Design and Development of an Automatic Mirror Segment Alignment System for the CTA Telescopes

Master Thesis

Mathematisch-naturwissenschaftliche Fakultät
der

Universität Zürich

Roman Gredig

rgredig@physik.uzh.ch

Universität Zürich UZH

Prof. Dr. Ulrich Straumann
Dr. Achim Vollhardt

Zürich
Oktober, 2011

Abstract

The Cherenkov Telescope Array CTA will be the next generation experiment for the analysis of very high energy cosmic photons. Therefore a set of telescopes is planned to be built on both hemispheres of the earth. Such telescopes consist of a reflective mirror that has to be tessellated because of economic reasons and to correct deformations during measurement. The mirror facets must be aligned to a desired orientation either with the help of motors or manually.

This thesis evaluates the usage of a digital camera system to properly align the mirror facets of the telescopes. A simulation framework has been written that can be used to estimate the optical requirements fulfilling the specified alignment precision. A test environment has been set up that was used to develop and evaluate automatic alignment algorithms. Such an algorithm has been developed in `C` that can be executed directly on the camera. The algorithm has been tested with an artificial light source and the simulation has been verified with a celestial source.

Acknowledgments

I thank Prof. Ueli Straumann for giving me the opportunity to write this thesis in his group. Special thanks goes to my tutor Dr. Achim Vollhardt who was an excellent supporter giving me the chance to let me gain my own experience but being always there when I stuck or ran into a complete wrong direction. A big thank goes to Stefan Steiner for his help and offering me always his expertise in mechanical problems. I also want to thank Arno Gadola for the fruitful discussions leading to important inputs to my work. Further I would like to thank Dr. Aaron Manalaysay for his comments and corrections. It was a pleasure for me being in a group where everyone was open for my questions.
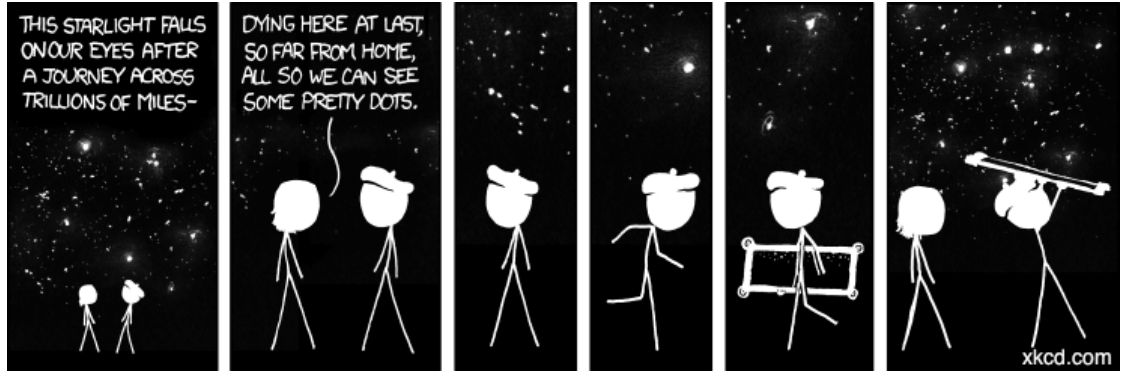
Contents

1. Introduction

1.1. Gamma Ray Astronomy

Gamma ray astronomy studies celestial objects by the observation of high energy photons. The energy domain of γ-ray astronomy spans over a region from about 10^6 eV up to energies exceeding 10^{20} eV. This enormous range of energies can be divided into different areas as listed in Tab. 1.1.

Table 1.1.: classification of the different γ-ray energies

energy type	abbr.	range	dectector type
low	LE	≤ 30 MeV	satellite/air-borne
high	HE	30 MeV $- 30$ GeV	satellite/air-borne
very high	VHE	30 GeV $- 30$ TeV	ground-based
ultra high	UHE	30 TeV $- 30$ PeV	ground-based
extremely high	EHE	> 30 PeV	ground-based

Observations of γ-rays are a promising method for the analysis of galactic and extra-galactic cosmic radiation. They are not only proposed to be generated by accelerated charged particles, but they also are the key in localizing cosmic ray accelerators. As γ-rays are not charged, they can freely propagate over long distances (with the exception of gravitational lensing) without being deflected by magnetic fields. For this reason γ-

radiation can be used to understand and localize the origin of charged cosmic particles that are hitting the earth from all directions due to the deflection by magnetic fields.

1.2. γ-ray Sources and Acceleration Mechanisms

Reaching high energies up to the 10^{20} eV scale, these γ-rays cannot be of thermal origin. Therefore γ-rays are mostly a byproduct of accelerated charged cosmic particles. Production mechanisms for γ-rays in cosmic accelerators are:

- Collisions of accelerated protons with an ambient nucleus, leading for instance to the production of π^0-mesons decaying in two gammas

- High energy electrons emitting γ-rays via bremsstrahlung or synchrotron radiation

- Inverse Compton scattering

Acceleration mechanisms for these "primary" particles and γ-ray sources are for example [1, 2]:

supernova remnants	The shock wave propagating from a supernova explosion can accelerate particles up to the 10^{15} eV range.
pulsars	Pulsar winds generating shockwaves accelerating particles
molecular clouds	Interactions of cosmic rays with the material of cosmic clouds producing γ-rays
starburst galaxies	Starburst galaxies as sources of γ-rays due to the high star-formation rates and the resulting stellar winds
black holes	jets originating from black holes are sources of accelerated particles

1.3. The CTA Project

The Cherenkov Telescope Array (CTA) [3] is a proposed ground based facility to observe high energy γ-rays. It is planned as an array of up to three different telescope sizes with in total up to 100 separate telescopes. The CTA project can be seen as the successor of the *MAGIC* [4], *VERITAS* [5] and *H.E.S.S.* [6] telescopes. The goal is to observe cosmic γ-rays with energies up to hundreds of TeV. The different telescope sizes (Fig. 1.1) cover different parts of energy bands of the γ-ray spectrum. The biggest telescopes will have diameters of about 23 meters, the smallest ones of about 6 meters. Most likely CTA will consist of three different telescopes [3, 7]:
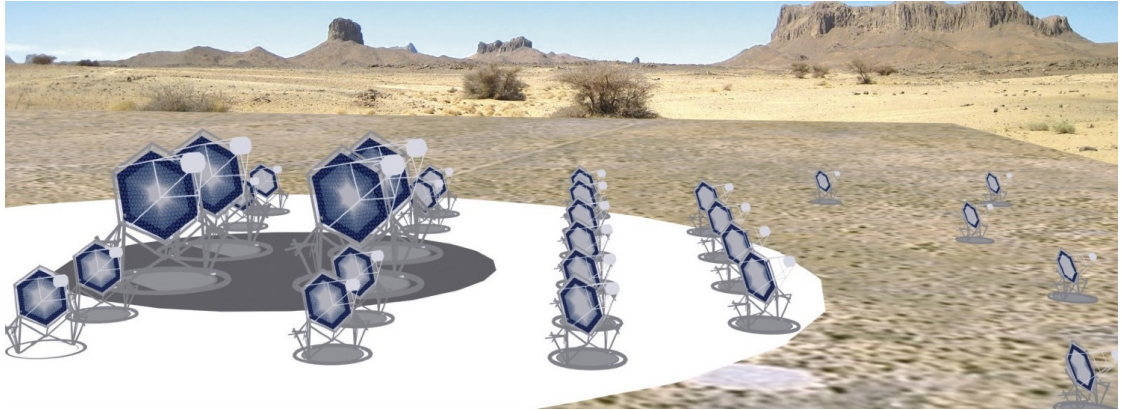
Figure 1.1.: Artistic view of the Cherenkov Telescope Array [3].

1. The large size telescopes (LST) covering energies starting at a few tens of GeV up to about 100 GeV.

2. The middle size telescopes (MST) for the 100 GeV to 10 TeV range, being the core energy of CTA

3. The small size telescopes (SST) for the energies higher than 10 TeV

The higher the energy of a γ-ray, the more Cherenkov light is produced. But at higher energies the γ-ray fluxes are lower. This leads to the following configuration: For the lowest energies the biggest mirrors are needed to focus enough light. For higher energies the mirror can be smaller, because there are more photons per event. The number of telescopes has to be increased for higher energies to compensate the smaller fluxes.

1.4. Scientific Motivation

CTA can help to investigate accelerating mechanisms in pulsars. With higher sensitivity than current experiments for the sub-50 GeV scale, CTA can improve the observation of active galactic nuclei (AGN) (Fig. 1.2).

At energies above 50 TeV CTA will help to understand acceleration mechanisms in objects like supernova remnants (SNR). The most important achievement with CTA will be the increased sensitivity in the energy region of $0.1 - 50$ TeV. For the first time a full very high energy (VHE) survey will be possible. About a thousand new γ-ray sources are expected to be discovered. The increased sensitivity reduces the required observation time, leading to increased time-resolution. Current telescopes can only resolve variations on a time scale of minutes. CTA will be the first experiment that allows sub-minute time resolution. Improved angular resolution down to $0.02'$ will help to resolve close sources. [9, 10]
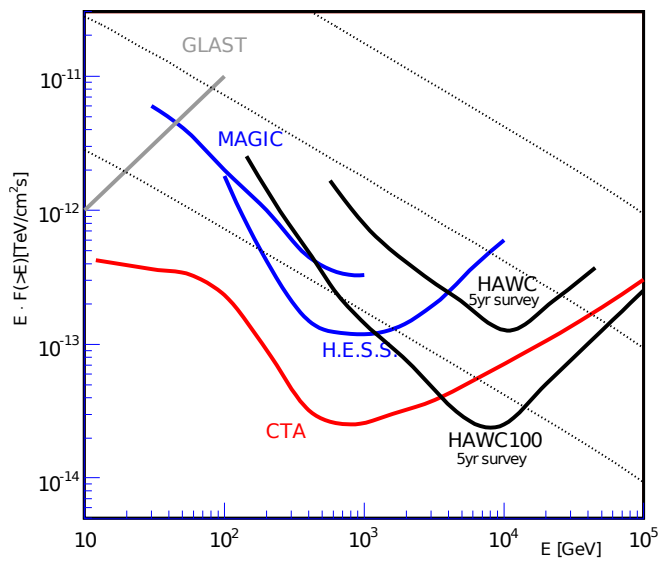
Figure 1.2.: Sensitivity aimed with CTA in comparison to existing facilities. The x-axis shows the energy of the incoming particles, the y-axis the flux multiplied with the corresponding energy. The lines indicate minimal fluxes needed to detect the origin source. The final sensitivity depends on how the system will be realized finally. [8]

2. Imaging Atmospheric Cherenkov Technique

2.1. γ-ray Detection

2.1.1. Air Shower

When a high energy photon hits the atmosphere, an electromagnetic shower is generated: the photon produces an electron / positron-pair within the coulomb field of an atmospheric nucleus. The produced lepton pairs can again emit photons via bremsstrahlung. This process is repeated until the energy loss of the particle is dominated by ionization rather than bremsstrahlung. This is the point where the particles are called "absorbed" by the atmosphere (Fig. 2.1).
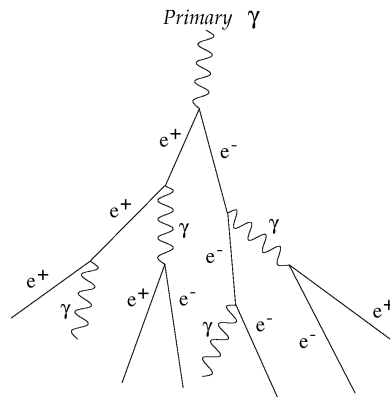
Figure 2.1.: Development of an electromagnetic shower.

Originating from a high energetic photon, the produced secondary leptons propagate at relativistic velocities ($\beta \approx 1$). Having a higher velocity than the speed of light in the atmosphere ($= \frac{c}{n}$) the charged leptons produce light due to Cherenkov radiation.

The development of the shower depends on the type of initial particle that enters the atmosphere. If the inciding particle is a photon or electron, the shower consists only of electron / positron pairs and photons. If the initial particle is a nucleus (mostly a proton), the shower is more complicated. For γ-ray astronomy it is important to identify hadronic showers as themselves being. Due to the strong interaction, more processes

can happen in hadronic showers than in electromagnetic showers. If a hadron hits an atmospheric nucleus, particles like pions and kaons can be produced. As long as the secondary particles have enough energy, they can interact with other nuclei producing more particles. The most produced particles are pions. The π^0 decaying in two gammas $(\pi^0 \rightarrow \gamma + \gamma)$ produces two electromagnetic sub-showers.

Due to the wider range of reactions, including decaying nuclei and spallation, the lateral distribution of hadronic showers is wider than that of pure electromagnetic showers. The lateral momenta of particles in those showers are on average 1000 times higher than in electromagnetic showers [11], being one feature to distinguish different shower types. Photons created in hadronic showers can produce electromagnetic sub-showers. It is important to identify them as sub-showers originating from a hadron. Fig 2.2 shows a typical shower evolution.

2.1.2. Cherenkov Radiation

A charged particle that passes an insulating dielectric medium polarizes the ambient atoms near the particle's track. The moving of charge in the medium generates electromagnetic waves. In general the waves vanish due to destructive interference. If a charged particle passes the medium with $\beta > \frac{1}{n}$ there is a common wavefront without destructive interference. This wavefront can be seen as Cherenkov light[1]. The Cherenkov photons produced by cosmic rays can undergo different absorption mechanisms while traveling to the ground. The resulting spectrum (measured at ground level) typically shows intensities in the visible and near UV spectrum, peaking around 330 nm.

2.1.3. Imaging Air Cherenkov Telescopes IACT

IACTs are telescopes designed for the Cherenkov radiation detection. The basic idea of such a telescope is to concentrate the emitted Cherenkov light on the focal plane of a big mirror where a camera detects the light. Usually the camera is built of an array of photon multiplier tubes (PMTs), each representing a pixel. The (ideal) telescope projects rays of the same incident angle at one point in the focal plane. Using more than one telescope the origin of the air shower in the atmosphere can be calculated. The energy of the incoming γ-ray can be determined from the amount of collected light by the photomultiplier tubes. The working principle of an IACT is shown in Fig. 2.3.

