eta PsA	(-452.9,-235.1)	(-451.9,-232.8)	2,5
5,43	7363	49Chi3Sgr	(502.1,-278.9)	(503.3,-277.7)	1,6
5,43	8137	30 Cap	(-160.2,25.3)	(-160.5,25.5)	0,4
5,46	8452	38 Aqr	(-492.7,233.1)	(-491.1,231.7)	2,1
5,47	7454		(494.1,0.9)	(497.0,-2.7)	4,6
5,49	7496		(446.2,-20.3)	(447.7,-20.1)	1,6

Table 4.8 Continued:

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
5,49	8187	18 Aqr	(-183.6,164.9)	(-184.1,164.8)	0,5
5,51	8015	7 Aqr	(4.2,224.1)	(3.6,223.7)	0,7
5,55	7985	5 Aqr	(52.8,330.7)	(52.1,329.9)	1,1
5,58	8319	48Lam Cap	(-326.3,222.3)	(-326.4,221.7)	0,7
5,6	8172		(-206.3,-94.4)	(-206.6,-93.9)	0,6
5,64	7443		(469.6,-104.2)	(471.1,-104.0)	1,5
5,65	7431	51 Sgr	(429.4,-278.3)	(430.9,-277.6)	1,6
5,65	7575		(462.6,327.5)	(462.0,327.0)	0,7
5,65	7845		(162.0,193.5)	(161.7,193.3)	0,3
5,68	8070	Del Mic	(-130.3,-304.9)	(-130.6,-304.0)	1
5,69	8245	37 Cap	(-272.4,-18.4)	(-272.5,-18.0)	0,4
5,7	8013		(-61.7,-210.6)	(-62.0,-210.3)	0,4
5,7	8195	19 Aqr	(-180.4,249.0)	(-180.9,248.5)	0,7
5,71	7593	57 Aql	(419.8,188.9)	(420.5,189.1)	0,6
5,71	7649	63 Sgr	(337.9,55.6)	(338.7,55.8)	0,9
5,73	7722		(177.8,-276.3)	(178.2,-276.3)	0,4
5,73	8253	8 PsA	(-296.3,-180.2)	(-296.5,-179.3)	1
5,78	8000	19 Cap	(-15.8,6.7)	(-16.1,6.6)	0,3
5,78	8207	35 Cap	(-228.0,-52.7)	(-228.4,-52.3)	0,5
5,79	7584	56 Aql	(420.9,178.9)	(421.5,179.3)	0,7
5,8	7905		(80.6,38.8)	(80.5,38.7)	0,2
5,81	8439	35 Aqr	(-491.1,40.1)	(-490.3,40.9)	1,1
5,85	7715	2Xi 2Cap	(276.4,96.4)	(276.8,96.5)	0,5
5,85	7801		(109.7,-306.3)	(109.9,-305.8)	0,4
5,86	7961		(-15.4,-204.0)	(-15.9,-203.7)	0,6
5,87	7748	4 Cap	(190.8,-135.8)	(191.1,-135.8)	0,3
5,87	8018		(-26.2,58.7)	(-26.6,58.7)	0,5
5,88	7637		(374.4,149.7)	(375.2,149.9)	0,8
5,88	7976		(32.3,142.9)	(31.9,142.6)	0,4
5,88	8295	44 Cap	(-310.7,138.2)	(-311.0,138.1)	0,3
5,89	8059	12 Aqr	(-27.3,334.5)	(-28.0,333.5)	1,2
5,92	7561	57 Sgr	(366.8,-100.7)	(368.1,-100.6)	1,3
5,93	7937	17 Cap	(21.9,-95.4)	(21.8,-95.4)	0,1
5,94	7830	12Omi Cap	(135.0,-37.2)	(135.0,-37.3)	0,1
5,96	8408		(-474.3,-189.3)	(-473.2,-187.4)	2,2
5,97	7473		(413.1,-236.4)	(415.3,-236.3)	2,3
5,99	7982	4 Aqr	(57.0,326.7)	(56.4,326.0)	0,9
5,99	8175	17 Aqr	(-164.0,258.5)	(-164.8,258.0)	0,9
5,99	8302	45 Cap	(-317.8,129.5)	(-318.0,129.4)	0,2
6	8318	47 Cap	(-319.5,278.9)	(-319.5,278.0)	1

Table 4.8 Continued:

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
6,01	7643		(286.4,-183.3)	(287.3,-183.3)	0,9
6,01	8024		(-26.5,100.2)	(-26.9,100.0)	0,5
6,04	7541		(440.5,110.2)	(441.5,110.5)	1,1
6,04	8122		(-140.3,39.8)	(-140.7,39.8)	0,5
6,05	8045		(-93.2,-221.7)	(-93.7,-221.3)	0,6
6,08	8273		(-277.1,238.3)	(-277.4,237.7)	0,7
6,11	7439		(469.2,-121.8)	(471.0,-121.7)	1,8
6,11	7532		(428.4,33.4)	(429.8,33.6)	1,4

154 stars were identified while using the model without distortions and 169 stars were identified while using the model with distortions. Using the model with distortions increased the amount of identified stars and reduced the difference between the expected position and the position in which the stars were detected as shown in **Table 4.7** and in **Table 4.8**.

Among the identified stars are the stars from the constellation of Capricornus (abbreviated as "Cap").

4.1.5 Test with few stars

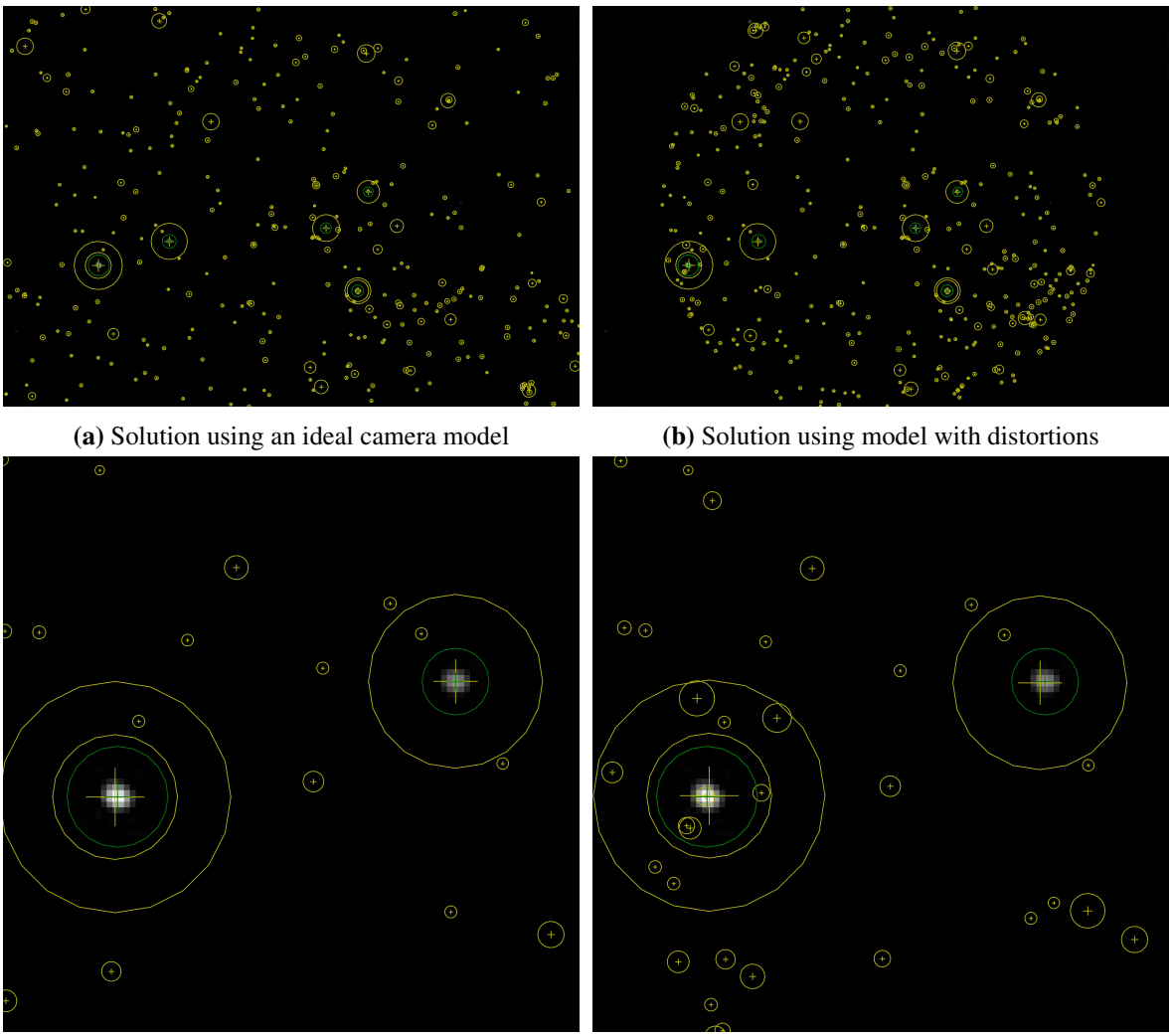
Figure 4.16 presents light pollution, and the fainter stars become difficult to detect. The image was preprocessed with the Gimp (30) software to increase contrast by applying a difference of gaussians high pass filter.

This image was chosen to test the algorithm's capacity to find correct solutions with few stars, avoiding ambiguities in the solution. According to (63), the image displays the constellations of Centaurus and the Southern Cross.

Figure 4.18 display the results of processing **Figure 4.17** using the lens model without distortions and with distortions.

Figure 4.16: Image of Centaurus and the Southern Cross, with heavy light pollution. Obtained at (63).

Figure 4.17: Result of applying Gimp's (30) difference of gaussians filter on **Figure 4.16**



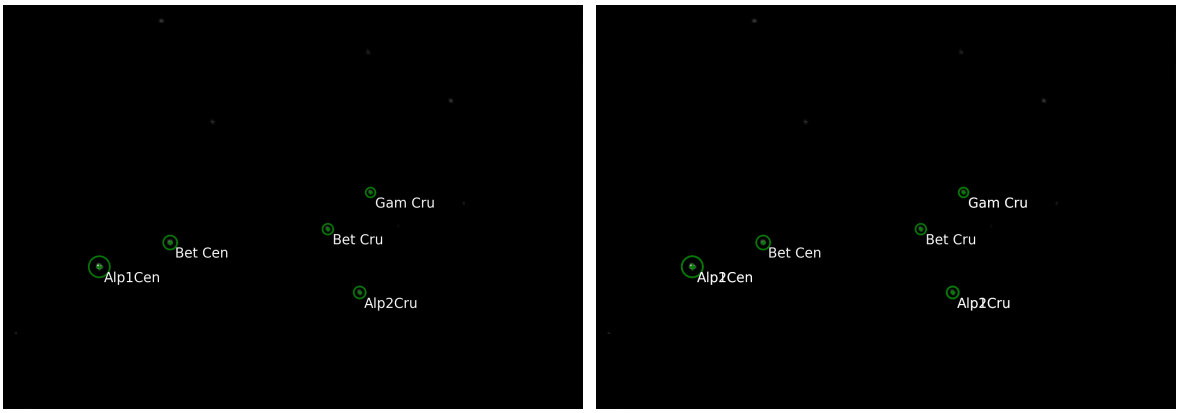
(a) Solution using an ideal camera model

(b) Solution using model with distortions

(c) Close-up of the ideal camera solution

(d) Close-up of the solution using distortions

Figure 4.18: Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars.



(a) Solution using an ideal camera model

(b) Solution using model with distortions

Figure 4.19: Star names as described in the BSC (21) for the stars identified in **Figure 4.18**.

Table 4.9 and **Table 4.10** display the results of the first and second identifications.