---

[1]named after Pavel Alekseyevich Cherenkov

50 GeV photon
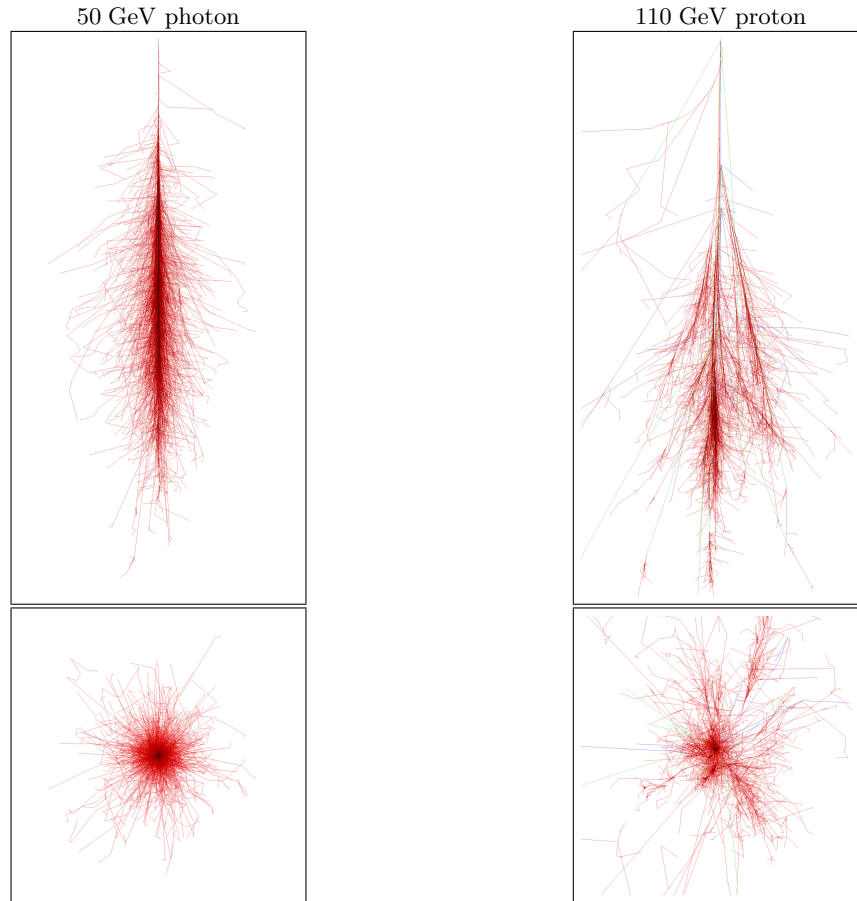
110 GeV proton

Figure 2.2.: Comparison of air-showers, the images on the left side show the shower of an incident photon at 50 GeV in the x-z and the x-y projection. The images on the right side show the shower of a 110 GeV incoming proton. This images are simulations done with the COsmic Ray SImulations for KAscade software *CORSIKA* [12]. red = $e^{\pm}$ and γ, green = μ, blue = hadrons

14

Principle of the Imaging Technique

top of atmosphere

primary particle

first interaction with nuclei of atmosphere at about 20 km height

$\theta_c \approx 0.5°$

~5 km

~20 m $\theta_c \approx 0.8°$

$\theta_c \approx 1.0°$

Cherenkov ligth emission (under characteristic angle: θ_c)

~8 km asl.

camera image

camera

Cherenkov Telescope

Figure 2.3.: Working principle of an IACT: The colors show the different projection on the focal plane for different incident angles. The bigger the incident angle of the Cherenkov light is (respecting the optical axis) the farther away from the camera center they will be projected. [13]

15

2.1.4. Image Formation and Analysis

An example of an image recorded by an IACT camera is shown in Fig. 2.4. A typical γ-ray shower produces an elliptical image in the camera plane. The orientation of the ellipse shows the incident angle of the shower axis. Using the positions of the telescopes the lateral position of the shower axis can be easily reconstructed by *stereoscopic projection*. The analysis 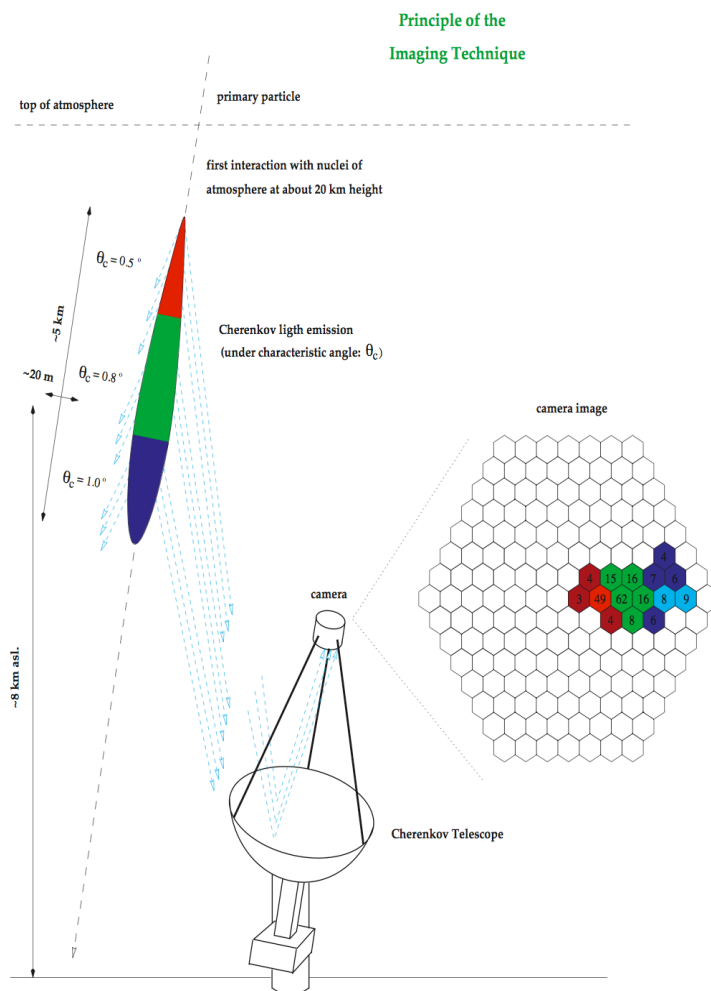of the shape of the ellipse shows the type of shower. Because of the significantly smaller lateral distribution of an electromagnetic shower, a γ-ray produces a rather narrow and long ellipse. A hadronic shower produces a more circular image. One way to analyze the picture has been introduced by A. Hillas [14] (see Fig. 2.4):

> "The image axis minimizes the signal-weighted sum of squares of perpendicular angular distance of the detectors. [...] The r.m.s. spread of light in directions parallel and perpendicular to this axis are referred to as the *length* and *width* of the image."
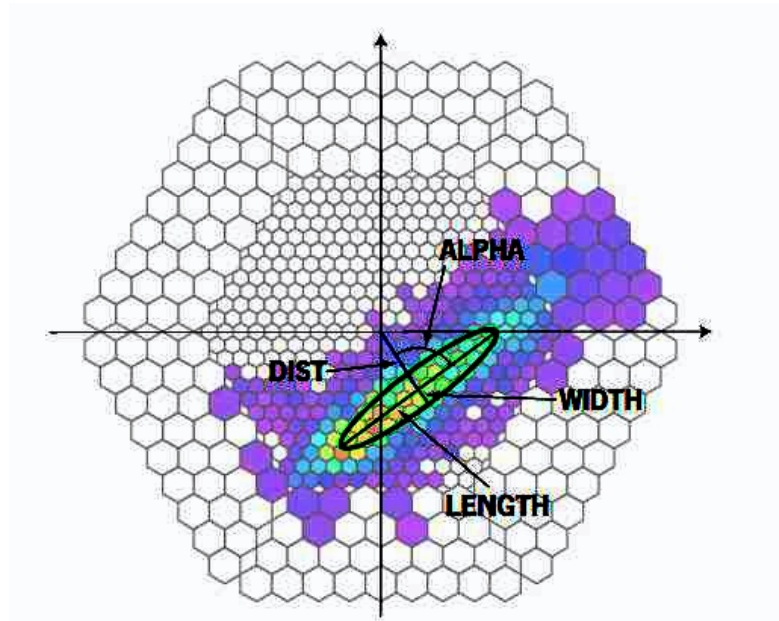

Figure 2.4.: Definition of the Hillas parameters [15]

Defining I_i as the number of photoelectrons detected by pixel i and $S := \sum_i I_i$, the first order moments read [16]

$$\langle x \rangle = \frac{\sum_i I_i x_i}{S} \quad \text{and} \quad \langle y \rangle = \frac{\sum_i I_i y_i}{S} \tag{2.1}$$

the second order moments are:

$$\left\langle x^2 \right\rangle = \frac{\sum_i I_i x_i^2}{S}$$
$$\left\langle y^2 \right\rangle = \frac{\sum_i I_i y_i^2}{S} \qquad (2.2)$$
$$\langle xy \rangle = \frac{\sum_i I_i x_i y_i}{S}$$

This leads to the light distribution

$$\sigma_{xx} = \left\langle x^2 \right\rangle - \langle x \rangle^2$$
$$\sigma_{yy} = \left\langle y^2 \right\rangle - \langle y \rangle^2 \qquad (2.3)$$
$$\sigma_{xy} = \langle xy \rangle - \langle x \rangle \langle y \rangle$$

With the following definitions

$$k = \sigma_{yy} - \sigma_{xx}, \; l = \sqrt{k^2 + 4\sigma_{xy}^2}, \; a = \frac{k+l}{2\sigma_{xy}} \qquad (2.4)$$

the width W and length L of the ellipse can be calculated as

$$W = \sqrt{\frac{\sigma_{yy} + a^2 \sigma_{xx} - 2a\sigma_{xy}}{1 + a^2}}$$
$$L = \sqrt{\frac{\sigma_{yy} + a^2 \sigma_{xx} + 2a\sigma_{xy}}{1 + a^2}} \qquad (2.5)$$

With the definition of the ellipse parameters, boundary conditions can be found that determine whether an image was produced by a γ-ray event or not. For γ-ray showers the width W is lower because of the smaller lateral distribution compared to hadronic showers, the length L is shorter because electromagnetic showers have a shorter interaction length than hadronic showers. This parameters can be defined with Monte Carlo studies simulating hadronic and electromagnetic showers. Additional parameters can be specified like the distance

$$D = \sqrt{\langle x \rangle^2 + \langle y \rangle^2} \qquad (2.6)$$

that can be used for the determination of the impact point. The angle α between the semi-major axis of the ellipse and the line between the center of the camera and the ellipse should be close to zero for showers along the telescope axis.

Fig. 2.5 shows how the shower parameters can be extracted from IACT pictures. The superposition of the pictures coming from different telescopes can be used to calculate the incident angle of the shower axis (and therefore for the trajectory of the γ-ray). The
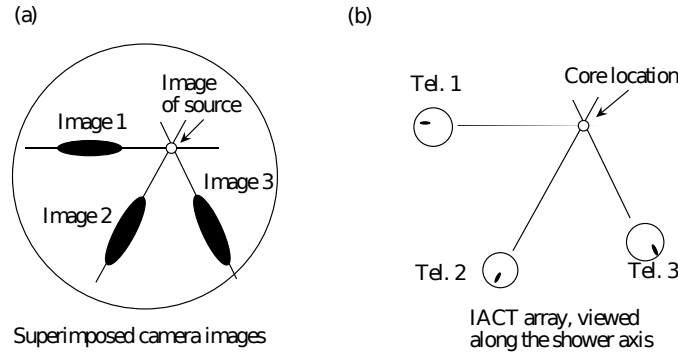
Figure 2.5.: a) Superimposed pictures from different telescopes, showing the incident angle. b) following the image axis, the intersection point shows the location of the shower core [17].

angle between the telescope and the shower axis can also be calculated from the angle between the image axis and the axis from the center of the camera to the image core.

The intersection of the image axes respecting the telescope positions show the position of the shower axis. This can be cross-checked with the distance between the image core and the camera center which gives the distance between the telescope and the incident point. Additional information on image analysis can be found in [17, 18].

2.2. Optical Design Parameters

In general IACTs are light collectors. Optical layouts like the prime-focus telescope or the Schwarzschild–Couder design are under consideration for CTA (Fig. 2.6). This work will concentrate on the prime-focus design as it is the major candidate for all telescope sizes planned for CTA.

The prime-focus design has *one* mirror and a camera in the focal plane. Considering this kind of telescope there are two main properties that have to be discussed [21]: First, the shape of the main dish and second, whether a single piece mirror or a tessellated mirror is used. The usage of tessellated mirrors opens a wider range of configuration parameters. A spherical shape is the simplest way to realize a telescope. Spherical mirrors suffer from spherical aberration[2] which leads to a worsening of the optical quality even for rays with small incident angles. Under ideal conditions a parabolic shape leads to a perfect imaging for on-axis incident rays. But parabolic mirrors produce optical aberrations like coma[3] and astigmatism[4].

The imaging quality is not the only criterium a telescope needs to be optimized for.

---

[2]Rays hitting the edge of a mirror or lens are not focused to the same position than rays hitting the centre.

[3]Off-axis parallel incident rays are not focused to one point.

[4]Rays propagating in two perpendicular planes are focused to different focal points.
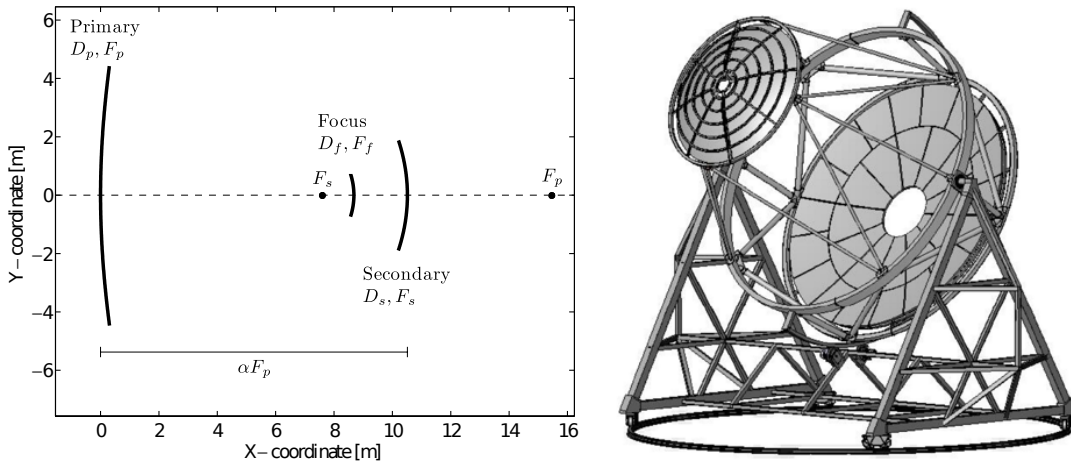
Figure 2.6.: Besides the well known prime-focus designs there is another candiate for CTA: the Schwarzschild–Couder telescope. This design allows to have smaller cameras that do not have to be protruding from the main dish with such large distances as needed for prime-focus designs [19, 20].

Another important property is the timing. The time needed by a photon to travel from the telescope to the camera should not depend on where it hits the telescope. This plays an important role for the separation of background signals. The parabolic shape is here the perfect candidate because it is isochronous.

Tessellated Mirrors

The regular way to build large prime-focus designs is tessellating. In this case the main dish is composed of up to hundreds of mirror facets which form the desired dish shape of the telescope. There are different reasons why the main dish is segmented to mirror facets instead of using one big mirror: it is extremely difficult to produce and handle such a big mirror (which means that it would be extremely expensive), imperfections in the shape can be corrected when the mirror segments are individually tiltable, gravity-induced sagging of the telescope structure and displacement of the camera during measurement can also be corrected if the tilting of such facets is done automatically.

From a physical point of view the most interesting feature is that tessellating opens up new ways of configuration. With a tessellated telescope, the normals of the mirror segments do not have to coincide with the normal of the shape whereon the mirrors are mounted (Fig. 2.7). The most prominent candidate that makes use of this principle is the Davies–Cotton design [22]. In this configuration the mirror are placed on a sphere with radius r. But the center normal of each facet is pointing to a point of the distance $2r$ on the optical axis. The mirrors have a curvature radius of $2f$, with $f = r$. This design or a derivative of it are very promising candidates for the CTA telescopes. The
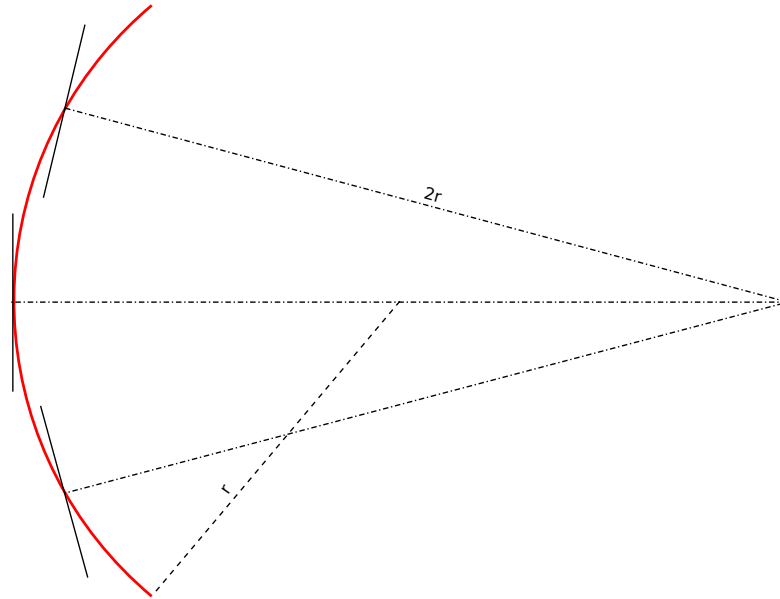
19

Figure 2.7.: Principle of the Davies–Cotton design. The normal of the mirror orientation is different than the normal of the shape on which they are mounted.

Davies–Cotton design provides the best imaging properties for off-axis incident rays and has still a very low time-spread respecting different incident positions of the photons on the mirror.

2.3. Automatic Mirror Alignment

Knowing the desired shape and orientation of the main mirror and its facets is the first step in optimizing the telescope. But regardless of whether the alignment is done manually by turning screws or automatically with an active mirror control (AMC), and whether it is done only once after mounting the mirrors or frequently to correct deformations of the telescope, one has to know *how* the mirrors have to be adjusted. For the LST the deformations are big enough that it is required to have an AMC. Also for the MST the usage of an AMC is under discussion, even though it is not clear whether it will be used for only a initial adjustment or for frequent correction of telescope deformations.