Table 4.9: Matches found for **Figure 4.17** using the model **without distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
-0,01	5459	Alp1Cen	(-160.6,-49.2)	(-160.1,-49.1)	0,4
0,61	5267	Bet Cen	(-101.5,-29.1)	(-101.5,-29.1)	0,1
1,25	4853	Bet Cru	(29.0,-18.0)	(28.9,-18.1)	0,2
1,63	4763	Gam Cru	(64.2,12.2)	(64.2,12.3)	0,1
1,73	4731	Alp2Cru	(55.4,-70.3)	(55.2,-70.4)	0,2

Table 4.10: Matches found for **Figure 4.17** using the model **with distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
-0,01	5459	Alp1Cen	(-159.8,-48.9)	(-160.1,-49.1)	0,4
0,61	5267	Bet Cen	(-102.4,-29.3)	(-101.5,-29.1)	0,9
1,25	4853	Bet Cru	(29.1,-18.1)	(28.9,-18.1)	0,2
1,33	4730	Alp1Cru	(55.5,-70.3)	(55.2,-70.4)	0,3
1,33	5460	Alp2Cen	(-159.8,-48.9)	(-160.1,-49.1)	0,4
1,63	4763	Gam Cru	(63.6,12.1)	(64.2,12.3)	0,6
1,73	4731	Alp2Cru	(55.5,-70.3)	(55.2,-70.4)	0,2
4,86	4729		(55.5,-70.7)	(55.2,-70.4)	0,4
5,46	5680		(-160.4,-49.1)	(-160.1,-49.1)	0,3
6,42	4764	Gam Cru	(63.5,12.5)	(64.2,12.3)	0,7

This test shows the identification of a sky region using only the 5 stars detected by the segmentation algorithm. It is an important result because it empirically explores the limits for a successful identification to be made with an unknown camera in unknown conditions. The other result from this test was that the distortion calculation diverged, just like the previous test with blurred stars, and the projection of the stars seemed to fold over itself. This has not been observed in other examples with a large quantity of stars.

Stars in the constellation of Centaurus and Southern Cross were correctly identified.

4.1.6 Test with picture from the surface of Mars

The image in **Figure 4.20** was taken by the Spirit (50) hover from the surface of planet Mars, using it's panoramic camera during the martian night. The picture exhibits the constellation of Orion and was taken in March 11 of 2004. The exposure time was 8 seconds.

Figure 4.21 display the results of processing **Figure 4.20** using the lens model without distortions and with distortions.

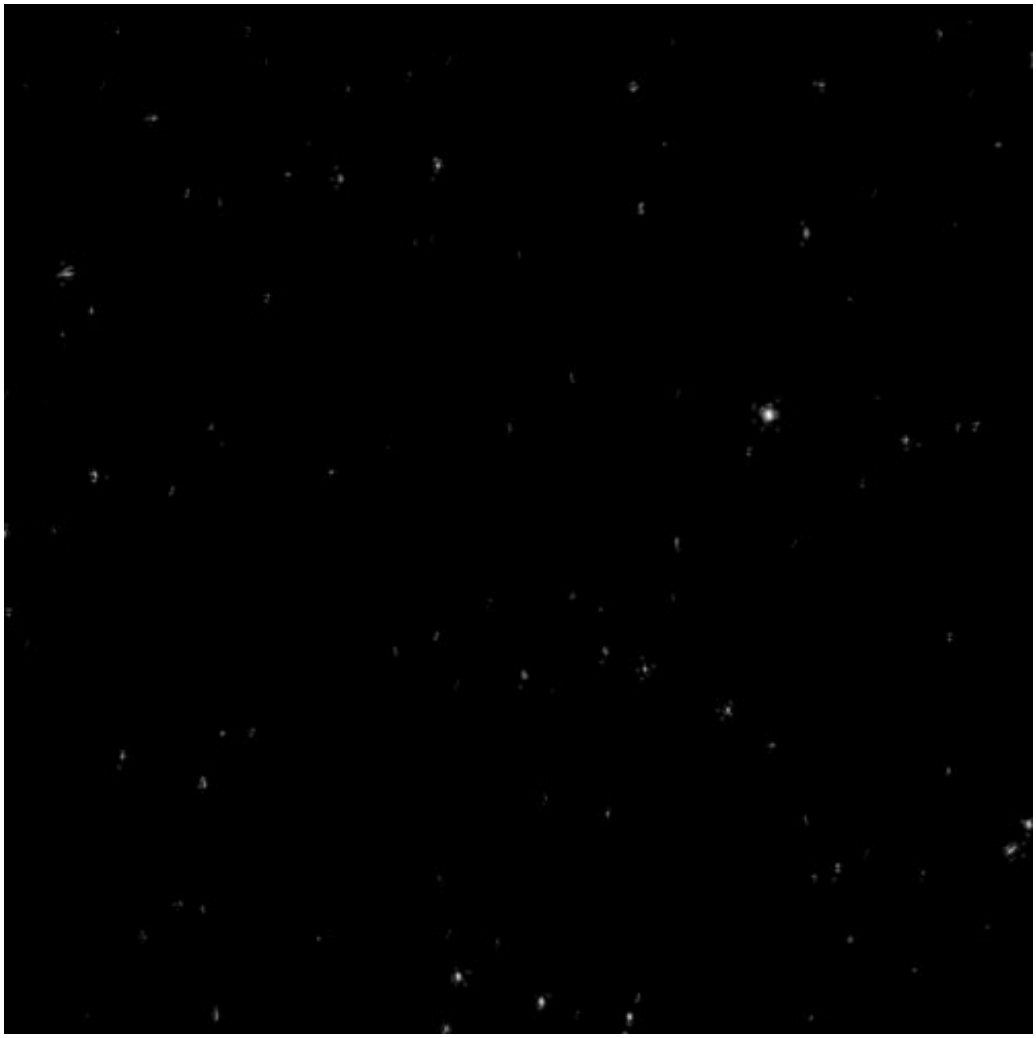
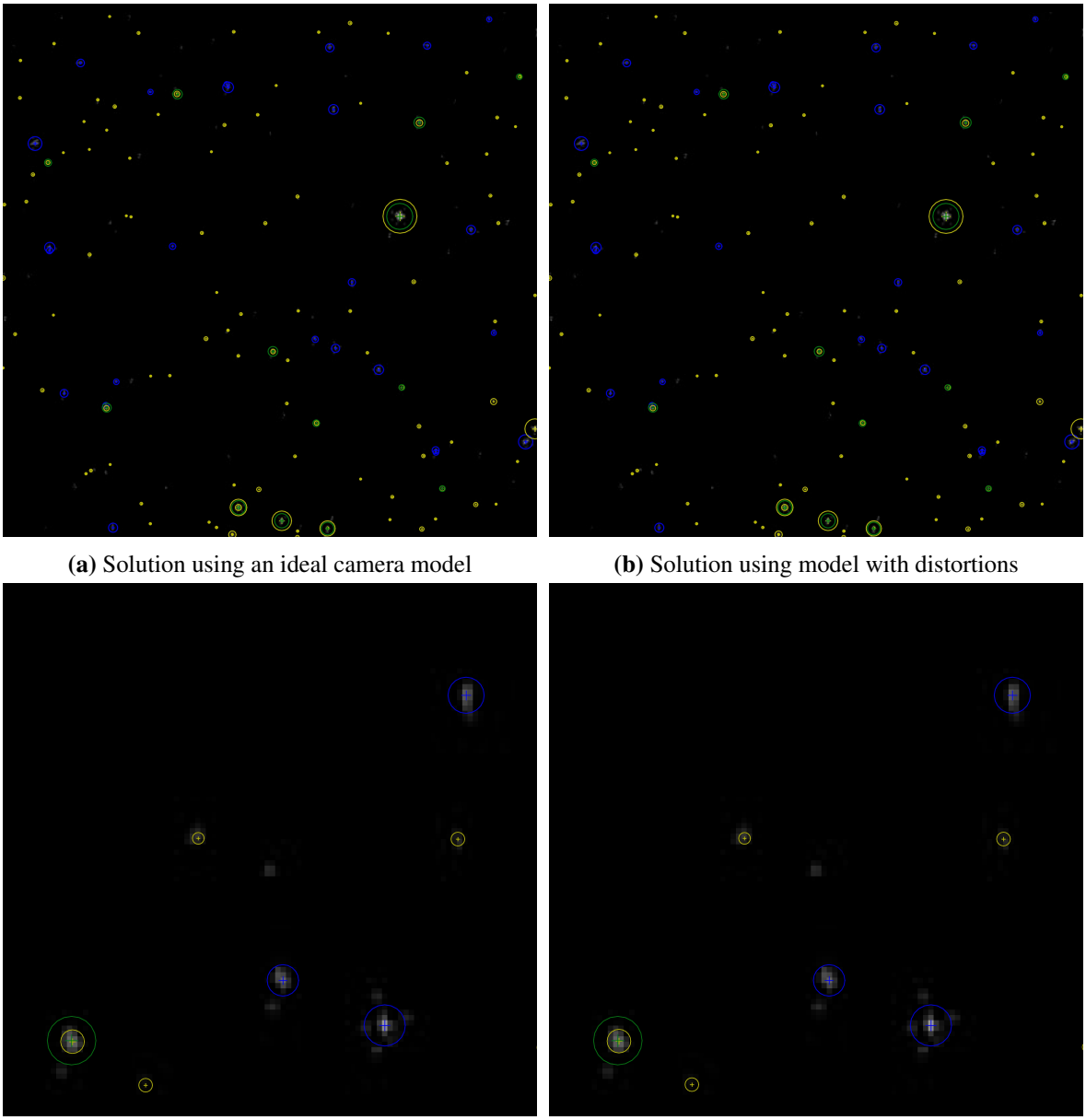


Figure 4.20: Image of the Orion constellation taken by the Spirit (50) hover from the surface of Mars with an 8 s exposure time.



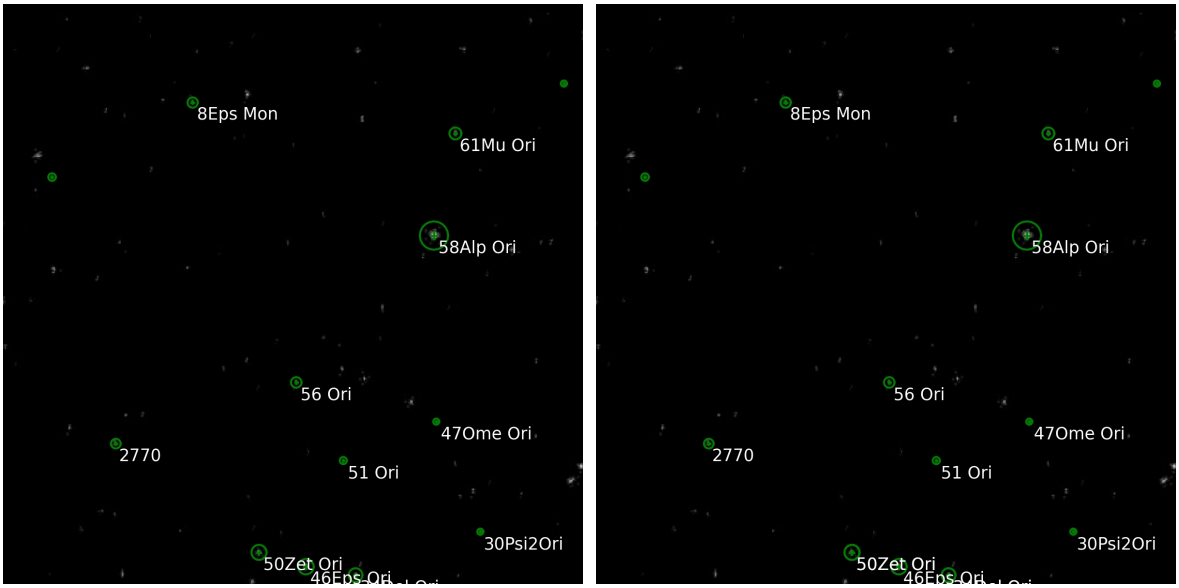
(a) Solution using an ideal camera model

(b) Solution using model with distortions

(c) Close-up of the ideal camera solution

(d) Close-up of the solution using distortions

Figure 4.21: Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars.



(a) Solution using an ideal camera model **(b)** Solution using model with distortions

Figure 4.22: Star names as described in the BSC (21) for the stars identified in **Figure 4.21**.

Table 4.11 and **Table 4.12** display the results of the first and second identifications.

Table 4.11: Matches found for **Figure 4.20** using the model **without distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0,5	2061	58Alp Ori	(121.7,50.9)	(121.5,50.7)	0,3
1,7	1903	46Eps Ori	(11.2,-234.6)	(11.2,-234.5)	0,1
2,05	1948	50Zet Ori	(-29.4,-222.1)	(-29.2,-222.2)	0,2
2,23	1852	34Del Ori	(54.0,-241.7)	(54.0,-241.8)	0
4,12	2124	61Mu Ori	(139.9,138.3)	(139.9,138.6)	0,3
4,44	2298	8Eps Mon	(-86.8,165.4)	(-86.3,165.3)	0,5
4,53	2113		(-152.8,-129.1)	(-152.6,-128.5)	0,6
4,57	1934	47Ome Ori	(123.2,-109.6)	(123.5,-109.6)	0,2
4,59	1811	30Psi2Ori	(161.4,-204.2)	(161.4,-204.4)	0,2
4,78	2037	56 Ori	(3.0,-76.0)	(2.9,-75.8)	0,3
4,91	1963	51 Ori	(43.7,-143.0)	(43.4,-143.1)	0,3
5,2	2334		(-207.6,101.2)	(-207.5,101.0)	0,2
5,7	2099		(233.9,181.7)	(233.4,181.6)	0,5

Table 4.12: Matches found for **Figure 4.20** using the model **with distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0,5	2061	58Alp Ori	(121.6,50.7)	(121.5,50.7)	0,1
1,7	1903	46Eps Ori	(11.2,-234.5)	(11.2,-234.5)	0
2,05	1948	50Zet Ori	(-29.3,-222.1)	(-29.2,-222.2)	0,2
2,23	1852	34Del Ori	(54.0,-241.8)	(54.0,-241.8)	0
4,12	2124	61Mu Ori	(139.9,138.1)	(139.9,138.6)	0,5
4,21	1949	50Zet Ori	(-29.3,-222.0)	(-29.2,-222.2)	0,2
4,44	2298	8Eps Mon	(-86.6,165.2)	(-86.3,165.3)	0,3
4,53	2113		(-152.7,-129.0)	(-152.6,-128.5)	0,5
4,57	1934	47Ome Ori	(123.1,-109.5)	(123.5,-109.6)	0,4
4,59	1811	30Psi2Ori	(161.4,-204.3)	(161.4,-204.4)	0,1
4,78	2037	56 Ori	(3.1,-75.9)	(2.9,-75.8)	0,2
4,91	1963	51 Ori	(43.7,-142.9)	(43.4,-143.1)	0,4
5,2	2334		(-207.4,101.1)	(-207.5,101.0)	0,1
5,7	2099		(233.8,181.5)	(233.4,181.6)	0,4
6,72	2299		(-86.6,165.3)	(-86.3,165.3)	0,2
6,85	1851	34Del Ori	(54.3,-241.4)	(54.0,-241.8)	0,5

13 stars were identified using the model without distortions and 16 stars when using the model with distor-

tions. All the detected stars in both solutions display an error of under 1 pixel. It is possible that the image has been geometrically calibrated by NASA before publication to the website.

Among the identified stars are the ones belonging to the Orion constellation, especially star Betelgeuse ("58Alp Ori"), which is mentioned in the picture description. This example can be used to explain the method of orientation using the stars to a lay audience. The test clearly shows in a simple way that the method used for navigation and in the algorithms in this work is not geocentric. It can be utilized in space and in other planets, without changing the star catalog, and shows that the constellations display the same shapes, whether they are being seen from Mars or from Earth.

4.1.7 Test with foreign body: city lights

Figure 4.23 was obtained from (28) and displays the Orion constellation. It was taken on February 14th 2002, using a Nikon Coolpix 885 and an exposure time of 10.27 s. The photograph has two street lamps in the lower part of the image and a white text over the image with information about the picture.

The star identification algorithm does not work on **Figure 4.23** as it is. The segmentation algorithm detected too many false stars in the text area and was unable to reach a solution, but if the text is removed and only the street lights remain, the algorithm successfully identifies the stars. **Figure 4.24** contains the version of the image that the algorithm is able to process.

Figure 4.25 display the results of processing **Figure 4.24** using the lens model without distortions and with distortions.

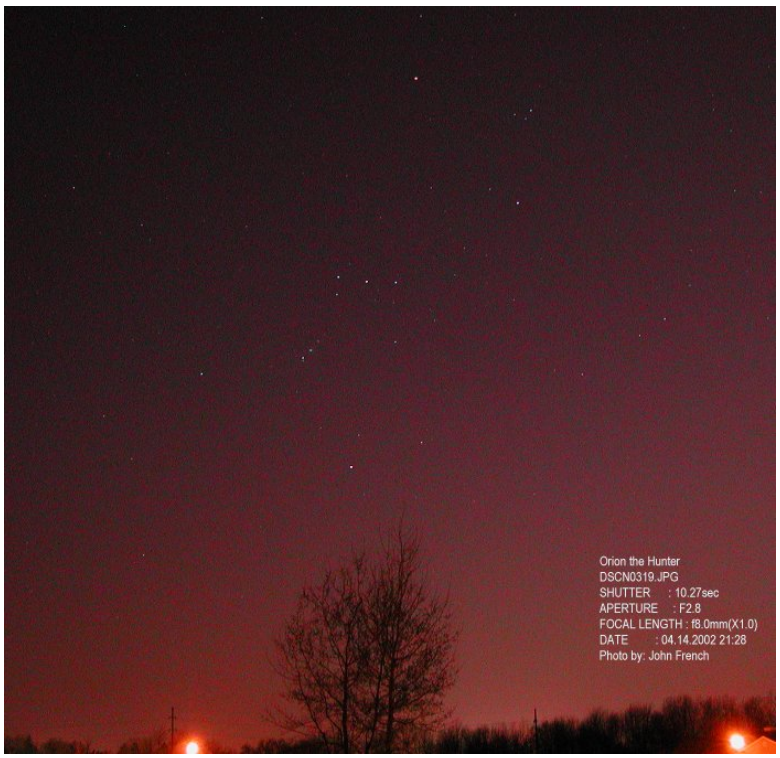
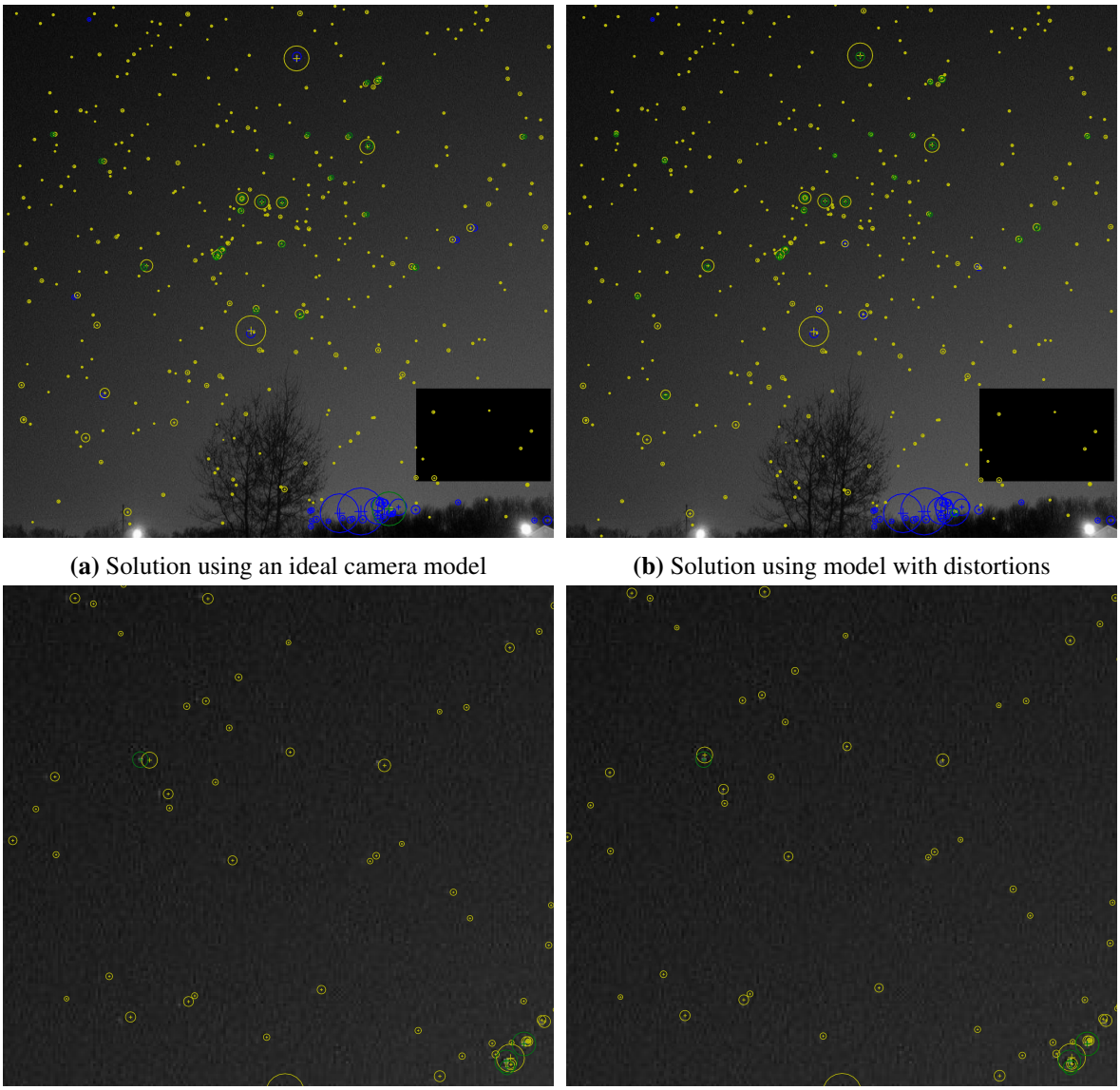


Figure 4.23: Picture of the Orion constellation taken with a Nikon Coolpix 885 in February 14th 2002 and exposure time of 10.27 taken from (28).

Figure 4.24: Edição da Figura 52 suprimindo a legenda em branco.



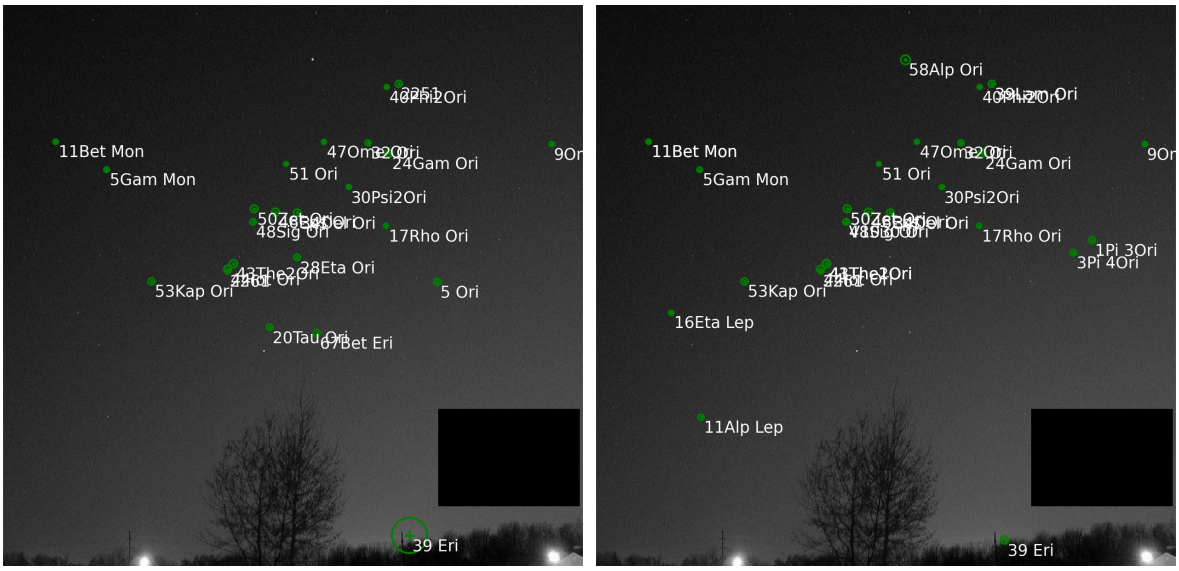
(a) Solution using an ideal camera model

(b) Solution using model with distortions

(c) Close-up of the ideal camera solution

(d) Close-up of the solution using distortions

Figure 4.25: Solutions with and without using the distortion model. Green markers are correct matches, blue markers are detected stars and yellow markers are "missing" stars.