This thesis investigates an automatic adjustment of the mirror facets with the usage of a CMOS or CCD digital camera in the focal plane. The idea is to have a fast and reliable system that allows the mirror positions to be adjusted as often as needed to achieve always the best possible optical quality, also taking deformations of the telescope during observation into account.

This work was done in three major steps. First, a simulation tool has been developed

that allows to estimate the illumination of the camera while changing different telescope parameters. Second, software that allows to test the alignment algorithms has been developed and third, a test-setup using this software has been built to analyze the performance.

3. Auto Alignment Approaches

When the telescope is equipped with an AMC, it would be worthwhile to have an automatic positioning of the mirrors. This could be either done only once and then stored in look-up tables or be done in real-time while the telescope is operating. When a real-time positioning is approached, it has to be fast and reliable.

3.1. Different Alignment Approaches

3.1.1. Light Source at the Sky

Digital Camera in the Focal Plane

One approach to align the mirror segments is with the help of an object in the sky. This can either be a star or an artificial star created with a laser[1] [23]. If the alignment is done only once without taking care of elevation depending deformations there is only one source needed. If the shape needs to be optimized with the respect of the different deformations, a source for all desired angles is needed. The working principle is the following: the telescope needs to be directed to the elevation angle at that an alignment is needed. If there is a bright enough star at that angle, it can be used for the alignment. Otherwise an artificial source is needed.

In both cases a digital camera is placed in the focal point of the telescope, looking in the direction of the dish. The camera needs to see the whole dish with all the mirror-segments on it. In this setup the camera sees always all segments, but depending on their orientation, the mirror segments themselves are either dark or bright. An automatic alignment can be achieved by searching for the position where the mirror segment looks the brightest in the camera.

Two different possibilities exist to scan around all possible mirror-segment orientations to find the optimal one. The first one has been developed to improve the properties of the VERITAS telescope [24]. This telescope is not equipped with an AMC. The telescope is pointed to a star and then scanned around this star with the whole dish. A perfectly aligned mirror would appear bright when the telescope is pointed directly to the star, misaligned mirrors are brightest when the angle between the ideal pointing and the position when the mirror is brightest is twice the misalignment angle.

---

[1]Laser Guide Star

Not being equipped with AMCs, the misalignment is determined with the described procedure and the mirror segments are corrected manually. This improves the overall point spread function, but it is not a dynamic adjustment. The corrections made must be then acceptable for all elevation angles.

With an AMC-equipped telescope the second possibility to align the mirror segments is to point the dish to a star and then scan with the mirror-segments themselves and not with the whole telescope. Because the camera sees all the segments, the scanning can in principal be done for all the segments simultaneously. It has also to be taken into account how time consuming the raster scanning is. This obviously depends on how big the scanning interval is and with what kind of resolution the scanning takes place. It seems to be the most critical point for realizing a real-time alignment.

Digital Camera Recording the Focal Plane

The mirrors of the H.E.S.S telescopes are also aligned with the help of starlight [25]. In this approach the camera is not taking pictures of the mirror facets. When the lid of the Cherenkov camera is closed, the telescope points to a star. All mirror facets collect the light and project it to the focal plane. A camera takes pictures of the focal plane whereon a light spot from every mirror facet is seen. If a mirror facet is turned, the camera can observe the movement of one of all the light spots. The individual movement is needed to assign the light spots to the different mirror facets. Afterwards the facets are moved in a way that all spots concentrate in the center of the Cherenkov camera.

3.1.2. Laser Based Alignment

A different system for the facet alignment is used for the MAGIC telescopes [26]. Every mirror-segment is equipped with a laser pointing to the focal plane where four LEDs are placed as a reference. A camera that images the focal plane from where it sees the spot of the laser relative to the LEDs. The position of the laser spot corresponds to the orientation of the mirror. The advantage of this system is that there is no additional shining source (e.g. a star) needed. On the other hand, it cannot be used to align the mirror-facets simultaneously as the camera cannot assign different laser spots to the corresponding mirror.

This system works only as long as the laser diode is fixed relative to the mirror position. A drift in the orientation of the laser diode cannot be discovered and leads to an active misalignment of the involved facet.

3.2. Investigated Strategy

This work evaluates the approach where the camera is positioned in the focal plane looking at the mirror dish. Compared to the way it is done for VERITAS, it takes the

advantage of having an AMC. With an AMC the system can align all the mirror facets simultaneously and, if needed, a realignment of the facets can be done with an immediate feedback of the result.

It is an approach that has not been systematically used or tested until now, and it is also a promising candidate for future telescopes because of its simplicity and the possibility to upgrade existing telescopes with rather small effort making it possible to test it on existing experiments.

4. Simulations

4.1. Intensities on a Detector

For simulating the response of a detector (i.e. CCD or CMOS) the intensity on the chip coming from a given source has to be calculated. This section describes the theoretical calculation of the illuminance on a detector. The simulation takes account for the brightness of a light source, the optical quality of the mirror facets and the type and quality of the camera lens. The simulation can determine the illumination for a whole telescope configuration (cf. Sec. 4.3.5) but is also used to determine the expected intensities for a test stand, that has been built (Sec. 5).
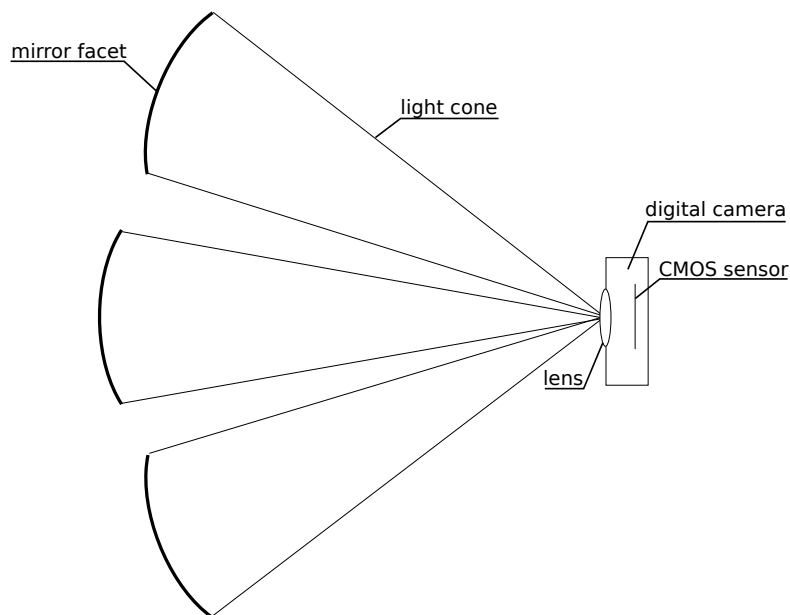
Figure 4.1.: Main parts for the mirror alignment: The light reflected by a mirror facet of the telescope dish is captured with a CMOS camera.

4.1.1. The Measurement of Light

The measurement of light has to be distinguished between radiometry that studies the measurement of electromagnetic radiation (including visible light) and photometry dealing with the brightness perceived by the human eye. For each radiometric unit there is a photometric pendant weighted with the luminosity function. Tab. 4.1 gives a comparison between the two units.

Table 4.1.: Comparison of photometric and radiometric units

Radiometric quantity	Unit	Photometric analogon	Unit
radiant energy	J	luminous energy	lumen second
radiant flux	W	luminous flux	lumen
radiant intensity	W/sr	luminous intensity	candela
irradiance	W/m^2	illuminance	lux

In astronomy the luminance of stars can be classified with the *apparent magnitude*. That is a measure of the brightness of a star seen from the Earth. For the calculations in this simulation, the apparent magnitude has been chosen as the initial parameter. Planning to use stars as light sources for the adjustment makes it very easy to estimate the camera illumination for a specific star because apparent magnitudes are well catalogued for the stars.

4.1.2. Overview about the Illuminance Calculus

The difference between two apparent magnitudes is defined by

$$m_1 - m_2 = -2.5 \cdot \log_{10}\left(\frac{f_1}{f_2}\right) \tag{4.1}$$

where m_i are the apparent magnitudes and f_i the illuminances. To get an absolute value for the illuminance of a specific object, a known reference is needed. This thesis uses the value of the sun, with $f_s = 134\,108$ lx[1] and $m_s = -26.8$ [27]. The illuminance for another object is then:

$$f = f_s \cdot 10^{\frac{m_s - m_2}{2.5}} \tag{4.2}$$

The easiest way to calculate the illuminance on the chip is done via the conservation of energy. With (4.2) we know the illuminance of a star at a specific magnitude. The next step is the collection of the light with a concave mirror. The collecting power is then $f \cdot A_{\mathrm{mir}}$, with A_{mir} being the light collection area of the mirror. The lens of the camera is in the focal plane of the mirror. Ideally the collected light of the mirror would be

---

[1]lx is the symbol for the illuminance lux

point-like, but the mirror is not ideal so the point is smeared out. The appearance of the spot is described by the *Point Spread Function* (cf. 4.1.3). Regarding the quality requirements in [28], 90% of the light is to be in an area of 0.6 mrad. With a focal length of about 16 m, this results in a spot with about 1 cm diameter.

The amount of light collected with the camera lens is proportional to the aperture radius squared. This has to be taken into account when the spot is bigger than the lens itself. This means that not all light collected by the mirror will pass the lens. Finally, the illumination on the sensor is given by the passed light divided by the area of the imaged mirror on the sensor.

4.1.3. The Point Spread Function

For further calculations it is important to describe the Point Spread Function (PSF) (i.e. the description of the light spot in the focal point). In a first approximation the PSF — or more precisely the illumination — is described by a two dimensional Gaussian distribution. The requirements for the mirror segments are described in [28]. It is specified that 90% of the collected light has to be within an angle of 0.6 mrad. With the assumption of a focal length of 16 m this results in a circle with a radius of $r = 0.6\,\text{mrad} \cdot \frac{f}{2} \approx 5\,\text{mm}$. This is sufficient to determine the parameters of the distribution. In general the two-dimensional rotational symmetric distribution reads:

$$f(x,y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4.3}$$

Now we transform into polar coordinates and from the specification we know that the integration to r should give 0.9:

$$\int_0^{2\pi} \int_0^r \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{r^2}{2\sigma^2}} r \; \mathrm{d}r \; \mathrm{d}\varphi = 0.9 \implies \sigma = 0.25\,\text{cm} \tag{4.4}$$

This gives us a parametrization of the PSF.

4.1.4. Light Cone of the Mirror

For an ideal mirror the reflected light of a mirror facet has the shape of a cone with the apex in the mirror's focal point. But the mirror is not perfect. This means that in the focal plane there is a gaussian distribution of the light instead of a infinitesimal point (cf. Sec. 4.1.3). When the lens of the camera is moved out of the focal plane, not all of the light is captured anymore (Fig. 4.2).
The position of the center of the light distribution depends on the mirror's orientation. A tilting of the mirror around the angle ϕ changes the direction of the outgoing rays around the angle $2 \cdot \phi$. For the situation where the lens is in the focal plane the passage
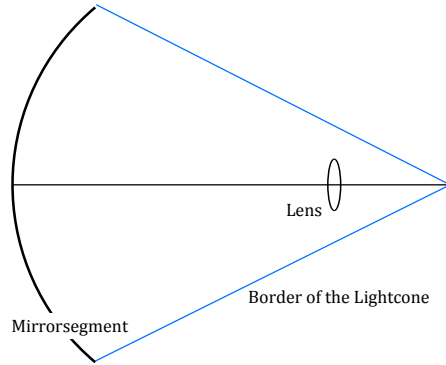
27

Figure 4.2.: Reflected light of a mirror segment in the ideal case. As the lens is not in the focal plane, only a part of the light is passing it.

of the light through the lens can be calculated by the following integral (coordinates in the lens system)

$$\iint_{\text{effective lens area}} f(x, \sigma_x, \mu_x, y, \sigma_y, \mu_y)\, \mathrm{d}A \tag{4.5}$$

Calculating the μ_x and the μ_y in the coordinate system of the lens is done like in a gnomonic projection. Defining φ as the tilting angle of the mirror along the x-axis in the lens-system it follows:

$$\mu_x = f_{\text{mirror}} \cdot \tan(\varphi) \tag{4.6}$$

Tilting along the other axis leads to a moving of the center along the y-axis:

$$\mu_y = f_{\text{mirror}} \cdot \tan(\vartheta) \tag{4.7}$$

Since the σ is given by the quality of the mirror (4.4), rotational symmetry is assumed

$$\implies \sigma_x = \sigma_y = \sigma \tag{4.8}$$

The amount of light passing the lens can now be calculated (in the lens-system)

$$\int_{-r_{\text{lens}}}^{r_{\text{lens}}} \int_{-\sqrt{r_{\text{lens}}^2 - x^2}}^{\sqrt{r_{\text{lens}}^2 - x^2}} f(x, \mu_x, y, \mu_y, \sigma)\, \mathrm{d}y\, \mathrm{d}x \tag{4.9}$$

Non optical axis mirror   The light of the mirrors that are not on the optical axis (in fact all mirrors but only the one in the center) has a non-perpendicular incidence on the plane of the lens. This is equivalent to having an elliptical aperture of the lens. If r_{lens} is the radius of the lens, the semi-axes of the ellipse a and b are calculated by the following equations:

$$a = r_{\text{lens}} \cos \vartheta \quad \text{and} \quad b = r_{\text{lens}} \cos \varphi \tag{4.10}$$

with the sperical angles between the lens and the mirror φ and ϑ. The case that the lens is not placed perpendicular to the optical axis also causes an elliptical aperture effect. This leads to new integration boundaries:

$$\int_{-a}^{a} \int_{-\sqrt{1-\frac{x^2}{a^2}b^2}}^{+\sqrt{1-\frac{x^2}{a^2}b^2}} f(x, \mu_x, y, \mu_y, \sigma)dydx \qquad (4.11)$$

The fact that the point spread function is also distorted because of the non-perpendicular incidence has been neglected.

Lens placed out of the focal plane   Eq. (4.3) is only valid as long as the lens is placed in the focal plane of the observed mirror segment. This is not the case as soon as the lens is moved in the z-direction of the telescope or regarding all the non-centered mirrors. Especially with the Davies–Cotton-like approach for the MST, the lens is not in the focal plane of all mirrors that are not on the optical axis.

The projection of the spot has been chosen in such a way, that the intensity decreases with the same σ at both sides but has a flat center. The intensity function needs to be normalized that the integration over the whole area equals one. Fig 4.3 shows the shape of the function.
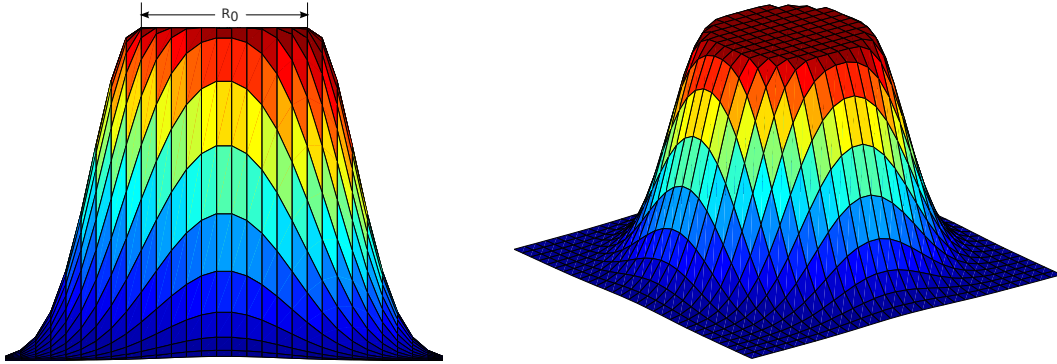

Figure 4.3.: Plots of the light distribution function of the light-cone (a.u.)