(a) Solution using an ideal camera model

(b) Solution using model with distortions

Figure 4.26: Star names as described in the BSC (21) for the stars identified in **Figure 4.25**.

Table 4.13 and **Table 4.14** display the results of the first and second identifications.

Table 4.13: Matches found for **Figure 4.24** using the model **without distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
1,64	1790	24Gam Ori	(113.1,158.5)	(115.9,160.3)	3,3
1,7	1903	46Eps Ori	(-21.0,88.5)	(-21.1,89.0)	0,5
2,06	2004	53Kap Ori	(-167.3,7.3)	(-170.8,5.5)	4
2,23	1852	34Del Ori	(4.8,87.7)	(5.3,88.3)	0,8
2,77	1899	44Iot Ori	(-77.3,21.3)	(-79.0,20.0)	2,1
2,79	1666	67Bet Eri	(27.1,-53.7)	(28.6,-56.9)	3,5
3,36	1788	28Eta Ori	(4.3,35.2)	(4.9,34.5)	1
3,6	1735	20Tau Ori	(-28.2,-47.1)	(-28.3,-50.0)	2,9
3,81	1931	48Sig Ori	(-47.2,77.2)	(-48.1,77.5)	0,9
3,98	2227	5Gam Mon	(-221.5,140.2)	(-225.1,140.5)	3,6
4,07	1580	9Omi2Ori	(309.9,171.5)	(312.7,171.6)	2,8
4,09	1907	40Phi2Ori	(111.0,238.4)	(113.2,240.6)	3,1
4,2	1839	32 Ori	(88.4,170.9)	(90.7,173.0)	3,1
4,21	1949	50Zet Ori	(-45.9,92.6)	(-46.8,93.2)	1
4,46	1698	17Rho Ori	(109.1,72.3)	(112.5,72.6)	3,4
4,5	2357	11Bet Mon	(-283.7,174.6)	(-286.6,174.3)	3
		11Bet Mon			
4,57	1934	47Ome Ori	(36.1,172.0)	(37.4,174.3)	2,6
4,59	1811	30Psi2Ori	(65.5,118.3)	(67.5,119.5)	2,4
4,87	1318	39 Eri	(143.2,-301.9)	(140.7,-301.8)	2,5
4,91	1963	51 Ori	(-8.9,145.5)	(-8.5,147.5)	2
5,08	1897	43The2Ori	(-70.9,27.9)	(-71.9,26.9)	1,4
5,33	1562	5 Ori	(173.6,4.4)	(174.4,5.1)	1,1
5,6	1883		(129.4,246.8)	(127.8,244.3)	3
5,67	1886		(-77.1,18.6)	(-78.6,17.5)	1,9

Table 4.14: Matches found for **Figure 4.24** using the model **with distortions**.

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
0,5	2061	58Alp Ori	(23.5,274.8)	(23.6,273.3)	1,5
1,64	1790	24Gam Ori	(114.9,161.0)	(115.9,160.3)	1,2
1,7	1903	46Eps Ori	(-21.2,89.6)	(-21.1,89.0)	0,6
2,05	1948	50Zet Ori	(-46.5,93.8)	(-46.8,93.2)	0,6
2,06	2004	53Kap Ori	(-169.8,7.4)	(-170.8,5.5)	2,2
2,23	1852	34Del Ori	(4.9,88.8)	(5.3,88.3)	0,6
2,58	1865	11Alp Lep	(-223.7,-156.8)	(-223.4,-158.8)	2,1

Table 4.14 Continued:

Visual magnitude	Star catalog number	Star catalog name	Expected position in picture	Actual centroid position in picture	Error in pixels
2,77	1899	44Iot Ori	(-78.2,21.5)	(-79.0,20.0)	1,7
3,19	1543	1Pi 3Ori	(248.0,56.3)	(249.1,55.4)	1,5
3,54	1879	39Lam Ori	(127.4,245.5)	(127.8,244.3)	1,3
3,69	1552	3Pi 4Ori	(225.0,41.3)	(226.4,40.3)	1,8
3,71	2085	16Eta Lep	(-259.1,-30.6)	(-259.4,-32.6)	2,1
3,81	1931	48Sig Ori	(-47.8,78.2)	(-48.1,77.5)	0,7
3,98	2227	5Gam Mon	(-224.7,142.3)	(-225.1,140.5)	1,9
4,07	1580	9Omi2Ori	(312.1,172.7)	(312.7,171.6)	1,2
4,09	1907	40Phi2Ori	(112.6,241.9)	(113.2,240.6)	1,5
4,2	1839	32 Ori	(89.8,173.5)	(90.7,173.0)	1
4,21	1949	50Zet Ori	(-46.5,93.8)	(-46.8,93.2)	0,6
4,46	1698	17Rho Ori	(110.6,73.3)	(112.5,72.6)	2
4,5	2357	11Bet Mon	(-286.4,176.2)	(-286.6,174.3)	2
		11Bet Mon			
4,57	1934	47Ome Ori	(36.6,174.6)	(37.4,174.3)	0,8
4,59	1811	30Psi2Ori	(66.4,120.0)	(67.5,119.5)	1,2
4,6	2356	11Bet Mon	(-286.3,176.2)	(-286.6,174.3)	2
4,78	1887		(-78.0,19.0)	(-78.6,17.5)	1,6
4,87	1318	39 Eri	(144.5,-304.8)	(143.2,-307.2)	2,7
4,91	1963	51 Ori	(-9.0,147.6)	(-8.5,147.5)	0,5
5,08	1897	43The2Ori	(-71.7,28.2)	(-71.9,26.9)	1,3
5,13	1895	41The1Ori	(-71.0,28.2)	(-71.9,26.9)	1,6
5,61	1880	39Lam Ori	(127.4,245.5)	(127.8,244.3)	1,3
5,67	1886		(-78.0,18.8)	(-78.6,17.5)	1,5
6,65	1932		(-47.9,78.4)	(-48.1,77.5)	0,9
6,7	1896	41The1Ori	(-71.0,28.3)	(-71.9,26.9)	1,6
6,73	1893	41The1Ori	(-70.9,28.2)	(-71.9,26.9)	1,6
6,85	1851	34Del Ori	(5.1,89.0)	(5.3,88.3)	0,7
7,96	1894	41The1Ori	(-70.9,28.3)	(-71.9,26.9)	1,7

24 stars were identified when using the model without distortions, 35 were identified when using the model with distortions. Using the model with distortions increased the number of identified stars and reduced the error between the expected position and the position in which the star was found.

The algorithm was unable to handle the large amount of spurious stars detected when the white text was left on the image.

4.1.8 Summary

The algorithm successfully identified stars in the proposed tests, all of them without previous knowledge of camera parameters or shooting conditions. In many cases, however, it was necessary to perform manual adjust-

ments to the image to improve the performance of the segmentation algorithm.

Manual image processing of the pictures was needed only on difficult cases. With better quality images of the sky the algorithm performed well.

Star identification, however, is not the only objective of the algorithm. The calibration of the camera - or the determination of the distortion parameters - is also important.

In the first test it was objectively demonstrated that the focal distance was correct and that the distortion correction was at least partially effective.

In most tests the use of the model with distortions brought improvements to the initial solution, reducing the error and increasing the number of identified stars. There were two exceptions however.

In the test with blurred stars and the test with few stars the distortion calculation diverged from the correct solution, generating star fields which do not cover the entire image and generate inconsistent star placement predictions. That may be a case of overfitting aggravated by the use of RANSAC. If there are too many parameters to optimize and two few constraints to the model, it is possible that the optimization converges to a solution which perfectly satisfies the constraints but makes no physical sense. RANSAC removes samples that do not fit well with the model, so it may make the problem worse.

The general recomendation is to use the model with distortions only if there are many good stars. If there are less than 10 stars or if the measurements are not very accurate (blurred stars, for example), it is best to use the model without distortions.

4.2 Triangulation algorithm

4.2.1 Nikon D5000 - BSC

The first version of the triangulation algorithm used the Bright Star Catalog (21), which contains less stars and is not as precise as the Hipparcos (53) catalog. The preliminary results, however, were very accurate, with the error being only 17 times larger than the achieved by Deep Space 1 (see **Section 2.1.1**). To the extent of the tests performed it is believed that either this outstanding result is an statistical anomaly or there still an unresolved issue with the second version of the algorithm. Many hours have been spent searching for a potential issue and none was found so far.

The raw output from the triangulation run with light speed correction begins with:

Listing 4.1: Part 1 of the raw log of the triangulation algorithm on the Nikon using BSC (21).

```
[(750734770.83455908, 17187.073605715814, ['jupiter', 'saturn']),
   (750734916.26770961, 1824110180.269901, ['saturn', 'jupiter']),
   (2429172250725.4658, 1454091570.7179723, ['saturn', 'mercury']),
   (2429178782536.9727, 78697026.850330636, ['mercury', 'saturn']),
   (2488295501830.4268, 174710637.57631636, ['jupiter', 'mercury']),
   (2488295955742.542, 900624164.33768106, ['mercury', 'jupiter']),
   (5463054367128.4424, 198199647.52461615, ['venus', 'mercury']),
   (5463054961855.918, 234563725.90498132, ['mercury', 'venus']),
   (15067564860323.004, 134394057.57404077, ['venus', 'saturn']),
   (15067565169652.596, 1474664138.7817092, ['saturn', 'venus']),
```

```
(15272699338604.75, 237046672.87785971, ['jupiter', 'venus']),
(15272703701722.742, 896068532.14546382, ['venus', 'jupiter']),
(36178250065592.383, 937053168.10825312, ['mars', 'jupiter']),
(36178252054583.312, 22621660.895046059, ['jupiter', 'mars']),
(36514677634236.156, 1414060875.0118699, ['saturn', 'mars']),
(36514683288227.719, 97970077.708865836, ['mars', 'saturn']),
(57544251460057.469, 77326517.725034252, ['mercury', 'mars']),
(57544258234967.312, 216401434.54781625, ['mars', 'mercury']),
(98441922661816.969, 131665673.63619593, ['venus', 'mars']),
(98441927728649.516, 273684460.10766762, ['mars', 'venus'])]
```

Which is the quality of the solutions. The first value in the line is the value of the cost function for the solution, and the second value is the actual error from the ground truth, followed by the name of the planets considered.

The time span in which the pictures were taken follows:

Listing 4.2: Part 2 of the raw log of the triangulation algorithm on the Nikon using BSC (21).

```
Measurements start time: 2016-07-27 22:34:15
Measurements finish time: 2016-07-27 22:40:39
```

Part 3 contains an statistical report on the solutions found. Based on the standard deviation of the measurements, the precision is estimated and the actual accuracy of the measurement is reported according to the model fitness. Measurements are in radians. The reported right ascension and declination help to validate the solution using Stellarium (7).

Listing 4.3: Part 3 of the raw log of the triangulation algorithm on the Nikon using BSC (21).

```
Observed bodies:
 Right ascension and declination in relation to observer: 11h29m07.88s
    04o34'23.35"
 Perceived brightness: 4949.9209073
 Measurement model fit estimated error (accuracy): 9.72418441803e-05
 Measurement standard deviation (precision): 1.35072417096e-05

 Right ascension and declination in relation to observer: 16h33m29.39s
    -20o15'50.64"
 Perceived brightness: 1422.92714043
 Measurement model fit estimated error (accuracy): 0.000155017545164
 Measurement standard deviation (precision): 1.50765915314e-05
```

Part 4 gives the estimated position of the identified bodies according to Horizon. That is used to also validate that the algorithm itself is performing the correct transformations and light speed corrections.