The light intensity function changes from the form given in Eq. 4.3 to one that is constant within a specific radius R_0 and slopes like the gaussian distribution on the outer part. R_0 is given by the light-cone of the mirror:

$$R_0 = \frac{r_{\text{mirror}} \cdot |\text{lensposition}_z|}{f_{\text{mir}}} \qquad (4.12)$$

By expanding the distribution function by this constant "plateau", it needs to be renormalized to ensure that the overall integration still equals 1 ($\Rightarrow$ scalefactor a). Switching

to polar coordinates facilitates this:

$$f(r, \varphi) = \begin{cases} \frac{a}{2\pi\sigma} & \text{for } r < R_0 \quad (1) \\ \frac{a}{2\pi\sigma} \cdot \exp\left(-\frac{(r-R_0)^2}{2\sigma^2}\right) & \text{for } r \geq R_0 \quad (2) \end{cases} \tag{4.13}$$

Normalization means:

$$\int_0^{2\pi} \int_0^\infty f(r, \varphi)\, r\, \mathrm{d}r\, \mathrm{d}\varphi = 1 \tag{4.14}$$

Using (4.13) as the integrand leads to

$$\underbrace{\frac{R_0{}^2 \cdot a}{2\sigma^2}}_{(1)} + \underbrace{\frac{a}{\sigma^2} \cdot \int_{R_0}^\infty r e^{-\frac{(r-r_0)^2}{2\sigma^2}}}_{(2)} \overset{!}{=} 1 \tag{4.15}$$

as there is no φ dependency. To calculate (2) of (4.13) the substitution $x = r - R_0$ is introduced. It follows:

$$\begin{aligned} &\int_0^\infty (x + R_0) e^{-\frac{x^2}{2\sigma^2}}\, \mathrm{d}x \\ &= \sigma^2 + R_0 \int_0^\infty e^{-\frac{x^2}{2\sigma^2}}\, \mathrm{d}x \\ &= \sigma^2 + \frac{R_0}{2} \int_{-\infty}^\infty e^{-\frac{x^2}{2\sigma^2}}\, \mathrm{d}x \\ &= \sigma^2 + \frac{R_0 \cdot \sqrt{\pi\sigma^2}}{\sqrt{2}} \end{aligned} \tag{4.16}$$

Putting this result back in (4.15) allows to calculate a.

$$\frac{R_0{}^2 \cdot a}{2\sigma^2} + \frac{a}{\sigma^2} \cdot \left(\sigma^2 + \frac{R_0 \cdot \sqrt{\pi\sigma^2}}{\sqrt{2}}\right) \overset{!}{=} 1 \implies a = \left(1 + \frac{R_0\sqrt{\pi}}{\sqrt{2}\sigma} + \frac{R_0{}^2}{2\sigma^2}\right)^{-1} \tag{4.17}$$

This defines the function (4.13).

4.2. Implementation in `MATLAB`

4.2.1. Intensity of one Mirror Segment

The simulation has been implemented in `MATLAB`. The core of the simulation is the method `luxOnCCD`. This method is invoked to calculate the intensity of the illumination on the camera by one mirror segment. The method needs the following input parameters:

- the apparent magnitude of the observed object

- radius of the mirror segment (circular mirrors assumed)

- reflectivity of the mirror segment
  1 for 100% reflection
  0 for 0% reflection

- focal length of the mirror segment [m]

- sigma of the mirror's point spread function

- the tilting of the mirror around the x-axis in the telescope frame [radian], 0 means perfect alignment

- the tilting of the mirror around the y-axis in the telescope frame [radian], 0 means perfect alignment

- radius of the camera lens. If there is an aperture with $r < r_{\text{lens}}$ the radius r of the aperture has to be inserted [m]

- the focal length of the lens [m]

- transmission factor of the lens

- tilting angle of the lens

- x, y and z-position of the lens in the mirror coordinate system [m]

- switch to draw a picture of the situation i.e. the light distribution and the lens position (used for debug purposes)

In the developing process this function was implemented first. This is the reason why the coordinates of the lens and the relative tilting angles have to be given in the mirror segment coordinate system. This has to be taken into account for the simulation of an entire telescope.

4.2.2. Simulation of the Entire Telescope

Including a whole telescope in the simulation allows the effects of all mirror segments to be estimated while tuning parameters like lens-position or apertures. The positions for the mirror segments are set in the simulation according to Ref. [29] that defines the positions and orientations of the segments for the MST:

- the mirror segments are placed in the x-y-plane with a distance of 20 mm in between

31

- the *z*-position is chosen in such a way that the mirrors are placed on the surface of a sphere with a radius of 19.2 m

- the orientation of the segments is such that they face a point in the telescope's focal plane in a distance of 32 m

For the implementation in the simulation the class `mirseg`[2] has been defined. This class includes the position of the mirror segment in the *x*-*y*-plane and the tilting relative to the ideal position. With this class a vector including all the mirror segments has been built. With the *x*- and the *y*-coordinate in the dish system the *z*-position has been calculated with the condition, mentioned above, of the placement on the 19.2 m sphere.

The calculation of the illumination intensities for each facet is done by looping through all the elements in the vector. Starting from the position of the facet in the telescope dish, the position of the lens is calculated for the coordinate system of the corresponding facet and then proceeded like in the section above. For analyzing changes in the illumination it is not only looped through all mirror segments, but there is also calculated the illumination for changing parameters, for example moving the lens. For each situation the application returns a vector of illumination that corresponds to the facets as shown in Fig. 4.4.
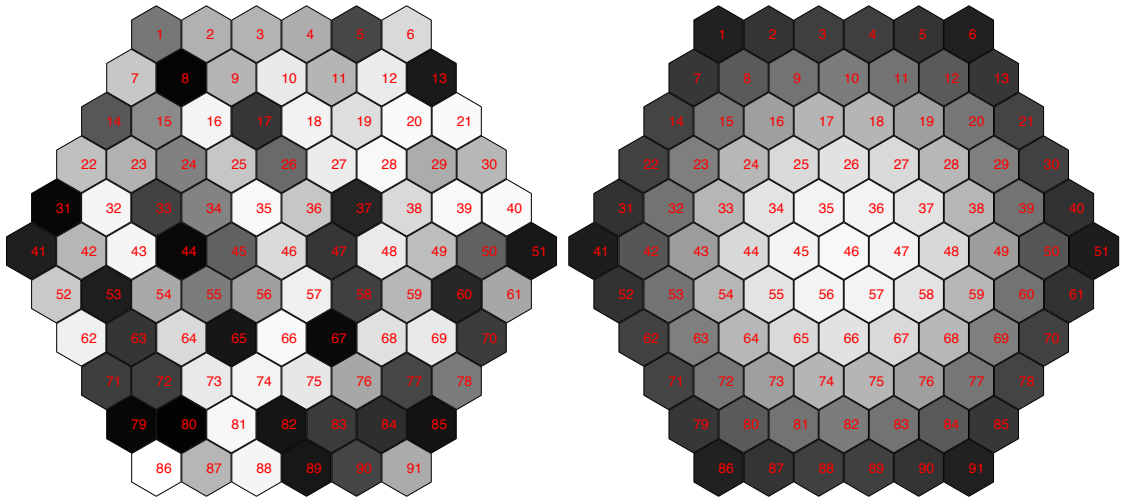

Figure 4.4.: Simulation of the image received by the camera. Left: illumination by a random alignment of the mirrors. Right: ideal aligned mirrors, lens displaced along the optical axis

---

[2]see sourcefile `mirseg.m` in the appendix

Coordinate Systems Used in the Simulation

The simulation uses two different coordinate systems. The first one is a global system along the telescope. It is used to define the positions of the mirror facets and the camera lens with respect to the whole telescope. The definition of the axis is shown in Fig. 4.5. In the x-y-plane the origin of the coordinate system is in the center of the telescope dish.
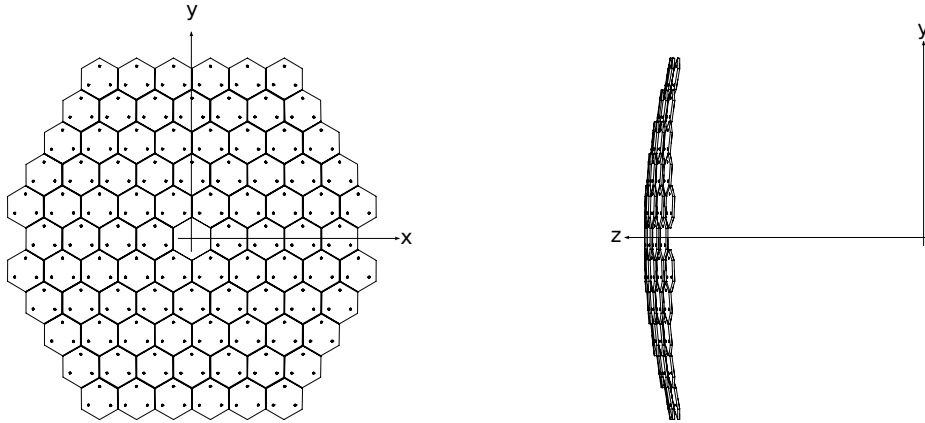

Figure 4.5.: Definition of the telescope coordinate system in the example of the MST prototype, the left picture shows a front view, the right a side view. Adapted from STEP files of [29]

The origin in the y-z-plane is the center of the 19.2 m sphere.

For the individual appearance of each facet, a second coordinate system is used: This coordinate system has the origin of the x-y-plane in the center of the mirror-facet. The origin of the z-axis is defined in the focal point of the mirror facet. The simulation converts the lens coordinate (that is given in the telescope coordinates) to the facet coordinate-system for each facet.

4.3. Simulation Scenarios

4.3.1. Test Setup

As it will be explained in detail in Sec. 5, a test setup has been built to develop automatic aligning algorithms. The simulation of the sensor illumination of the test setup camera helps to choose a bright enough source to target at to obtain a sufficient sensor illumination. Fig. 4.6 shows the illumination depending on the apparent magnitude of the source. The minimal detectable illumination is estimated to be three times over the standard deviation of the background illumination. This means that it depends on the shutter width of the camera and the noise coming from the night sky background. The latter depends again on the location of the setup and the time of day.

In the example in Fig. 4.6 a shutter width of 70 ms is assumed. The standard deviation of the night sky background has been recorded by pointing the setup to the sky. In this configuration the illumination threshold is 0.22 lx. This corresponds to a source with apparent magnitude of about 0.04. Additional information on the camera calibration and comparison with the simulation is given in Sec. 6.3.
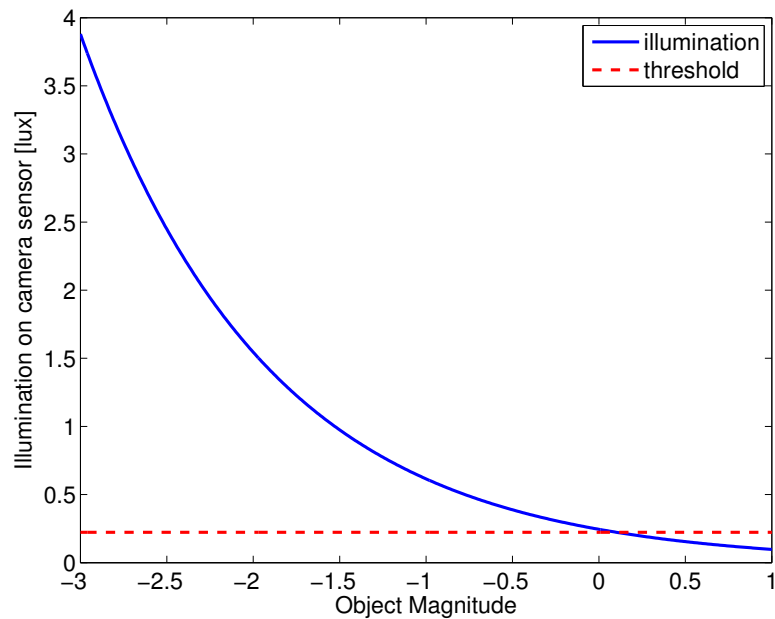

Figure 4.6.: Simulated illumination on the camera sensor vs. magnitude of observed object. The dashed red line shows the threshold intensity for the chip used in the camera of the test set-up with 70 ms shutter width and the Zurich City night sky background.

4.3.2. Moving the Lens along the z-axis

With the goal to realize an automatic alignment system it is interesting how the intensity changes on the sensor chip when the lens of the camera is moved out of the focal point. In a first step the behavior is analyzed when the lens is moved along the optical axis. Fig. 4.7 shows the results comparing an ideal and a real mirror segment. The smooth change of the illumination is a product of the smeared light spot produced by an imperfect mirror.
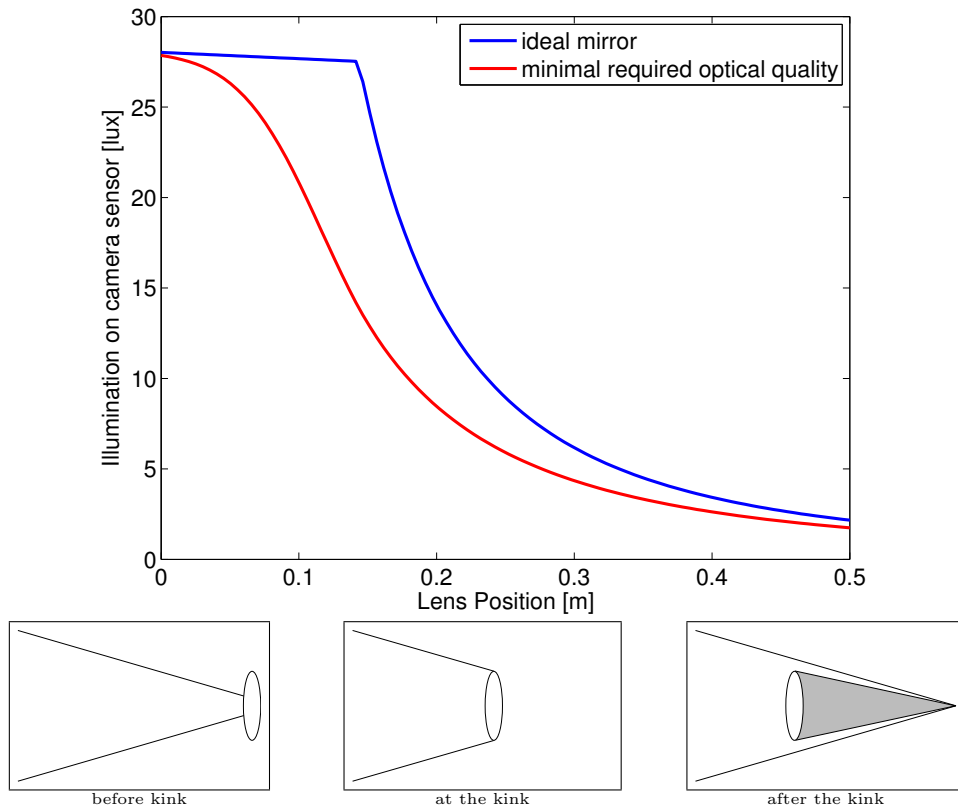

Figure 4.7.: Sensor illumination vs. position of the camera-lens. The kink at about 0.14 m in the plot shows the position where the lens will not capture all the light coming from the mirror anymore. Before the kink, the illumination is changing slightly because of the changing projection size of the mirror on the CMOS sensor.

As long as the lens can capture all the light, the *luminous flux* does not change, but because the lens is approaching the mirror, the imaging of the mirror itself on the camera sensor changes the size and therefore the illuminance.

4.3.3. Tilting the Mirror

The most important simulation is the characterization of the illumination depending on the deviation of the ideal orientation of the mirror. This simulation assumes the ideal position of the lens (Fig. 4.8).
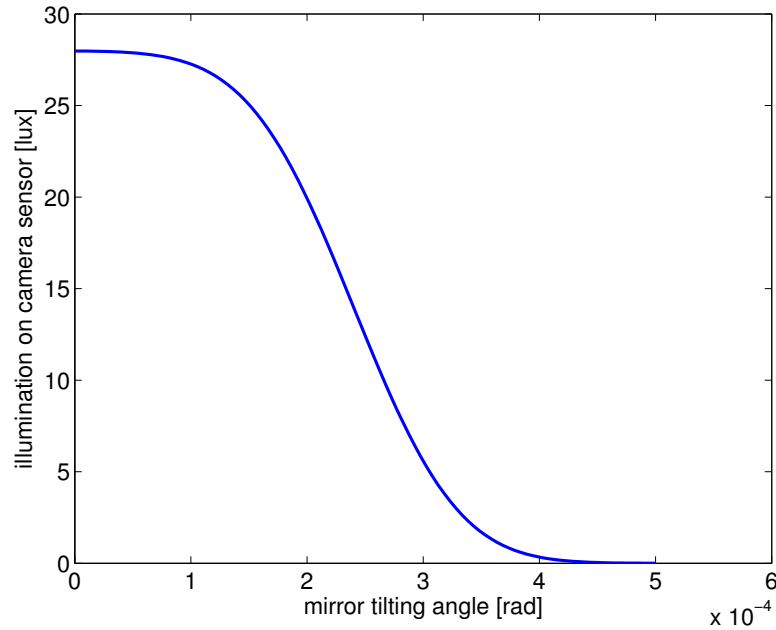

Figure 4.8.: Sensor illumination vs. tilt angle of the mirror. The mirror is tilted along one axis.