Listing 4.4: Part 4 of the raw log of the triangulation algorithm on the Nikon using BSC (21).

```
[-16340.56618059   54891.65145399   28243.72334474]
63857.7748659 0.000420455243335
```

```
Horizons estimated position for Jupiter as seen from Earth:
  11h29m08.56s  04o34'13.31"
Horizons estimated position for Saturn as seen from Earth:
  16h33m29.80s -20o15'49.40"
Horizons estimated position for Jupiter as seen from Earth with rough
   light speed correction:
  11h29m08.04s  04o34'16.60"
Horizons estimated position for Saturn as seen from Earth with rough
   light speed correction:
  16h33m29.36s -20o15'48.29"
```

Part 5 gives the final error of the position found in relation to the Horizons database.

Listing 4.5: Part 5 of the raw log of the triangulation algorithm on the Nikon using BSC (21).

```
Best error achieved (km): 17187.0736057
Time for Earth to transit the error distance (minutes): 9.76439706861
Measurement start time (GMT): 2016-07-27 22:34:15
Measurement finish time (GMT): 2016-07-27 22:40:39
```

Considering that at the time of that measurement Jupiter was about 918 million kilometers away from earth and Saturn was about 1409 million kilometers away, the measurement had an error of about one part in 50000 ($918 \times 10^6/17187$), or an angular error of approximately 2×10^-5 radians, which is about half the value of 3.7×10^{-5} radians calculated for the Rayleigh limit of the Nikon optical system in **Section 4.3.1** and corresponds to about $1/5$ of the angle covered by a pixel in the Nikon optical system.

As stated in the beginning of the section, the other results this result may be an statistical anomaly, or even an issue belonging to the second version of the algoritm. Many hours have been spent searching for a potential issue and none was found so far.

4.2.2 Nikon D5000 - Hipparcos

Following versions of the triangulation algorithm used the Hipparcos Catalog (53). The final results were about 100 times worse than Deep Space 1 location performance (see **Section 2.1.1**).

The log of the star matching part gives a report on the identified objects in the frames. It also gives an estimate of the absolute error, but uses information of the fitness of the stars in the identification step to estimate a value between 2×10^-5 and $5 \times^1 0 - 5$ radians. The standard deviation gives a good idea of how good the centroid measuring algorithm is, giving a precision of 1×10^-5 radians or better.

Listing 4.6: Section the raw log of the **star matching** algorithm on the Nikon using Hipparcos

```
Unmatched stars: {0, 21}
Testing unmatched centroid index  0
Testing unmatched centroid index  21
Found 3 objects with two or more detections in different pictures.
Object 0 was detected in 5 frames:
    measurement_data/DSC_4334.centroids.txt at 2016-07-27 21:34:15 +0000
    measurement_data/DSC_4335.centroids.txt at 2016-07-27 21:34:34 +0000
```

```
    measurement_data/DSC_4336.centroids.txt at 2016-07-27 21:34:56 +0000
    measurement_data/DSC_4337.centroids.txt at 2016-07-27 21:35:25 +0000
    measurement_data/DSC_4338.centroids.txt at 2016-07-27 21:35:48 +0000
  mean observed brightness: 4949.9209073
  mean position solution: [-0.98778801  0.13385852  0.07972916]
  mean RADE position solution: 11h29m07.83s  04o34'22.79"
  average estimated error from solutions: 4.50521116556e-05
  standard deviation of measurements: 1.09215020074e-05
Object 1 was detected in 4 frames:
    measurement_data/DSC_4339.centroids.txt at 2016-07-27 21:37:46 +0000
    measurement_data/DSC_4343.centroids.txt at 2016-07-27 21:40:10 +0000
    measurement_data/DSC_4344.centroids.txt at 2016-07-27 21:40:24 +0000
    measurement_data/DSC_4345.centroids.txt at 2016-07-27 21:40:39 +0000
  mean observed brightness: 1422.92714043
  mean position solution: [-0.34576403 -0.87206394 -0.34634047]
  mean RADE position solution: 16h33m29.32s -20o15'49.12"
  average estimated error from solutions: 1.66786415632e-05
  standard deviation of measurements: 6.24059680985e-06
Object 2 was detected in 2 frames:
    measurement_data/DSC_4344.centroids.txt at 2016-07-27 21:40:24 +0000
    measurement_data/DSC_4345.centroids.txt at 2016-07-27 21:40:39 +0000
  mean observed brightness: 35.0611023712
  mean position solution: [-0.47392562 -0.85819977 -0.19719956]
  mean RADE position solution: 16h04m21.86s -11o22'23.68"
  average estimated error from solutions: 2.34225949015e-05
  standard deviation of measurements: 2.23948417495e-06
```

The following entries are parts of the triangulation log, and begin with a check to verify that the algorithm is performing the correct transformations and would see all the planets in their correct locations:

Listing 4.7: Part 1 of the raw log of the **triangulation** algorithm on the Nikon using Hipparcos (53).

```
Apparent position of  mars at 2016-07-27 21:34:15 +0000 is:
 uncorrected:         15h38m59.50s -22o26'14.90"
 corrected for time:  15h38m58.65s -22o26'11.12"
Apparent position of  uranus at 2016-07-27 21:34:15 +0000 is:
 uncorrected:         01h30m50.40s  08o50'09.26"
 corrected for time:  01h30m50.12s  08o50'07.58"
Apparent position of  venus at 2016-07-27 21:34:15 +0000 is:
 uncorrected:         09h27m43.42s  16o29'58.81"
 corrected for time:  09h27m41.90s  16o30'05.56"
Apparent position of  jupiter at 2016-07-27 21:34:15 +0000 is:
 uncorrected:         11h29m06.93s  04o34'24.09"
 corrected for time:  11h29m06.40s  04o34'27.38"
Apparent position of  mercury at 2016-07-27 21:34:15 +0000 is:
 uncorrected:         09h51m32.25s  14o08'57.20"
```

```
 corrected for time:  09h51m30.94s  14o09'06.68"
Apparent position of  saturn at 2016-07-27 21:34:15 +0000 is:
 uncorrected:         16h33m30.08s -20o15'49.46"
 corrected for time:  16h33m29.64s -20o15'48.34"
```

The projected positions of the unidentified objects are then shown, with the observation time, and the result best matching the data is presented.

Listing 4.8: Part 2 of the raw log of the **triangulation** algorithm on the Nikon using Hipparcos (53).

```
[5, 4, 2]
0 2016-07-27 21:34:15 +0000 11h29m07.89s  04o34'24.74"
0 2016-07-27 21:34:34 +0000 11h29m07.68s  04o34'26.54"
0 2016-07-27 21:34:56 +0000 11h29m07.64s  04o34'15.52"
0 2016-07-27 21:35:25 +0000 11h29m08.13s  04o34'23.82"
0 2016-07-27 21:35:48 +0000 11h29m07.82s  04o34'23.32"
1 2016-07-27 21:37:46 +0000 16h33m29.19s -20o15'47.26"
1 2016-07-27 21:40:10 +0000 16h33m29.52s -20o15'52.55"
1 2016-07-27 21:40:24 +0000 16h33m29.29s -20o15'48.10"
1 2016-07-27 21:40:39 +0000 16h33m29.30s -20o15'48.55"
Horizons data:
mars A.D. 2016-Jul-27 21:34:15.0000 1469655255.0
earth A.D. 2016-Jul-27 21:34:15.0000 1469655255.0
uranus A.D. 2016-Jul-27 21:34:15.0000 1469655255.0
venus A.D. 2016-Jul-27 21:34:15.0000 1469655255.0
jupiter A.D. 2016-Jul-27 21:34:15.0000 1469655255.0
mercury A.D. 2016-Jul-27 21:34:15.0000 1469655255.0
saturn A.D. 2016-Jul-27 21:34:15.0000 1469655255.0
Best result:
identified planets: ('jupiter', 'saturn')
error: 107177 kilometers
       8.4 earth diameters
       1.01 hours at earth speed
       0.4 seconds at light speed
X, Y, Z position: [  8.78167888e+07  -1.13647789e+08  -4.92994738e+07]
```

The matching results and the error of 107177 kilometers is presented.

Listing 4.9: Part 3 of the raw log of the **triangulation** algorithm on the Nikon using Hipparcos (53).

```
Best 10:
(3.8359668151460957e-11, 107176.98806119281, ('jupiter', 'saturn'))
(2.2157232294191263e-07, 78764723.091217026, ('mercury', 'saturn'))
(1.2317145282802832e-06, 134517951.71964693, ('venus', 'saturn'))
(2.6209800141962098e-06, 23707048.431100197, ('jupiter', 'mars'))
(4.4478143430981729e-06, 98155220.062480628, ('mars', 'saturn'))
(4.3893824710579411e-05, 198282876.60505879, ('venus', 'mercury'))
(8.9272349675584381e-05, 132180990.47291431, ('venus', 'mars'))
```

```
(9.6481950790924183e-05, 77775232.14805387, ('mercury', 'mars'))
(0.00034271407465518843, 177401236.34642059, ('jupiter', 'mercury'))
(0.039307818212242933, 10013898144878.834, ('uranus', 'venus'))
```

The last part of the log shows the value of the cost function of the solution, the ground truth error of the solution followed by the name of the planets that are part of it.

Considering that at the time of that measurement Jupiter was about 918 million kilometers away from earth and Saturn was about 1409 million kilometers away, the measurement had an error of about one part in 10000 ($918 \times 10^6/107177$), or an angular error of approximately 1.1×10^-4 radians, which is about three times the value of 3.7×10^{-5} radians calculated for the Rayleigh limit of the Nikon optical system in **Section 4.3.1** and corresponds to about the same angle covered by a single pixel in the Nikon optical system.

4.2.3 Raspberry Pi Camera NoIR version 1

The latest version of the triangulation algorithm using the Hipparcos Catalog (53) was applied to the Raspberry Pi raw data. The final error was about 170 times larger than the attained by Deep Space 1 (see **Section 2.1.1**).

The log of the star matching part gives a report on the identified objects in the frames. It also gives an estimate of the absolute error, but uses information of the fitness of the stars in the identification step to estimate a value between 1×10^-3 and $4 \times^1 0 - 3$ radians. The standard deviation gives a good idea of how good the centroid measuring algorithm is, giving a precision of 1×10^-4 radians or better.

Listing 4.10: Section the raw log of the **star matching** algorithm on the Nikon using Hipparcos (53). Ellipsis indicate parts suppressed for readability.

```
Found 55 objects with two or more detections in different pictures.
Object 0 was detected in 47 frames:
    measurement_data/000_marte.centroids.txt at 2015-12-16 07:20:45 +0000
    ...
    measurement_data/059_marte.centroids.txt at 2015-12-16 07:22:48 +0000
  mean observed brightness: 219.200253191
  mean position solution: [-0.94241194 -0.31633562 -0.10858278]
  mean RADE position solution: 13h14m13.23s -06o14'01.06"
  average estimated error from solutions: 0.00335436985545
  standard deviation of measurements: 0.000621370770706
Object 1 was detected in 18 frames:
    measurement_data/002_jupiter.centroids.txt at 2015-12-16 07:13:28
       +0000
    ...
    measurement_data/050_jupiter.centroids.txt at 2015-12-16 07:15:04
       +0000
  mean observed brightness: 3007.92388889
  mean position solution: [-0.99051725  0.11630387  0.07313648]
  mean RADE position solution: 11h33m12.75s  04o11'38.96"
  average estimated error from solutions: 0.00174339594731
  standard deviation of measurements: 9.81110200515e-05
Object 2 was detected in 17 frames:
```

```
    measurement_data/001_marte.centroids.txt at 2015-12-16 07:20:48 +0000
    ...
    measurement_data/054_marte.centroids.txt at 2015-12-16 07:22:39 +0000
  mean observed brightness: 89.0514529412
  mean position solution: [-0.94615274 -0.26329305  0.18833808]
  mean RADE position solution: 13h02m12.16s  10o51'20.93"
  average estimated error from solutions: 0.00342459353089
  standard deviation of measurements: 0.000367143169059
Object 3 was detected in 13 frames:
    measurement_data/007_jupiter.centroids.txt at 2015-12-16 07:13:37
    ...
    measurement_data/052_jupiter.centroids.txt at 2015-12-16 07:15:12
       +0000
  mean observed brightness: 71.5322615385
  mean position solution: [-0.80113844  0.31548314 -0.50855573]
  mean RADE position solution: 10h34m01.39s -30o34'04.76"
  average estimated error from solutions: 0.00226336789876
  standard deviation of measurements: 0.0024580983595
```

The triangulation log begins with a check of the correct operation of the transformations in the algorithm, to make sure the estimated viewing position of all the planets is correct.