The simulation shows that the key sensitivity is in the orientation of the mirrors. As soon as there is an angle between the ideal position and the orientation, the outgoing rays are deviated twice that angle. The larger the distance between the camera and the dish, the faster the illumination decreases while tilting the mirror. This fast drop of the illumination must be taken into account, when setting the step sizes of the AMC of the mirrors. This behavior can be changed with the aperture size of the camera's lens. As soon as the light collection area of the lens gets bigger, the system accept wider displacements of the mirrors implying a worser alignment accuracy.

4.3.4. Aperture Effect of the non-axis Mirrors

The light of the mirrors that are not placed on the dish optical axis will not hit the lens of the camera perpendicular, as the mirror- and the lens-plane have an non-zero angle in between. This results in an elliptical aperture effect for the light passing the lens from those mirrors. Fig. 4.9 shows the consequences of the aperture effect on the light intensities on the sensor. From this plot one can see that in the ideal alignment not all
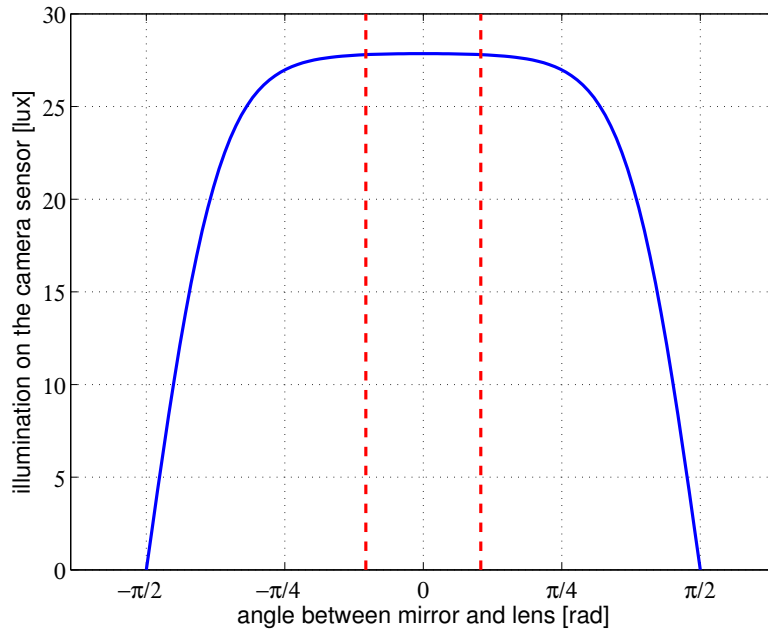

Figure 4.9.: Sensor illumination vs. angle between mirror and lens. The red dashed lines indicate the borders for the MST dish ($\pm 18.68°$) wherein the illumination changes about 0.2%.

the mirrors produce the same illumination on the sensor. Nevertheless within the angles given for the MST dish the change is only about 0.2%.

4.3.5. Telescope Scenarios

According to the implementation explained in Sec. 4.2.2, different scenarios in the whole telescope have been simulated.

Tilting of the Mirrors

The plots in Fig. 4.10 show the illumination change while tilting the mirrors. All mirror segments are tilted simultaneously. The numbering is according to Fig. 4.4.
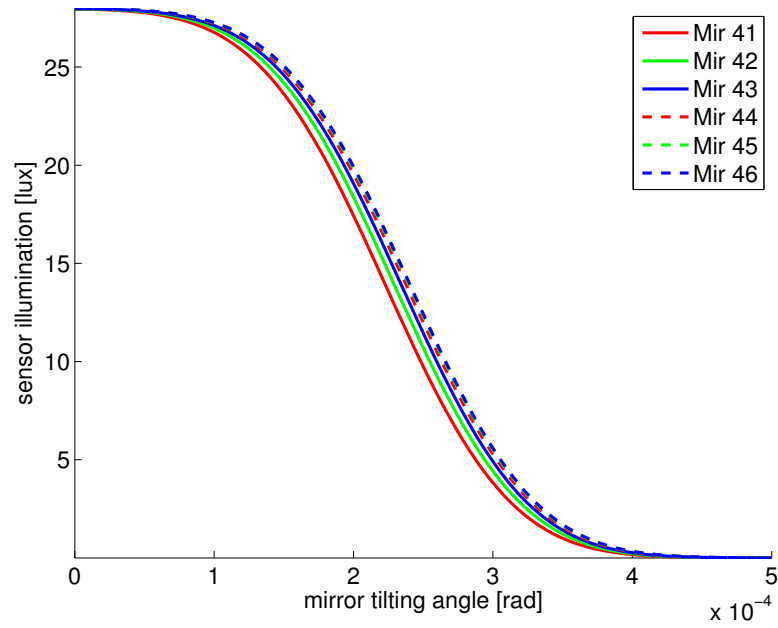

Figure 4.10.: Camera sensor illumination change depending on the tilt angle of the mirror segments. For symmetry reasons not all the mirror facets have been plotted.

Deviation of the ideal lens position along the optical axis

Fig. 4.11 shows the change of the illumination when the lens is moved along the optical axis (i.e. along the z-axis).
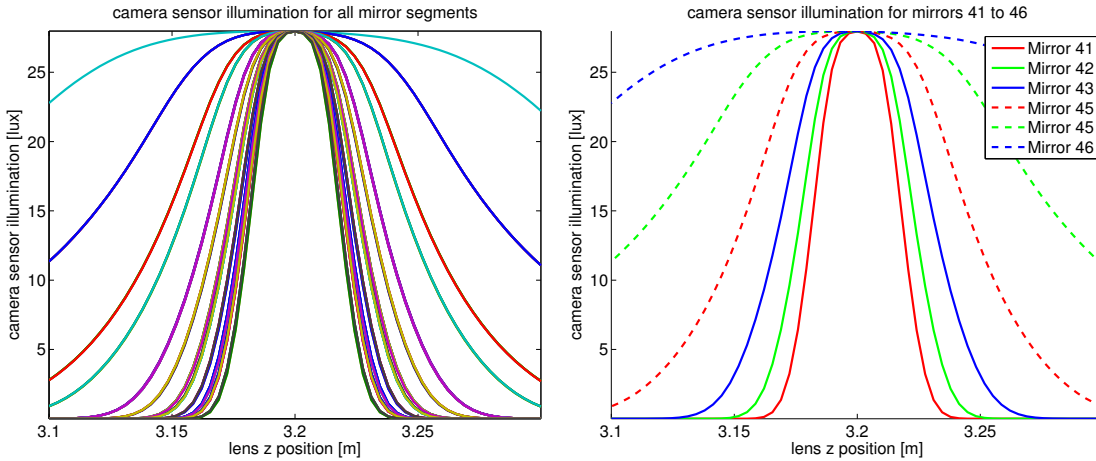

Figure 4.11.: lens shift along the telescope's optical axis. The coordinate of the lens is in the telescope system according to Sec. 4.2.2

It is clearly visible that a displacement of the lens along the optical axis leads to different changes in the illumination that is produced by different mirror facets. If the lens is moved towards the telescope dish, not all the light of the mirror facets will be captured anymore. The light coming from outer mirrors will miss the lens, the light from the central mirror still passes partly the lens (cf. Fig. 4.1 and Fig. 4.12)
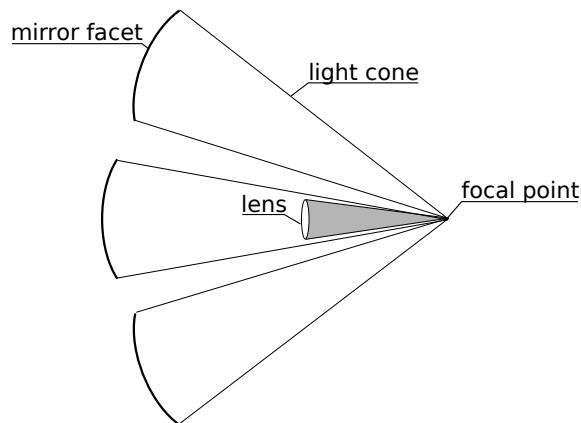

Figure 4.12.: Light captured by the camera lens when the camera is moved towards the telescope

Lateral Deviation of the Lens Position

With the desired optical quality of the mirror facets, the illumination changes rapidly when the lens is moved within the focal plane (i.e. with constant z, Fig. 4.13). This means that the camera has to be placed in the center of the camera. One can consider to place a mirror in the middle of the camera plane if it is not possible to place the camera itself there. Compared to the results in Fig. 4.11 the camera sensor is much more sensitive for displacements of the lens within the focal plane. On the other hand the changes in the illumination does not differ strongly between the different mirrors.
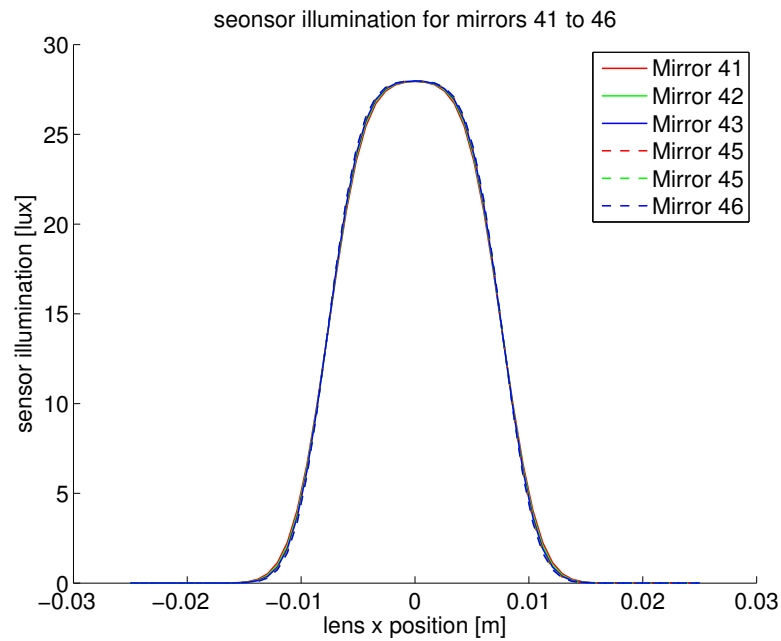
Figure 4.13.: lateral lens shift along the x-coordinate within the focal plane. The illumination changes are the same for all mirror facets.

5. Test Setup

Figure 5.1.: Sketch of the test setup [30]

The Physics Institute's mechanical workshop has constructed a test setup for evaluating different algorithms and camera optics. The setup consists of a digital camera and a parabolic mirror that can be moved with the help of two stepper motors.

5.1. Hardware

The test stand is divided in three main parts:

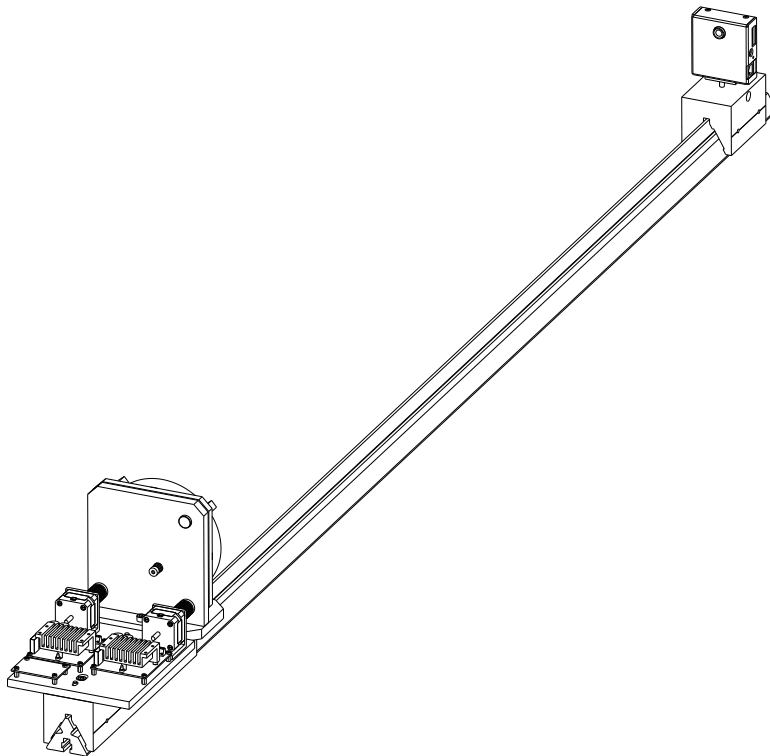|	
optical rail	The optical rail holds the other parts that can be moved for correct alignment.
mirror	The second part consists of a parabolic mirror mounted on a plate that can be tilted.
camera	In the focal plane of the mirror is the CMOS camera mounted.

Figure 5.2.: top view of the setup

5.1.1. Camera with CMOS Sensor

The test stand uses a camera produced by Supercomputing Systems AG, the *leanXcam*. This camera includes a small computer running μClinux[1], a Linux distribution optimized for the usage on embedded systems. As the camera is using a Linux operating system, it can be accessed using common protocols like Telnet or Secure Shell (ssh). The camera can record pictures and is also capable to analyze data on-line using a `C` or `C++` application.

The holder supports the camera with a turnable rod. Therefore the camera can be orientated to different directions. The lens used on the camera is Lensagon BHR16012. It has a focal length $f = 16$ mm. The aperture of this lens has a diameter of 16 mm. A schematic of the lens with more details can bee seen in [31]. This type of lens provides a field of view to have the possibility to use it on a real telescope like the MST of the CTA experiment. The camera uses a 752×480 pixel CMOS sensor by Micron. This is a monochrome sensor with 6 μm × 6 μm size pixles. The minimal detectable light is less than 0.1 lx [32]. There are different I/O ports on the camera. To transfer data between the camera and another computer it is equipped with an ethernet interface. Two optical isolated inputs and outputs can be used to transmit data or control other devices. The motor drivers are controlled using the two optical I/Os (see also next section).

The usage of an intelligent camera has several advantages. Because data analysis and controlling of the motors can be done in the camera, there is no additional computer needed. In addition, because the camera has an ethernet port, it makes easy to transfer data (images, measuring data, move commands for remote motors, etc) over longer distances. For the test setup in this work, all data analysis has been done in the camera.

---

[1] `http://www.uclinux.org`

A PC is only needed to start and stop different test scenarios and to read out the results.

Figure 5.3.: leanXcam case with the camera pcb [33].

Accessing and Controlling of the Camera

The operating system on the camera offers per default different possibilities to connect and control the camera[2]. All of them use the `TCP/IP` protocol using the ethernet port on the camera. Mainly the camera is controlled by using a secure shell connection providing a terminal. For a detailed description of the camera features the reader is referred to the App. A.1.

5.1.2. Mirror and Mirrormounting

The mirror facet that is used is a parabolic mirror with a diameter $d = 10$ cm and a focal length of $f_{\mathrm{mir}} = 1.5$ m. To change the mirror orientation, the mirror is mounted on a plate that can be tilted along three axes (Fig. 5.4). The mirrorplate (A) is connected to a fixed plate (B). The two plates are connected through one screw (C) that pulls the mirrorplate back to (B) with a spring. The screws (D), (E) and (F) can be turned to tilt the mirrorplate by pushing the plate away from (B). For our purposes, screw (D) is only turned manually for an initial set-up. The screws (E) and (F) are each connected to a stepper motor to turn them automatically, described in Sec. 5.1.3.

The adjustment screws move along the optical axis when they are turned. To couple these moving screws to the fixed motor shaft of the stepper motors, a metal bellows

---

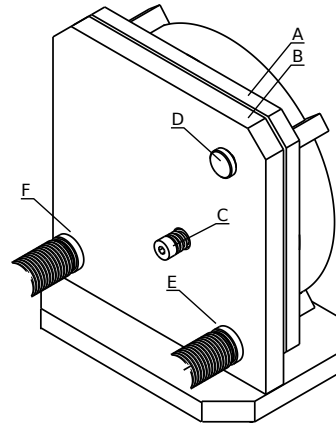[2]A detailed description is given in the App. A.1.1

Figure 5.4.: Sketch of the mirror-mounting: The screws (E) and (F) are coupled to the stepper-motors using metal bellows couplings (adopted from [30]).

coupling is needed to handle the changing distances. These couplings can also handle shafts that are not in a perfect coaxial alignment.

5.1.3. Stepper Motors and -Driver

Stepper Driver

Turning the above mentioned screws is done with two stepper motors. For each screw there is a combination of a stepping motor and a corresponding stepper driver. The two motors are driven with the *Orientalmotor* driver board CSD2109-T. A stepper motor can be connected to these boards and easily driven with only two inputs. One input defines the direction (clockwise or counterclockwise), the other input is used to move the motor one step per input pulse. The stepper driver powers then the coils of the motors in the right cycle to turn the motor in the desired direction.

Stepper Motors

The motors used to turn the alignment screws are two *Orientalmotor* PK243-03B stepper motors. They can resolve 200 steps per rotation achieving a 1.8° resolution in turning the motor shaft. The advantage of stepper motors is the knowledge of the actual position by the application counting the steps and not requiring position sensors. As the motors makes discrete steps, the position in always reproducible. With stepper motors high torque can be applied also while having low rotation speed.
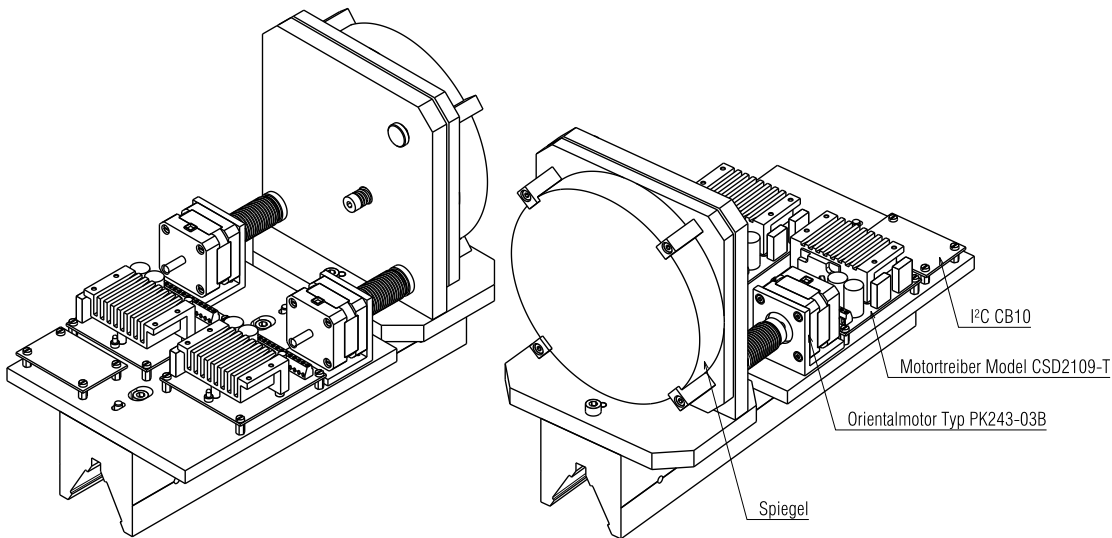
Figure 5.5.: Sketch of the mirror part, consisting of the mirror, the stepper drivers and -motors and the couplings to the mirror [30].

5.1.4. 8-bit I^{2}C Expander

As mentioned in Sec. 5.1.1 the camera has only two digital outputs but the controlling of two stepper boards needs four I/O channels instead. For this reason an 8-bit I/O expander has been used that can be remote controlled with the I^{2}C bus. This is a device that receives commands via the I^{2}C bus having eight inputs and outputs that can be controlled with this bus. The software on the camera communicates via the I^{2}C bus, which pins on the I/O expander has to be high or low to generate the controlling signals for the two stepper boards, corresponding to the time diagram in Fig. A.1. A block diagram in Fig. A.2 explains the functions in detail.

The I^{2}C Bus

I^{2}C is a serial low-speed (in the order of about 100 kbit) digital bus that allows different peripherals to communicate between each other. It needs only two bidirectional data lines, *SDA* (Serial Data Line) and *SCL* (Serial Clock Line). All devices share these two lines via an open-collector connection. For that reason the lines are connected via a pull-up resistor to 5 V. Fig. A.3 shows the principal connection of two devices on a I^{2}C bus.

The bus has two different types of nodes. A *master* and a *slave*. But these roles are not fixed. Every node can start a transmission being a master. In this work, the camera acts always as a master and the 8-bit expander as a slave. Therefore the communication is always initialized by the camera. For additional information on I^{2}C see [34].

5.1.5. Summarized Data Flow of the Test Setup

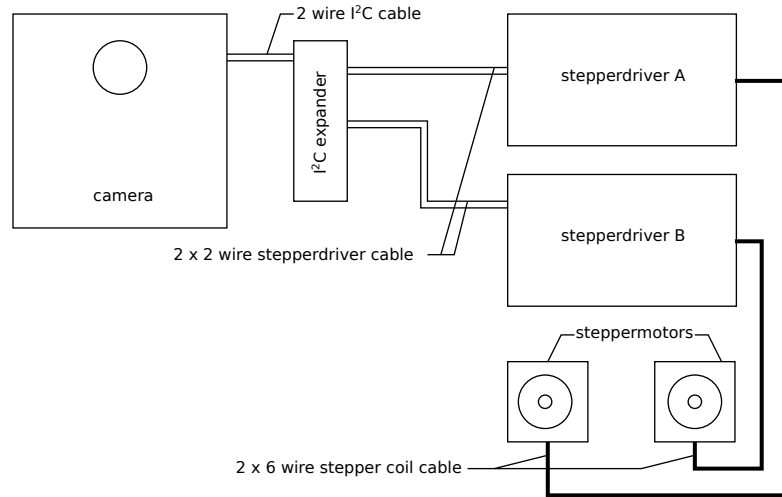The block diagram in Fig. 5.6 shows the communication path between the camera and the motors.

Figure 5.6.: Schematics of the communication path between camera and motors. The camera controls with two wires the I^{2}C expander. The expander itself controls then the two stepper driver that powers the coils of the motors. Not to scale.

5.1.6. Optical Parameters

Tab. 5.1 summarizes the optical parameters of the test stand.

Table 5.1.: Optical parameters of the test stand

mirror	diameter	10 cm
	shape	parabolic
	tilting resolution axis 1	0.035 mrad per motorstep
	tilting resoultion axis 2	0.049 mrad per motorstep
	focal length	1.5 m
lens	diameter	16 mm
	focal length	16 mm

5.2. Software Development

The developing environment is delivered as a virtual machine running a Linux with all the needed software preinstalled. Having a virtual appliance running on a virtualization software that is available for Mac OS X, Windows and Linux makes it easy to start developing software for the camera without walking through operating system specific installations and preparing prerequisites. The virtual appliance provided by the camera manufacturer can be run with the *Oracle VM VirtualBox* application that can be downloaded for free [35] (Fig. 5.7).
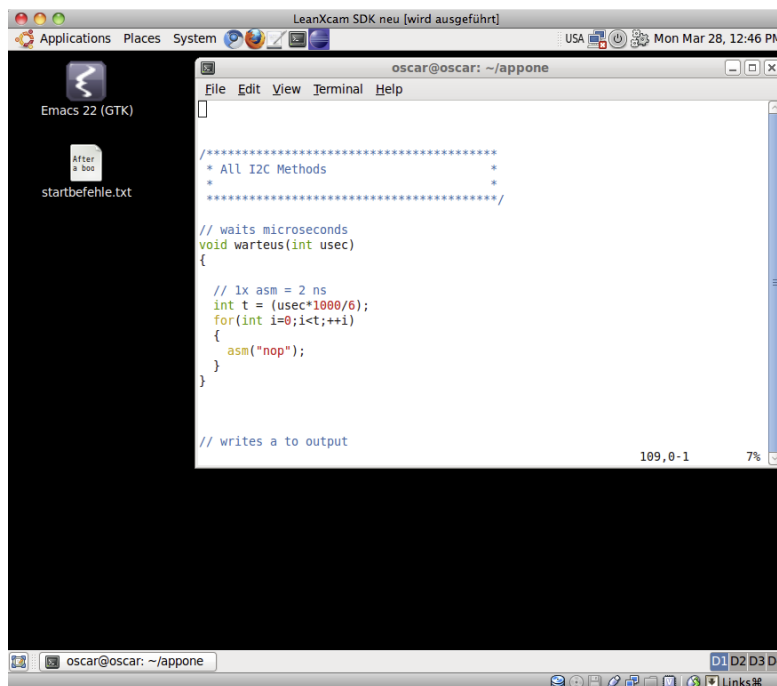

Figure 5.7.: Screenshot of the Ubuntu Linux virtual machine

5.2.1. Camera Framework "Oscar"

The camera provides a framework that allows most of the camera's features to be accessed with prewritten methods, allowing fast access to the hardware of the camera without dealing with low-level hardware programming. The framework is called *Oscar*, a acronym for *O*pen *S*ource *C*amera.

The programming language used in Oscar is `C` and `C++`. All applications developed for this thesis are written in `C`. The framework provides also different ways to compile the application (i.e. `make` targets). It is possible without any changes in the code to com-

pile the application and run it on the development computer itself. This facilitates the process of software debugging because it can be done on the computer directly without copying it to the camera. Also, the reading and processing of images can be emulated with this framework. Instead of recording a real image with the camera it is possible to load a given bitmap picture. This is helpful for the debugging procedure because one knows exactly what kind of picture the camera recorded, which means that a certain situation can be exactly reproduced.

5.2.2. Coordinate System used by the Camera

The applications written for the camera use a coordinate system directly assigned to the hardware. From a zero position that has to be defined, x_m represents the number of steps that motor A has completed in the clockwise direction. Negative values stand for steps in the counter-clockwise direction. The variable y_m is the same for motor B. The motors are not equipped with a absolute position sensor. The application has to count always the steps done and store the current position. When the application is launched the current motor-position has to be entered manually. This could be avoided with the usage of home-switches giving a fixed reference for the application.
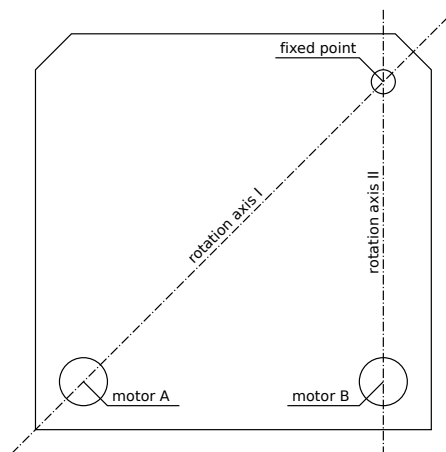

Figure 5.8.: View of the mirror back-plate with the two rotation axes (dashed lines)

A second coordinate system is introduced to visualize the illumination results. The specific positions of the motors and of the fixed point (Fig. 5.8) given by the mirror-mount lead to different angular resolutions for the two motors. Additionally the two rotation axes are not orthogonal to each other. Finally the rotation axes move with the movement of the motors. To understand better what happens to the outgoing rays of the mirror, a rectangular coordinate system in the focal plane is used. The x_r- and y_r-values of this coordinate system specify the position of ideally focused virtual rays coming from the mirror.

5.2.3. I^{2}C Communication

One part of the application deals only with the emulation of the I^{2}C protocol. A specific point has to be mentioned: the I^{2}C-specifications [34] define minimal time lengths in that a SDA or SCL state has to stay before changing the value again. An elegant way to achieve such interrupts in the algorithm are the Standard-`C`-library functions `usleep()` and `nanosleep()`. In these functions it can be defined for how long an application has to sleep, before it continues. The advantage of the usage of such functions is that the time is not depending on the CPU clock speed and the CPU is not busy doing useless commands like `nop` during this time.

Unfortunately this approach is suboptimal for this purpose. These commands guarantee only a minimum waiting time. After the given time, the program is only "ready" to continue. But the kernel decides when the program can continue. When emulating the I^{2}C-protocol, this behavior slows the transmission of commands dramatically down, because a lot of artificial interrupts are needed (see also Fig. A.1). The solution for this problem was to go back to simple algorithms with deterministic execution times. This was done by including the assembler command `nop` into a for-loop. The main disadvantage here is that the processing speed of this kind of algorithm depends on the CPU speed. Therefore this part could not been tested on the computer itself, but only on the camera. Also waiting needs now CPU power. Finally a function is provided that can be used to move the motors to a given position:

```
void moveToPos(int x, int y)
```

where `x` and `y` are the positions of the two stepper motors in steps from a given zero point. This function then calls low-level functions that send the commands to the 8-bit expander that the stepper motors move to the desired position. As the application always stores the current motor-positions in the memory, this function checks again whether the destination coordinates are still within the allowed range. Otherwise the application will quit and print an error message to the console.

5.3. Application Running on the Camera

Several applications have been developed to use on the camera. The most important application is called *Telescope_Tuner*. This application simulates an on-line controlling of the mirrors of a telescope. The other applications (Sec. 5.3.2) are derivatives of the main application. They have been written to do systematical test of the alignment procedure and for calibration analysis of the raw camera sensor data.

All applications are written to run either automatically without the need of any interaction or in an interactive mode which allows different parameters to be entered during runtime. If the application is started without configuration parameters it switches to the interactive mode.
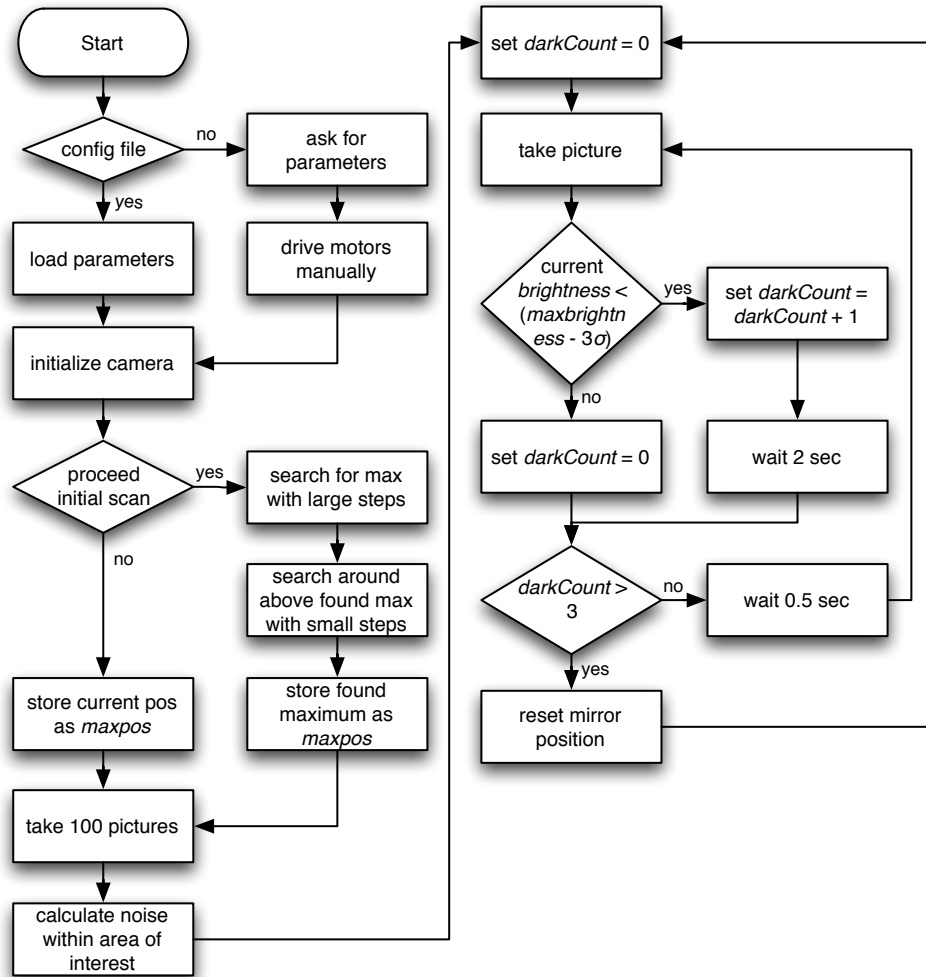
5.3.1. Telescope Tuner

Figure 5.9.: Flowchart of the main application *Telescope_Tuner*.