Listing 4.11: Part 1 of the raw log of the **triangulation** algorithm on the Raspberry Pi using Hipparcos (53).

```
Apparent position of  mercury at 2015-12-16 07:13:23 +0000 is:
 uncorrected:         18h40m55.66s -25o23'03.58"
 corrected for time:  18h40m54.24s -25o23'04.79"
Apparent position of  saturn at 2015-12-16 07:13:23 +0000 is:
 uncorrected:         16h30m54.41s -20o12'39.89"
 corrected for time:  16h30m53.97s -20o12'38.74"
Apparent position of  venus at 2015-12-16 07:13:23 +0000 is:
 uncorrected:         14h44m47.97s -13o33'03.91"
 corrected for time:  14h44m47.29s -13o33'00.07"
Apparent position of  uranus at 2015-12-16 07:13:23 +0000 is:
 uncorrected:         01h01m21.26s  05o50'37.75"
 corrected for time:  01h01m20.98s  05o50'36.00"
Apparent position of  jupiter at 2015-12-16 07:13:23 +0000 is:
 uncorrected:         11h33m10.28s  04o10'14.01"
 corrected for time:  11h33m09.76s  04o10'17.29"
Apparent position of  mars at 2015-12-16 07:13:23 +0000 is:
 uncorrected:         13h14m07.02s -06o14'25.23"
 corrected for time:  13h14m06.22s -06o14'19.93"
```

This part shows the group sizes and the individual position measurements.

Listing 4.12: Part 2 of the raw log of the **triangulation** algorithm on the Raspberry Pi using Hipparcos (53). Ellipsis indicate parts suppressed for readability.

```
[47, 18, 17, 13, 12, 9, 8, 7, 7, 6, 5, 4, 4, 4, 3, 3, 3, 3, 3, 3, 3, 3,
   3, 3, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
   2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
0 2015-12-16 07:20:45 +0000 13h14m51.91s -06o22'44.09"
0 2015-12-16 07:20:48 +0000 13h14m08.35s -06o12'04.00"
...
0 2015-12-16 07:22:43 +0000 13h14m08.46s -06o14'00.93"
0 2015-12-16 07:22:44 +0000 13h14m09.57s -06o12'54.75"
0 2015-12-16 07:22:46 +0000 13h14m07.99s -06o12'51.51"
0 2015-12-16 07:22:48 +0000 13h14m20.37s -06o15'30.71"
1 2015-12-16 07:13:28 +0000 11h33m13.53s  04o12'34.35"
1 2015-12-16 07:13:32 +0000 11h33m14.02s  04o11'29.40"
...
1 2015-12-16 07:14:42 +0000 11h33m14.97s  04o12'02.00"
1 2015-12-16 07:15:03 +0000 11h33m14.52s  04o11'02.82"
1 2015-12-16 07:15:04 +0000 11h33m12.22s  04o12'33.80"
```

The best result is found and the error is calculated. "Object 0" was matched with Mars and "Object 1" was matched with Jupiter.

Listing 4.13: Part 3 of the raw log of the **triangulation** algorithm on the Raspberry Pi using Hipparcos (53).

```
Horizons data:
mercury A.D. 2015-Dec-16 07:13:23.0000 1450250003.0
saturn A.D. 2015-Dec-16 07:13:23.0000 1450250003.0
venus A.D. 2015-Dec-16 07:13:23.0000 1450250003.0
uranus A.D. 2015-Dec-16 07:13:23.0000 1450250003.0
jupiter A.D. 2015-Dec-16 07:13:23.0000 1450250003.0
mars A.D. 2015-Dec-16 07:13:23.0000 1450250003.0
earth A.D. 2015-Dec-16 07:13:23.0000 1450250003.0
Best result:
identified planets: ('mars', 'jupiter')
error: 168517 kilometers
       13.2 earth diameters
       1.55 hours at earth speed
       0.6 seconds at light speed
X, Y, Z position: [  1.65737395e+07   1.34583317e+08   5.81528084e+07]
Best 10:
(3.0135858429666516e-09, 168517.33163387008, ('mars', 'jupiter'))
(4.2863348228742706e-08, 2492331557.8621459, ('saturn', 'venus'))
(5.2062067678817569e-08, 2727998227.2159591, ('saturn', 'jupiter'))
(6.3267878679436218e-08, 2487539574.1452079, ('saturn', 'mars'))
(1.0083969514637107e-07, 138380752.81092989, ('venus', 'jupiter'))
(1.4520762847102191e-07, 5195127079.3383379, ('saturn', 'uranus'))
(2.245405566743323e-07, 2384874605.8206911, ('saturn', 'mercury'))
```

```
(8.2875085635928781e-07, 405615823.8415997, ('mercury', 'jupiter'))
(1.7013242898023744e-06, 2987618105.8807058, ('mercury', 'uranus'))
(6.3719824124854544e-06, 2928945945.8154402, ('venus', 'uranus'))
```

The last part of the log shows the value of the cost function of the solution, the ground truth error of the solution followed by the name of the planets that are part of it.

Considering that at the time of that measurement Mars was about 274 million kilometers away from earth and Jupiter was about 792 million kilometers away, the measurement had an error of about one part in 2000 ($274 \times 10^6/168517$), or an angular error of approximately 6×10^-4 radians, which is about the value of 5.4×10^-4 radians calculated for the Rayleigh limit of the Raspberry Pi optical system in **Section 4.3.1** and corresponds to about 1.6 times the angle covered by a single pixel in the Raspberry pi optical system.

Table 4.15 compares approaches using the Deep Space Network, Deep Space 1 sensors, the Nikon D5000 and the Raspberry Pi camera. The absolute error was calculated as if the nearest reference body was at 2.7×10^8 km, which was the distance from Mars to Earth at the time of the Raspberry Pi experiments. The angular error of Deep Space 1 was estimated using the distance from DS1 to Venus and Mars in June 1999, which was about the same for the two planets at 9.6×10^7 km (the closest two planets to the spacecraft). The measurements obtained by DS1 and the experimental results of this work are close to the diffraction limit of the optical systems.

Solution	Cost (USD)	Diffraction limit	Angular error	Absolute error at 2.74×10^8 km
Deep Space Network	110000/y (at 104h/y (42))	1.2×10^-8 rad S band 9×10^-10 rad X band (11)	$\underline{10^-9 \text{ rad (11)}}$	2.7×10^-1 km
Deep Space 1	153 million (43 to launch) (13)	7.3×10^-6 rad (600nm, MICAS, 10cm)	10^-5 rad ($\underline{10^3 \text{ km}}$ at 9.6×10^7 km)	2.7×10^3 km
Nikon D5000	500 to 650 (2015 price)	3.7×10^-5 rad	$\underline{1.1 \times 10^-4 \text{ rad}}$	3×10^4 km
Raspberry Pi NoIR camera	81.60 (2015 price)	5.4×10^-4 rad	6×10^-4 rad	$\underline{1.7 \times 10^5 \text{ km}}$

Table 4.15: Comparison of cost and performance of solar system navigation alternatives. The underlined numbers are the error that was provided or measured in the experiments, the opposite column is a calculated quantity.

4.3 Error sources

4.3.1 Diffraction limit

Optical systems are limited by diffraction. In theory, a system buit with perfect mirrors or lenses can distinguish between two point light sources that are no closer than it's optical diffraction limit.

One of the ways to define such a limit is the Rayleigh criterion (58) which arbitrated that two point sources of light are resolvable by a system with a circular aperture if the diffraction patterns from the two point sources are separated by more than the distance between the center peak of the airy disk and the first minimum. That criterion is used for evaluating optics systems such as telescopes and microscopes (65) even today.

According to the Rayleigh Criterion, the smallest resolvable angle is given by the formula:

$$\theta = 1.2120\frac{\lambda}{2R} \tag{4.1}$$

Where λ is the wavelength of the light and $2R$ is the aperture diameter.

For the Nikon 50mm D f/1.8 lens stopped down to f/2.8 we have:

$$\theta = 1.2120\frac{550nm}{\frac{50mm}{2.8}} = 1.2120\frac{2.8 \times 5.50 \times 10^{-7}m}{5 \times 10^{-2}m} = 3.7330 \times 10^{-5}radians \tag{4.2}$$

For a nominal 50mm focal length, that corresponds to a distance of $50mm \times 3.7330 \times 10^{-5} = 1.866\mu m$ on the focal plane. For the Nikon D5000 camera used the nominal size of the pixels is $5.5 \times 5.5\mu m$ (51).

The sensor has an optical low pass filter, which filters out high frequency components of the image beyond the sensor's sampling capability and further limits the resolving power achievable.

The resolving power of the optical system consisting of the D5000 camera and the 50mm f/1.8 D lens is not primarily limited by diffraction, since the pixel size itself is much larger than the theoretical minimum resolvable angle projected to the image plane. In addition to that, the sensor is intentionally built to add some blur to the image and there are still lens aberrations to be taken into account.

For the Raspberry Pi NoIR camera we have:

$$\theta = 1.2120\frac{550nm}{\frac{3.6mm}{2.9}} = 1.2120\frac{2.9 \times 5.50 \times 10^{-7}m}{3.6 \times 10^{-3}m} \approx 5.37 \times 10^{-4}radians \tag{4.3}$$

For a nominal 3.6mm focal length, that corresponds to a distance of $3.6mm \times 5.37 \times 10^{-4} \approx 1.9\mu m$ on the focal plane. For the Pi NoIR camera used the nominal size of the pixels is $1.4 \times 1.4\mu m$ (57).

The minimum resolvable angle projected to the image plane is larger than the pixel size of the sensor and the datasheet does not mention the presence of an optical low pass filter (12). Diffraction, therefore, plays an important role in the sensor's sampling capability.

4.3.2 Light speed delay

The distances corresponding to the triangulation errors could be traversed by the observer at it's current speed in the time of about 1 hour. It does not mean that the observer would have reached the position one hour later or that it was at the position one hour before, but is useful to gauge the importance of time in the measurements.

It means that the time it takes for the light to arrive at the observer is relevant. To travel the 280 million kilometers distance from Mars to the Earth present in one of the acquisition dates, light takes 15 minutes. To travel the 1400 million kilometers distance from Saturn to the Earth present in another acquisition date, light would take more than a full hour.

The processing logs of the triangulation also show exactly how much would the apparent position of the planet change if light speed was not taken into account.

4.3.3 Geometric distortion

The triangulation results show roughly one order of magnitude difference between the precision of the centroid measurements (estimated by their standard deviation) and the actual accuracy (estimated by the difference of the detected position of the stars to their expected location from the catalog).

One possible explanation for that result is that the centroid location algorithm is good, but the model is not good enough. It is still unclear if the distortions that the algorithm failed to model to the same accuracy as the centroid finding algorithm originate from the atmosphere or from the optical system.

4.3.4 Noise (centroid finding accuracy)

The performance of the algorithms hinges on the accurate determination of the centroid locations. Noise diminishes the ability of accurately determining the position of a star like body in the image.

A few tests were performed with synthetic images and noise when developing the centroid finding algorithm that relies on a gaussian function fit to a bayer pattern array (see **Section 3.6.2** for a brief description). The test showed that the algorithm degrades gracefully with the addition of noise, but the accuracy decreases with increasing noise levels.

4.3.5 Angular motion of camera (centroid finding accuracy)

The star identification test using blurred images showed that the blur diminishes the centroid finding accuracy and can even prevent the star identification algorithm from working correctly.

If a long exposure is made with a camera in a fixed tripod, the rotation of the earth itself will cause blurred images.

Earth rotates at approximately $7.3 \times 10^{-}5$ rad/s. If a camera with a 3000x4000 pixels sensor and a 30 degree (0.5 rad) horizontal field of view takes an image of the sky with 10 s of exposure, the stars near the celestial equator will have $\times 7.3 \times 10^{-}5 \times 10 \times \frac{4000}{0.5} = 5.8$ pixels of blur.

It is possible to reduce blur by selecting a shorter exposure time or some means to keep the camera pointed in the same direction of the sky during the exposure. Shorter exposure pictures can be merged with specialized software to reduce noise.

4.3.6 Catalog inaccuracies

Errors in the star and planet catalogs would propagate through all the steps of the algorithm. They are not taken into account in this work because the errors for Hipparcos are in the order of $10^{-}8$ radians (53), which is three orders of magnitude less than the errors that the experiments show and much beyond the resolving capability of the sensors and algorithms used.

The author was not able to determine the NASA JPL Horizons catalog (47) uncertainty, as the system provide the value "n.a." in the uncertainty fields. More investigation is warranted on this subject.

4.3.7 Tadeoff between tolerances and field of view for star identification

The star identification algorithm is tuned to match *starprint* components within a certain tolerance. There is a tradeoff between the minimum field of view that the algorithm can operate with and the size of the accepted tolerances.

Although not exactly an error source, this tradeoff limits the capacity of the algorithm to find both the calibration parameters of the camera and the star identifications in narrow fields of view. If large tolerances are accepted to be able to identify cameras with big distortions, then it is not possible to identify stars in pictures taken with narrow fields of view.

The possibility of an ambiguous identification rises with the number of stars in the coverage set and rises with the increase of the tolerances. The number of stars in the coverage set increases with a decrease of the minimum field of view supported, therefore, if one wants to limit the possibility of ambiguities, it is not possible to have both large tolerances and support narrow fields of view.

If the algorithm is to be used with telescope images, for example, it will not be able to determine the calibration of the telescope and still be able to identify the stars. One possible approach would be to tighten the tolerances and rectify the telescope images (cancel distortions) before they are presented to the algorithm.

Chapter 5

Conclusion

A robust technique was demonstrated for star identification in pictures of the sky taken with uncalibrated cameras. That technique also produces the camera attitude relative to the star reference frame and the angular position of the stars and star-like bodies in the picture.

A technique was also demonstrated for location determination based on the output of the first technique, producing results that are within two orders of magnitude of the results obtained with the spacecraft Deep Space 1 (around 1000km according to Bhaskaran (4)) and correspond to angular errors equal to the Rayleigh diffraction limit in the case of the Raspberry Pi camera and an error of one pixel in the case of the Nikon hardware, corresponding to three times the diffraction limit.

Compared to the ground truth obtained from NASA JPL's Horizons database (47), tests with a Nikon D5000 camera with a 50mm lens yielded results with 101km of and tests with a Raspberry Pi NoIR camera yielded results with 169km of error.

The equivalent angular error of the Raspberry Pi optical system is approximately $6 \times 10^{-}4$ radians, which is about the value of $5.4 \times 10^{-}4$ radians calculated for the Rayleigh limit of the Raspberry Pi optical system in **Section 4.3.1** and corresponds to about 1.6 times the angle covered by a single pixel in the Raspberry Pi optical system.

The equivalent angular error of the Nikon optical system is approximately $1.1 \times 10^{-}4$ radians, which is about three times the value of 3.7×10^{-5} radians calculated for the Rayleigh limit of the Nikon optical system in **Section 4.3.1** and corresponds to about the same angle covered by a single pixel in the Nikon optical system.

The measurements performed in this work differ from the measurement performed by Deep Space 1 in that the pictures in this work were taken from inside Earth's atmosphere, the hardware used was low cost and the measurements took only a few minutes, while on Deep Space 1 they took several months and used Kalman filtering to converge to a more accurate solution over time.

The main contribution of this work the implementation and experimental demonstration of an algorithm that can be used for autonomous optical navigation using a very low cost sensor.

5.1 Future work

A fully embedded algorithm can be developed. The current version was not run embedded because the code is mainly written in Python 3, which is not the default Raspbian OS environment. Also code development and

debugging were faster on the Desktop. To accomplish that, Python 3 would have to be installed at the Raspberry Pi OS and interfaces would have to be developed to feed the calculation results through a serial bus that would connect to the spacecraft's main computer.

Other topics of future work can be the publication of the source code of this project, the implementation of temporal filters in the acquisition (offset and average frames), the implementation of a Kalman filter for reducing error of measurement over several days, camera calibration using multiple frames instead of only one, vacuum test and irradiation test of the equipment, deconvolution by using the camera's local PSF and multiple frames, an experiment using mirrors instead of lenses for the optical systems, utilizing truly monochromatic cameras and utilizing a more complete calibration step with flat fielding and spectral calibration.

Daniel Araújo Miranda
Aluno de pós-graduação PPMEC
Matrícula 15/0167695

Prof. Dr. Daniel Mauricio Muñoz Arboleda
Co-orientador
Departamento de Engenharia Mecânica - PPMEC

Prof. Dr. Fabiano Araujo Soares
Orientador
Faculdade de Engenharia - FGA

Bibliography

1 ARIAS, E. F. et al. The extragalactic reference system of the International Earth Rotation Service, ICRS. **Astronomy and Astrophysics**, v. 303, p. 604–608, Nov. 1995. Available from: <https://www.researchgate.net/publication/234492471_The_extragalactic_reference_system_of_the_International_Earth_Rotation_Service_ICRS>.

2 BERLIN Space Technologies Star Tracker ST400 and ST200 specifications. [S.l.: s.n.]. Available from: <https://www.berlin-space-tech.com/portfolio/star-tracker-st400-and-st200/>. Visited on: 14 June 2017.

3 BESSEL, Friedrich-Wilhelm. On the parallax of 61 Cygni. **Monthly Notices of the Royal Astronomical Society**, v. 4, p. 152–161, 1838. DOI: 10.1093/mnras/4.17.152. Available from: <http://adsabs.harvard.edu/abs/1838MNRAS...4..152B>.

4 BHASKARAN, Shyam. Autonomous Navigation for Deep Space Missions. **SpaceOps 2012 Conference**, American Institute of Aeronautics and Astronautics (AIAA), June 2012. DOI: 10.2514/6.2012-1267135. Available from: <http://dx.doi.org/10.2514/6.2012-1267135>.

5 BROWN, Duane C. Decentering distortion of lenses. **Photometric Engineering**, v. 32, n. 3, p. 444–462, 1966. Available from: <https://eserv.asprs.org/PERS/1966journal/may/1966_may_444-462.pdf>.

6 CHANDLER, David. **The Night Sky, Southern Hemisphere (Large) Star Finder**. [S.l.]: David Chandler Co., 1998. ISBN 0-9613207-3-7. Available from: <https://www.amazon.com/Night-Southern-Hemisphere-Large-Finder/dp/0961320737>.

7 CHÉREAU, Fabien et al. **Stellarium - free open source planetarium**. [S.l.: s.n.], 2001–2017. Available from: <http://www.stellarium.org/>.

8 COFFIN, Dave. **dcraw - Decoding raw digital photos in Linux**. [S.l.: s.n.], 1997–2017. Available from: <http://www.cybercom.net/~dcoffin/dcraw/>.

9 CREDNER, Till; KOHLE, Sven. **Capricornus - The Deep Photographic Guide to the Constellations**. [S.l.: s.n.]. Available from: <http://www.allthesky.com/constellations/capricornus/constell.html>.

10 CUMMINGS, Nicholas; EARLE, Kevin, et al. **Solar Orbit Transfer Vehicle**. Department of Aerospace, Ocean Engineering Virginia Polytechnic Institute, and State University Blacksburg, VA 24061 (540) 231-2314 (540) 231-9632 (FAX) cdhall@vt.edu, June 2001. Available from: <http://www.dept.aoe.vt.edu/~cdhall/courses/aoe4065/OldReports/sotv.pdf>.

11 CURKENDALL, David W; BORDER, James S. Delta-DOR: The One-Nanoradian Navigation Measurement System of the Deep Space Network - History, Architecture and Componentry. **The Interplanetary Network Progress Report**, v. 42, p. 193, May 2013. Available from: <http://ipnpr.jpl.nasa.gov/progress_report/42-193/193D.pdf>. Visited on: 5 Jan. 2017.

12 DATASHEET - preliminary specification of the OV5647 sensor. [S.l.: s.n.]. Available from: <http://cdn.sparkfun.com/datasheets/Dev/RaspberryPi/ov5647_full.pdf>.

13 DEEP Space 1 at the NASA Space Science Data Coordinated Archive. [S.l.: s.n.]. Available from: <https://nssdc.gsfc.nasa.gov/nmc/spacecraftDisplay.do?id=1998-061A>.

14 DIGITAL PHOTOGRAPHY REVIEW. **Canon PowerShot S1 IS photographic tests**. [S.l.: s.n.], Aug. 2004. Available from: <https://www.dpreview.com/reviews/canons1is/5>. Visited on: 17 June 2017.

15 _____. **Canon PowerShot S1 IS Review**. [S.l.: s.n.], Aug. 2004. Available from: <https://www.dpreview.com/reviews/canons1is>. Visited on: 17 June 2017.

16 EUROPEAN SPACE AGENCY. **Rosetta - Rendezvous with a comet**. [S.l.: s.n.], Nov. 2015. Available from: <http://rosetta.esa.int/>. Visited on: 30 Nov. 2015.

17 FEY, A. L. et al. The Second Realization of the International Celestial Reference Frame by Very Long Baseline Interferometry. **The Astronomical Journal**, IOP Publishing, v. 150, n. 2, p. 58, July 2015. DOI: 10.1088/0004-6256/150/2/58. Available from: <http://stacks.iop.org/1538-3881/150/i=2/a=58>.

18 GIPSMAN, A.; GUELMAN, M.; KOGAN, A. Autonomous navigation and guidance system for low thrust driven deep space missions. **Acta Astronautica**, v. 44, 7-12, SI, p. 353–364, Sept. 1999. ISSN 0094-5765. DOI: 10.1016/S0094-5765(99)00058-2. Available from: <http://www.sciencedirect.com/science/article/pii/S0094576599000582>.

19 GONZALEZ, Rafael C.; WOODS, Richard E. **Digital Image Processing (4th Edition)**. [S.l.]: Pearson, 2017. ISBN 0133356728.

20 HO, Koki. A survey of algorithms for star identification with low-cost star trackers. **Acta Astronautica**, v. 73, p. 156–163, Apr. 2012. ISSN 0094-5765. DOI: http://dx.doi.org/10.1016/j.actaastro.2011.10.017. Available from: <http://www.sciencedirect.com/science/article/pii/S0094576511003195>.

21 HOFFLEIT, D.; WARREN Jr., W. H. VizieR Online Data Catalog: Bright Star Catalogue, 5th Revised Ed. (Hoffleit+, 1991). **VizieR Online Data Catalog**, v. 5050, Nov. 1995. Available from: <http://cdsarc.u-strasbg.fr/viz-bin/Cat?V/50>. Visited on: 9 Jan. 2016.

22 HUSSEY, Kevin; RODRIGUES, Michael; MCCONNELL, Shannon et. al. **Deep Space Network Now**. [S.l.: s.n.], Sept. 2015. Available from: <https://eyes.nasa.gov/dsn/dsn.html>. Visited on: 24 Sept. 2015.

23 INTERNATIONAL ASTRONOMICAL UNION. **International Celestial Reference Frame**. [S.l.: s.n.], Jan. 2015. Available from: <https://www.iers.org/IERS/EN/DataProducts/ICRS/icrs.html>. Visited on: 12 June 2016.

24 JAXA. **Asteroid Explorer Hayabusa 2 homepage**. [S.l.: s.n.], Nov. 2015. Available from: <http://global.jaxa.jp/projects/sat/hayabusa2/>. Visited on: 30 Nov. 2015.