The idea of this main application is to correct deformations of a telescope by moving the mirrors in such a way, that they will be again aligned correctly. The flowchart in Fig. 5.9 shows the basic application flow. The initial parameters (either given in the config file or entered manually) are the following:

motor positions	the coordinates of the two motors, when the application is started
manual drive	controlling whether a manual initial moving of the motor is done
initial scan	if *yes* the application scans through all possible mirror orientations (within the allowed range) to find the position, where the camera is illuminated the brightest. If the value is *no* the camera assumes the current position as the ideal one and starts to record the statistics.
roi center	center of the region of interest (roi) circle on the ccd.
roi radius	square value of the roi radius.
step width	defines the width (in number of motorsteps) of the large steps
sub step width	defines the width (in number of motorsteps) of the small steps
CMOS exposure time	sets the exposure time of the CMOS sensor in microseconds
motor range	sets the maximum number of steps the motors are allowed to drive.

If the application is set to do an initial scan, it scans a rectangular grid within the motorposition 0 and the given maximum range. The steps between two motor-positions are given in the *step width* setting. The camera searches the position where the mirror is illuminated the brightest. Afterwards the mirror moves to that position and there is another scan proceeded around this point with the *sub step width*. Fig. 5.10 shows the inital scanning procedure. *A* is the position where the highest illumination occurs after the first scan, *B* shows the highest illumination found after the detailed scan.

After finding the position where the illumination is the highest, the camera takes 100 pictures in that position to record statistics on the illumination in the region of interest (i.e. the average intensity $\overline{I}$ and its standard deviation σ_I). Deviations are not only caused by the noise of the camera sensor, also the illumination on the mirror itself can change e.g. due to atmospheric disturbances or a not constant shining source.

After that the program is in its main loop. It takes a picture every 0.5 s and integrates the illumination over the region of interest. This value is compared with the average illumination $\overline{I}$ calculated before. If the measured illumination is below a threshold value (can be specified in the application, default is $\overline{I} - 3\sigma_I$) the application flags this image as having a reduced intensity. But as a lower illumination can be caused not only by a deformation of the telescope, the application does not immediately start to realign the mirror. A lowered value of illumination can also be caused by a temporarily blocked path of the light, e.g. when an aircraft is between the telescope and the pointed light source. To handle such a situation, the application waits two seconds but increments a
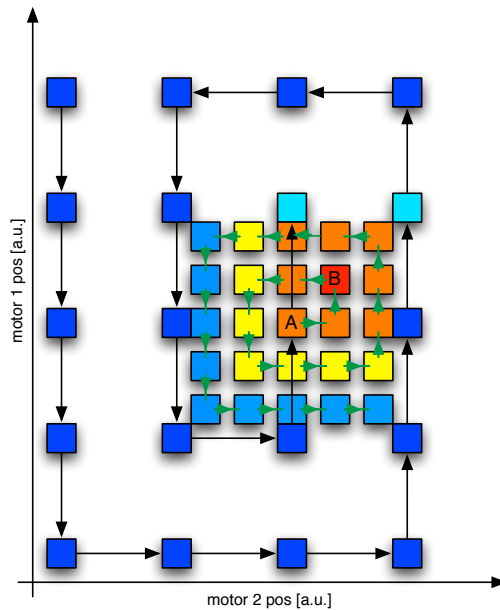
Figure 5.10.: Coordinates of the motorposition for the *initial scan*. Every square symbolizes a motor position. The color indicates the illumination strength on the camera. Blue means low, red high illumination. Black arrows show the first scan with large steps; green arrows show the detailed scan with small steps

variable called `darkCount`. After the two second delay the application measures again the illumination. Only if the illumination is below the threshold more than three times in succession, a realignment of the mirror is started.

The realignment is done by a spiral scanning around the point where the last mirror position was. Assuming that the deformation of the telescope is usually not random but close to the situation before, the application searches for the right position by moving the mirror directly with the *sub step width*. The application does not sweep through all possible mirror positions. As soon as the illumination reaches the threshold, the application stops the search algorithm and goes back into the main loop. If the application cannot find the right position inside of a given radius, it starts again with the initial scan writing an error message into the log file that it was not able to find the new position.

5.3.2. Software Branches

Spot Finding Test   To test the application, two branches from the main application have been written. The first one described here is an application that allows to test systematically the realignment-procedure of the mirror. Fig 5.11 shows a flowchart of this application. After the initial optimal alignment is found, the application displaces the mirror ten times in a certain direction, always the same distance relative to the
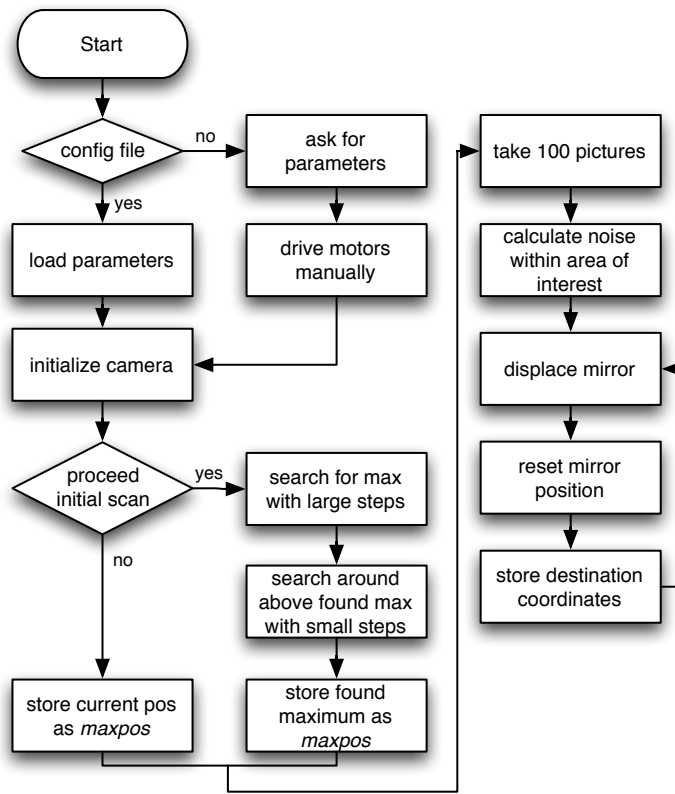
Figure 5.11.: Flowchart for the spot finding test.

initial alignment. After that the application realigns the mirror in the same way as a deformation had be occurred.

Calibration The second branch of the application is used to calibrate the illumination values of the sensor with sources of known luminosity. The application records every second a picture with a programmable shutter width and writes the results to a text file. This data has beed used to compare the illumination results with the simulation (Sec. 6.3).

6. Test Measurements

6.1. Test Stand Initialization

After setting up the test stand, the position of the mirror in the cameras must be specified. For this, a picture of the camera is recorded while the test setup is illuminated. Fig. 6.1 shows typical pictures of the camera. With the help of that picture the position of the mirror can be specified. For the test stand there are two parameters needed: the center of the mirror and the radius.
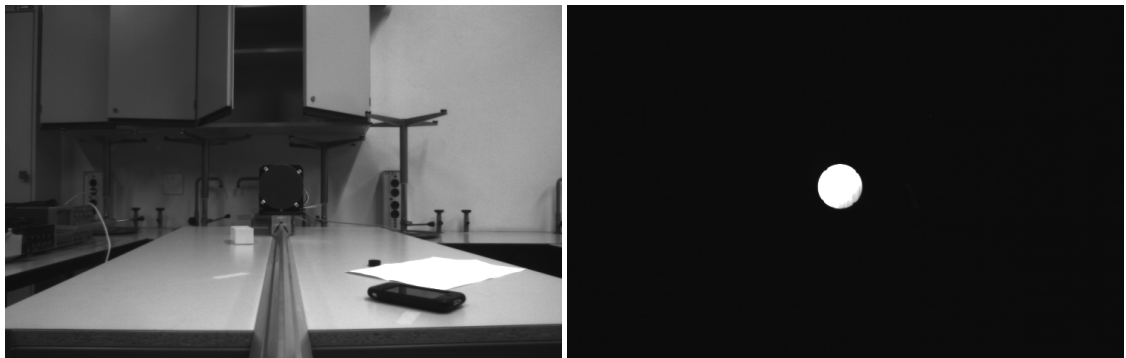

Figure 6.1.: Left: Recorded image of the camera for specifying the mirror position. **Right:** Image of the camera while pointing to a bright source. The dark edge of the bottom right of the mirror indicates that it is not perfectly aligned.

6.2. Spotting the Tower

A point-like light source, far away, is required to test the application. A continuously shining red light on an antenna tower at the Zürich Chäferberg (Fig. 6.2) is used for this purpose. The advantages of this light source are that it is far enough to act as a point-like source. It is bright enough that the focused spot is visible for the human eye and it is not a moving source.

Especially during the development process of the application, it was extremely helpful having a spot, that was visible in the focal plane. This allowed a rough alignment by hand and malfunctions were immediately discovered. Nevertheless the source was distant enough to have a small spot in the focal plane (about 2.5 mm in diameter).
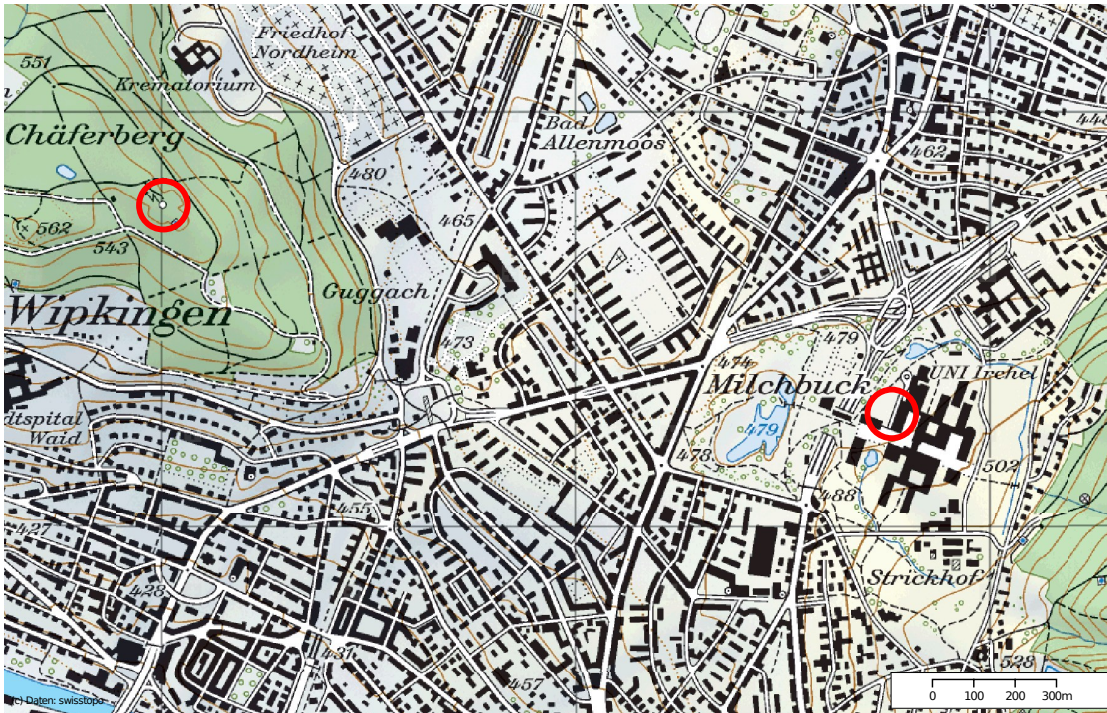
Figure 6.2.: Plan of the test setup [36]. The left circle shows the position of the tower with the red light source. The right circle shows the position of the mirror. The distance is about 1850 m.

For systematic tests of the algorithm the test stand is displaced repeatedly to a new position to simulate a deformation of the telescope. It is not easily done to displace the whole test stand reproducible and if done so it would be time consuming. For this reason, the application mentioned in Sec. 5.3.2 is used for displacing the mirror systematically.

For additional analysis (e.g. comparison with the simulation) the application also records a "heatmap". This means that the application records the illumination for all mirror positions within the allowed range with the sub step width resolution. This information is exported to a text file and is not stored in the memory. Therefore this information is only available offline for the analysis of the algorithm's behavior and is not part of the finding procedure itself.

6.2.1. Spot Finding

For every spot finding test the application displaces the mirror ten times to the positions shown in Tab. 6.1 (relative to the maximum coordinates) and starts a realignment procedure (as described in Sec. 5.3.2)

Table 6.1.: Displacement of the mirror in motor coordinates relative to the ideal position. The mirror is displaced in all indicated direction ten times.

motor 1	motor 2	indicator
-15	-15	↙
15	15	↗
-15	0	←
15	0	→
0	-15	↓
0	15	↑

Fig. 6.3 shows the found positions for different test runs. The third row shows some tests where the optimum position found by the algorithm lies outside the region of the lens. This behavior is caused by a bug in the algorithm, while trying to add a new feature: To concentrate the hits more in the center of the lens, the application made an additional scan. After having found the new position of the spot, it started to do a horizontal and vertical cross scan to locate the center of the lens. But this approach has been dismissed. It makes no sense to have a precision smaller than the pixel size (the used camera lens has a smaller size than a common photomultiplier tube) and the PSF of the mirrors will produce a spot in the focal plane that is bigger than the lens itself. This would produce a heat map as seen in the simulation before as there is no region with a (theoretically) equal illumination. Since the application is directly running on the camera the algorithms have been designed for low memory and CPU usage. The values for different positions are not stored in the memory. There are always only comparisons between the last measured value and another one (e.g. the last peak value). If a camera-system with more resources was used more sophisticated algorithms could be implemented.
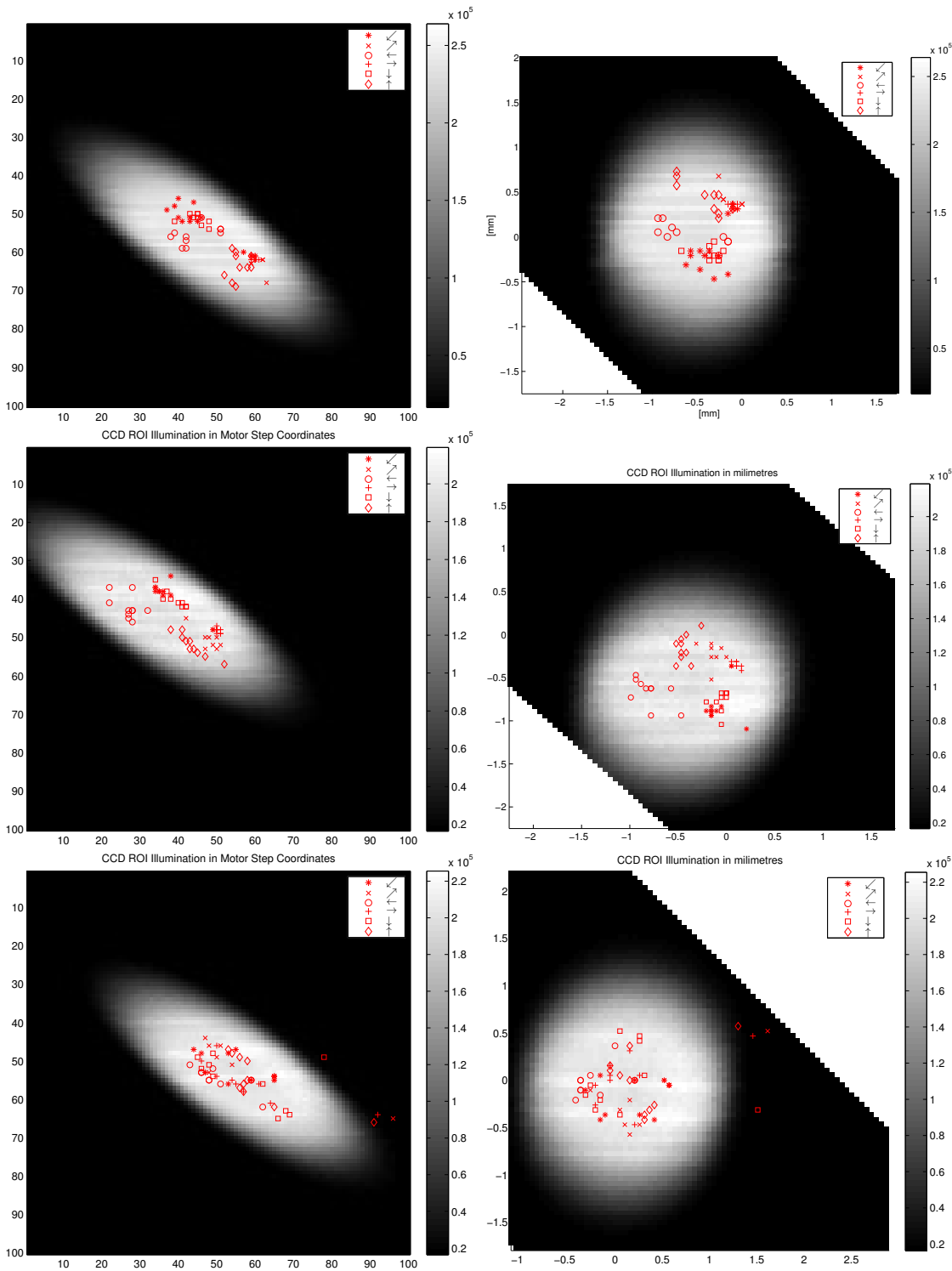
Figure 6.3.: Results of the spot finding algorithm tests. Each row shows the results of one turn. The heat map shows the intensities for all motor positions, the red markers the destination of the realignment. The left column shows the results in the motor coordinate system, the right column uses the millimeter system.