25 JAXA. **Asteroid Explorer HAYABUSA (MUSES-C) homepage**. [S.l.: s.n.], June 2013. Available from: <http://global.jaxa.jp/projects/sat/muses_c/index.html>. Visited on: 11 June 2017.

26 JENA-OPTRONIK. **Astro APS Star Tracker picture**. [S.l.: s.n.], 7 Dec. 2013. European Space Agency website. Available from: <http://www.esa.int/spaceinimages/Images/2013/07/Development_of_Astro_APS_star_tracker>.

27 JET PROPULSION LABORATORY, California Institute of Technology. **Space Technology 6 - Star Camera**. [S.l.: s.n.], May 2004. Website. Available from: <https://www.jpl.nasa.gov/nmp/st6/TECHNOLOGY/star_camera.html>.

28 JOHN French's Constellation Photos. [S.l.: s.n.]. Available from: <http://www.pa.msu.edu/people/frenchj/const/index1.html>.

29 KARIMI, Reza Raymond. **Designing An Interplanetary Autonomous Spacecraft Navigation System Using Visible Planets**. [S.l.]: Texas A&M University, 2012. Available from: <http://hdl.handle.net/1969.1/ETD-TAMU-2012-05-11084>.

30 KIMBALL, Spencer; MATTIS, Peter, et al. **Gimp - GNU Image Manipulation Program**. [S.l.: s.n.], 1996–2017. Available from: <https://www.gimp.org/>.

31 KLAUS, A. et al. Camera calibration from a single night sky image. In: PROCEEDINGS of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004. [S.l.: s.n.], June 2004. v. 1, p. 151–157. DOI: 10.1109/CVPR.2004.1315026.

32 KUBITSCHEK, Daniel G. et al. Deep Impact Autonomous Navigation: the trials of targeting the unknown. **29th Annual AAS Guidance and Control Conference; February 4-8, 2006; Breckenridge, CO; United States**, Apr. 2006. Available from: <http://www.researchgate.net/publication/228695032_Deep_Impact_Autonomous_Navigation_the_trials_of_targeting_the_unknown>.

33 KUBOTA, T. et al. Guidance and Navigation of Hayabusa Spacecraft for Asteroid Exploration and Sample Return Mission. In: 2006 SICE-ICASE International Joint Conference. [S.l.]: Institute of Electrical and Electronics Engineers (IEEE), Oct. 2006. p. 2793–2796. ISBN 89-950038-4-7. DOI: 10.1109/sice.2006.314761.

34 LEEUWEN, Floor van. **Hipparcos, the New Reduction of the Raw Data**. Ed. by Floor van Leeuwen. [S.l.]: Springer Netherlands, 23 Sept. 2007. ISBN 978-1-4020-6342-8. DOI: https://dx.doi.org/10.1007/978-1-4020-6342-8. Available from: <http://vizier.u-strasbg.fr/viz-bin/VizieR-3?-source=I/311>.

35 LI, Allen. **NASA's Deep Space Network: Current Management Structure Is Not Conducive to Effectively Matching Resources with Future Requirements**. [S.l.], Apr. 2006. Available from: <http://www.gao.gov/products/GAO-06-445>. Visited on: 30 Nov. 2015.

36 LIEBE, C.C. Star trackers for attitude determination. **IEEE Aerospace and Electronic Systems Magazine**, Institute of Electrical and Electronics Engineers (IEEE), 345 E 47TH ST, NEW YORK, NY 10017-2394, v. 10, n. 6, p. 10–16, June 1995. ISSN 0885-8985. DOI: 10.1109/62.387971.

37 LODIOT, Sylvain et al. The first european asteroid 'flyby' - Rosetta operations for the flyby of asteroid 2867 Steins. **ESA Bulletin**, European Space Agency, n. 137, p. 68–74, Feb. 2009. ISSN 0376-4265. Available from: <http://www.esa.int/esapub/bulletin/bulletin137/bulletin137.pdf>.

38 LORENZ, David A. et al. Lessons learned from OSIRIS-REx autonomous navigation using natural feature tracking. In: 2017 IEEE Aerospace Conference. [S.l.]: IEEE. DOI: 10.1109/aero.2017.7943684.

39 MARTINZ, Anita. **Colorful spring garden Bayer.png**. [S.l.: s.n.], July 2013. Wikipedia. Available from: <https://commons.wikimedia.org/w/index.php?curid=27188384>.

40 MONTENBRUCK, Oliver et al. Precision Spacecraft Navigation Using a Low-cost GPS Receiver. **GPS Solut.**, Springer-Verlag New York, Inc., Secaucus, NJ, USA, v. 16, n. 4, p. 519–529, Oct. 2012. ISSN 1080-5370. DOI: 10.1007/s10291-011-0252-6. Available from: <http://dx.doi.org/10.1007/s10291-011-0252-6>.

41 NASA. **Deep Impact - Mission to a Comet**. [S.l.: s.n.], Sept. 2013. Available from: <http://www.nasa.gov/mission_pages/deepimpact/main/>. Visited on: 30 Nov. 2015.

42 NASA. **NASA's Mission Operation and Communication Services**. [S.l.: s.n.], Oct. 2014. Available from: <https://deepspace.jpl.nasa.gov/files/dsn/6_NASA_MOCS_2014_10_01_14.pdf>.

43 NASA JET PROPULSION LABORATORY. **DAWN Journal - Optical Navigation**. [S.l.: s.n.], Jan. 2015. Available from: <https://dawnblog.jpl.nasa.gov/2015/01/29/dawn-journal-january-29/#opticalnav>. Visited on: 5 July 2017.

44 _____. **DAWN Technology - Ion Propulsion System**. [S.l.: s.n.], 2007. Available from: <http://dawn.jpl.nasa.gov/technology/ion_prop.asp>. Visited on: 30 Nov. 2015.

45 _____. **Deep Impact - EPOXI**. [S.l.: s.n.], Oct. 2011. Available from: <http://www.jpl.nasa.gov/missions/deep-impact-epoxi/>. Visited on: 30 Nov. 2015.

46 _____. **Deep Space 1 mission page**. [S.l.: s.n.], Dec. 2001. Available from: <http://nmp.jpl.nasa.gov/ds1/>. Visited on: 30 Nov. 2015.

47 _____. **NASA Horizons Web Interface**. [S.l.: s.n.], Jan. 2016. Available from: <http://ssd.jpl.nasa.gov/horizons.cgi>. Visited on: 8 Jan. 2017.

48 _____. **Stardust: NASA Comet sample return mission**. [S.l.: s.n.], Oct. 2009. Available from: <http://stardust.jpl.nasa.gov/home/index.html>. Visited on: 30 Nov. 2015.

49 NASA LUNAR AND PLANETARY LABORATORY. **OSIRIS REx Mission - Navcams**. [S.l.: s.n.], 2015. Available from: <http://www.asteroidmission.org/mission/#explore-instruments>. Visited on: 30 Nov. 2015.

50 NASA/JPL/CORNELL. **Stars in Orion as Seen from Mars**. [S.l.: s.n.], Nov. 2004. Available from: <http://marsrovers.jpl.nasa.gov/gallery/press/spirit/20040311a.html>.

51 NIKON D5000 camera technical specifications. [S.l.: s.n.], Apr. 2009. Available from: <http://imaging.nikon.com/lineup/dslr/d5000/spec.htm>. Visited on: 16 Jan. 2017.

52 PALUSZEK, Michael; LITTMAN, Michael; MUELLER, Joseph. Optical Navigation System. In: AIAA Infotech@Aerospace 2010. [S.l.]: American Institute of Aeronautics and Astronautics (AIAA), Apr. 2010. DOI: 10.2514/6.2010-3462.

53 PERRYMAN, M. A. C. et al. The HIPPARCOS Catalogue. **Astronomy and Astrophysics**, v. 323, July 1997. Available from: <http://vizier.cfa.harvard.edu/viz-bin/VizieR?-source=I/239>.

54 PICKING, Christopher J. **Southern cross by moonlight**. [S.l.: s.n.], Jan. 2005. Published with permission from the author. Available from: <http://www.starrynightphotos.com/southern_sky/crux_tree.htm>.

55 R. SEDWICK, K. Huffman. **Lightweight, Low Power Inexpensive Star Tracker**. [S.l.: s.n.]. retrieved on Feb 4, 2013. Available from: <http://ssl.mit.edu/research/startracker.html>.

56 RAD, Amir Moghtadaei; NOBARI, Jafar Heyrani; NIKKHAH, Amir Ali. Optimal attitude and position determination by integration of INS, star tracker, and horizon sensor. **IEEE Aerospace and Electronic Systems Magazine**, Institute of Electrical and Electronics Engineers (IEEE), v. 29, n. 4, p. 20–33, Apr. 2014. ISSN 0885-8985. DOI: 10.1109/maes.2014.130093.

57 RASPBERRY Pi Camera Module with NoIR (No Infrared) Filter. [S.l.: s.n.], 2014. Available from: <https://www.element14.com/community/docs/DOC-69389/l/camera-module-with-noir-no-infrared-filter>.

58 RAYLEIGH. XXXI.Investigations in optics, with special reference to the spectroscope. **Philosophical Magazine Series 5**, Informa UK Limited, v. 8, n. 49, p. 261–274, Oct. 1879. DOI: 10.1080/14786447908639684.

59 RIEDEL, JE et al. Autonomous Optical Navigation (AutoNav) DS1 Technology Validation Report. **JPL Publication, JPL, Pasadena CA, USA**, 2000. Available from: <https://pdssbn.astro.umd.edu/holdings/ds1-c-micas-3-rdr-visccd-borrelly-v1.0/document/doc_Apr04/int_reports/AutoNav_Integrated_Report_A.pdf>.

60 RIEDEL, J et al. Using autonomous navigation for interplanetary missions: The validation of deep space 1 AutoNav. **International Conference on Low-Cost Planetary Missions**, 2000. Available from: <https://trs.jpl.nasa.gov/handle/2014/14133>.

61 SHARPE, Louis E. et al. **Design criteria for spacecraft star trackers**. [S.l.], July 1970. Available from: <http://ntrs.nasa.gov/search.jsp?R=19700029405>. Visited on: 30 Nov. 2015.

62 SPRATLING, Benjamin B.; MORTARI, Daniele. A Survey on Star Identification Algorithms. **Algorithms**, v. 2, n. 1, p. 93–107, 2009. ISSN 1999-4893. DOI: 10.3390/a2010093. Available from: <http://www.mdpi.com/1999-4893/2/1/93>.

63 STARNAMERS blog. [S.l.: s.n.]. Available from: <http://starnamer.blogspot.com.br/>. Visited on: 11 July 2017.

64 SYNNOTT, S. et al. Interplanetary optical navigation - Voyager Uranus encounter. In: ASTRODYNAMICS Conference. [S.l.]: American Institute of Aeronautics and Astronautics, Aug. 1986. p. 192–206. DOI: 10.2514/6.1986-2113.

65 TECHNICAL Guide of the IDEX Optics and Photonics Marketplace. [S.l.: s.n.], 2014. Available from: <http://marketplace.idexop.com/store/Forms/GetTechGuideDownload>.

66 THOMAS, Valerie C. et al. The Dawn Spacecraft. **Space Science Reviews**, Springer Nature, v. 163, n. 1-4, p. 175–249, Dec. 2011. ISSN 0038-6308. DOI: 10.1007/s11214-011-9852-2. Available from: <http://dx.doi.org/10.1007/s11214-011-9852-2>.

67 THORNTON, Catherine L.; BORDER, James S. **Radiometric Tracking Techniques for Deep Space Navigation**. Hoboken, N.J. Great Britain: Wiley-Blackwell, 11 Feb. 2003. ISBN 0471445347. DOI: 10.1002/0471728454. Available from: <http://descanso.jpl.nasa.gov/monograph/series1/Descanso1_all.pdf>. Visited on: 5 Jan. 2017.

68 WOOD, Lincoln J. The Evolution of Deep Space Navigation: 1989-1999. Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration, 2008., June 2008. Available from: <http://descanso.jpl.nasa.gov.s3.amazonaws.com/evolution/AAS_08-051.pdf>.