6.3. Calibration Measurement

For the comparison between simulation and measurement, as well as for estimation of the sensitivity of the camera a calibration analysis has been done. The test setup is first pointed to the night sky without pointing to a specific star. The recorded illumination of the camera sensor within the area of interest (i.e. the imaged mirror) gives a value of the noise generated by the sensor and the night sky. The application returns the raw
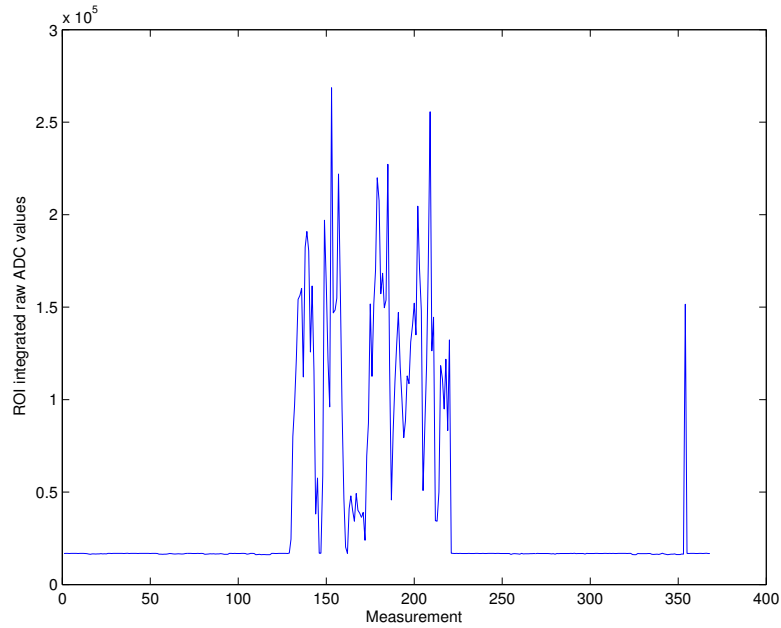

Figure 6.4.: Sum of the raw ADC values within the region of interest for different measurements. The illumination changes depending on the mirror alignment. Measurement number 0 to 130 are taken for the background noise estimation.

ADC values over the region of interest (i.e. the pixels of the sensor where the mirror is), Fig. 6.4. This value already contains a subtraction of the dark ADC value of camera-sensor. There is a blinded area on the sensor that is used for averaging the dark values. For every measurement the average ADC value has been exported. With this data the mean and the standard deviation of the average ADC value have been calculated. The mean background ADC value per pixel is $B_{\mathrm{ADC}} = 13.29 \pm 0.01$. The standard deviation is $\sigma_{\mathrm{ADC}} = 0.13$. The minimal ADC threshold T_{ADC} is then:

$$T_{\mathrm{ADC}} \equiv B_{\mathrm{ADC}} + 3 \cdot \sigma_{\mathrm{ADC}} \tag{6.1}$$

The responsivity of a pixel is [37]:

$$R_{\mathrm{pix}} = 4.8 \frac{\mathrm{V}}{\mathrm{lx} \cdot \mathrm{s}} \tag{6.2}$$

The measured value is given as

$$U_{\mathrm{measured}} = \frac{U_{\mathrm{FS}} \cdot \text{ADC-Value}}{2^R}. \qquad (6.3)$$

where $U_{\mathrm{FS}} = 1.4$ V, the full range voltage of the ADC, and $R = 8$, the vertical resolution.

The reflected light of Sirius in the center of the lens leads to a mean pixel voltage of $U_{\mathrm{mean}} = 1.10$ V. With an exposuretime of 70 μs this corresponds to a sensor illumination of (3.262 ± 0.002) lx. Compared to a simulated value of 0.94 lx it is about 3 times higher than the simulation. The reason for this could be that the sensitivity specified in [37] is a minimal value.

The threshold illumination is then 0.0064 lx, whereby this value is only valid for the given night sky background (NSB) at the time of measurement.

The raw data that has been read out includes noise of the NSB and of the sensor itself. To estimate the two contributions, a measurement of the ADC data is done with the whole camera covered. This measurement gives the contribution of the sensor. Tab. 6.2 summarizes the result.

Table 6.2.: Summarized Noise Values

Noise Source	ADC p. pix.		Lux	
	mean	σ	mean	σ
Total	13.27	0.13	0.216	0.002
Sensor	0.00	0.02	$0 \cdot 10^{-4}$	$4 \cdot 10^{-4}$
NSB	13.27	0.12	0.216	0.002

Tab. 6.2 shows that in the test measurement the noise is dominated by the NSB and not by the sensor itself. The result emphasizes that the minimal brightness of the light source, as specified in Sec. 4.3.1, is needed because of the noise generated by the NSB and not of the chip itself. Therefore a better sensitivity at sites with lower NSB can be expected. The minimal illumination needed respecting only the sensor noise is 0.0012 lx.

7. Discussion

7.1. General Aspects

The tests done with the setup show that it is possible to realize an automatic focusing with the usage of a CCD/CMOS sensor. To distinguish between the "small" camera for the alignment and the "big" camera with the photomultiplier tubes we use the term *sensor* for the small camera indicating the usage of an CCD or CMOS sensor and *camera* for the PMT camera. The system proved to produce reproducible results within the accuracy provided by the optical components. In this case the accuracy was dominated by the aperture size of the lens. Nevertheless some aspects, like camera oscillation, sensor position and source distance, need to be discussed in the following paragraphs.

7.2. Camera Oscillation

The limited stiffness of the telescopes leads not only to a deformation when the orientation of the telescope is changed but also while the telescope stays at a position. The alignment algorithm must take into account the fact that the camera, on its long lever arm, undergoes oscillations that occur at a faster rate than the speed of the alignment search. The most reasonable way is to average recorded pictures of the sensor at least over the period time of the natural oscillation modes of the telescope structure. It is clear that this leads to a worsening of the alignment-precision. However, these oscillations also occur during an actual measurement, and therefore it is not possible to align the mirrors with a precision better than that given by the amplitude of the camera's oscillations.

The main problem is that averaging is time consuming, making the whole algorithm slower. This especially affects the scanning algorithms where a lot of different pictures are recorded. It has to be known how long the oscillation period is (typically lower than 0.5 s) to configure a reasonable averaging time of the pictures.

7.3. Position of the Sensor

For a direct matching of the ideal mirror position and a positive result on the sensor, the sensor must be placed in the center of the focal plane. This location might be already occupied by photomultiplier tubes. If it is not possible to place the sensor or a mirror

in the center of the camera, there are two other ways to avoid this problem: The sensor can be placed somewhere else, like on the edge of the main-camera. In this case the sensor does not interfere with the photomultipliers. But in this case the mirror facets are not orientated in the optimal way. However, even though their orientations are not optimal, it is known. Therefore, the true optimum orientations can be calculated.

The second possibility is to place the sensor outside the detector-lid. In this case an alignment can only be done as long as the detector-lid is closed. A on-line realignment would not be possible then.

7.4. Focusing to the Right Distance

When the alignment is used while pointing to a distant source like a star it has to be considered that the system focuses to a source of infinite distance. The Cherenkov light instead is produced in an altitude of order 10 km. This problem also occurs with the usage of an artificial star produced with a laser beam, producing a source at a distance of about 90 km [23]. If it is done with a distant source, it is at least known, where the mirror-facets are pointing then. It can be calculated how far they have to be moved for a closer focusing.

7.5. Time Consumption

An important point for the alignment procedure is the time consumption. Depending on the computing hardware used, the calculation of the mirror intensities can be time-consuming. With the low-performance computer on the camera used for this work this could already be noticed while optimizing the algorithm. In a first version of the application the comparison whether a certain pixel of the image belongs to the region of interest or not, was done in a inefficient way. This slowed down the initial scanning procedure.

The fact that the sensor has to scan either through all possible position or around a given point is the major disadvantage of this alignment system. The sensor does not know how far a facet is misaligned. It only knows whether it is misaligned or not. To avoid scanning useless positions or in a too high resolution the step width and sub-step width have to be chosen carefully.

In the main purpose of the application, the realignment of the mirrors, the different duration times are visible, depending how far away the light spot has been displaced. The time needed for the realignment is recorded in the simulation as described in Sec. 6.2.1. The minimal time that the system needed was 3 s, the maximum time was 36 s (Fig. 7.1).

Another study of the time consumption is performed. After an initial alignment, the mirror is displaced randomly within a radius of 30 motor-steps. The time to realign the
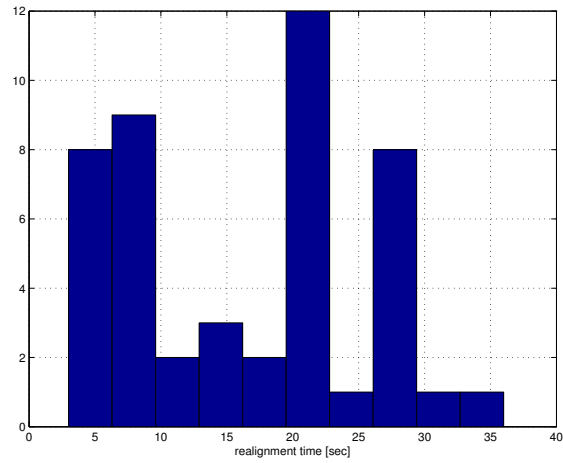
Figure 7.1.: Histogram of the duration time for the realignment of the mirror when it has been displaced like described in Sec. 6.2.1. The x-axis shows the time in seconds.

mirror is then recored. In this case, the differences in time consumption are even larger: the minimal time is 1 s, the maximum time is 144 s.

The difference for these realignment times can be explained as following. In the test setup, the focused spot of the mirror is smaller than the aperture of the lens. This means that as soon as the spot is within the aperture, the system sees a fully illuminated mirror and assumes that the mirror is properly aligned at this position. In the procedure, where the mirror has been displaced ten times to different positions, the situation can be as drawn in Fig. 7.2.

After the initial scan, the application chooses the position indicated in Fig. 7.2. When it displaces the mirror equal in distance once in the positive direction of one motor-coordinate and once in the negative direction, the distance until the application reaches the area with ideal illumination is different. Therefore, a spiral scan around the new point needs different times to realign the mirror.
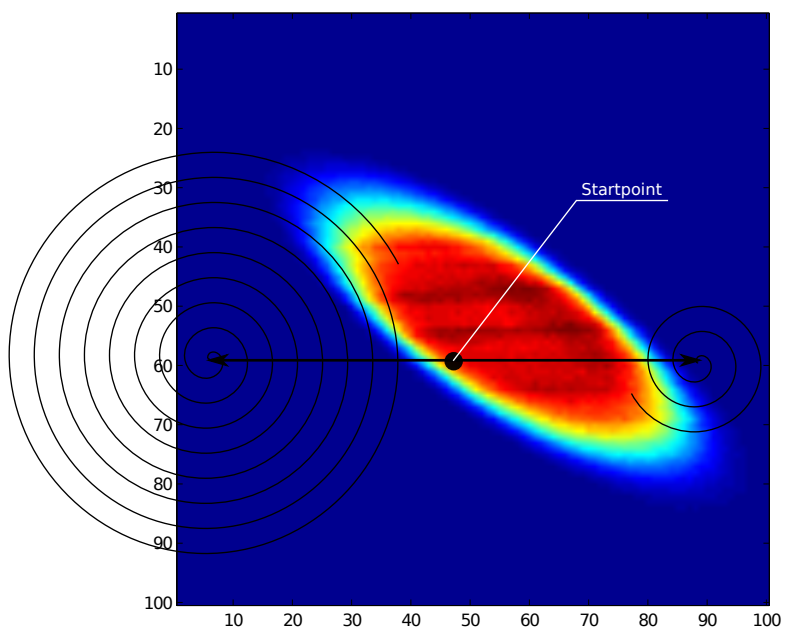
Figure 7.2.: Possible situation for demonstrating the different realignment times.

A. Hardware

A.1. Camera

A.1.1. Communication

ssh The camera can be controlled via a ssh connection. Since the µLinux runs on single user mode, the only user than can be used is the *root*-user. The ssh interface provides a terminal that can be used to configure the camera settings and start applications running on it.

telnet Telnet provides the same functionalities like ssh. The only difference is, that the communication is here going through an unencrypted channel. While using telnet, the camera is not asking for the root user password.

web server A web server running on the camera can be used to interact with an application running on the camera and to receive data via a common web browser.

A.1.2. Camera Startup Configuration

During a proper start up, the camera reads different configuration variables from a memory and sets the corresponding environment values. This variables are called *U-boot environment variables*. The current values can be read with the `fw_printenv` command. To set a variable the command `fw_setenv` *variable value* is used. The most important ones are listed in Sec. A.1.3.

Since the `gatewayip`-setting that sets the default gateway address is not working correctly, the camera has no route entry for the communication to destinations in another subnet. If proper network connection is desired, this setting has to be entered with the following command:

```
$ route add default gw ip-address
```

where the string `ip-address` is exchanged with the address of the gateway. All the startup-commands that need a proper network connection (e.g. synchronizing the internal clock) must be executed manually after the gateway setting.

The camera does not run any kind domain name services. Therefore, the camera cannot handle host names. All outgoing communications must be initiated with the use of IP addresses.

A.1.3. U-boot environment variables

runscript defines the application or script that will be launched on camera startup. The command is used as follows:

```
$ fw_setenv runscript testapp/run.sh
```

ipaddr defines the ip-address of the camera, e.g.

```
$ fw_setenv ipaddr 192.168.1.10
```

netmask defines the network mask, e.g.

```
$ fw_setenv netmask 255.255.255.0
```

gatewayip defines the default gateway ip-address. Used to add a route entry on the camera when inter-subnet communication is desired, e.g.

```
$ fw_setenv gatewayip 192.168.1.1
```

On the µLinux used for this work, this command is not working. A bug report to this problem is still unanswered[1].

ntpserver can be used to set an ip-address of a ntp[2] server, e.g.

```
$ fw_setenv ntpserver 130.60.7.42
```

If set, the camera is designed to synchronize its internal time to the current time. This command is also not working, as long as the ntp server is in another subnet, because the `gatewayip` setting is not working. So the camera can't contact such a ntp server during startup.

For additional startup configurations and environment variables consider the camera documentation [33].

---

[1]`http://leanxcam.origo.ethz.ch/node/292`
[2]Network Time Protocol

A.2. Stepper Motor

For electrical isolation, the stepper boards use optocoupler inputs. In the following signal diagrams a high-level (or logical *true*) indicates the optocoupler-led switched on, low-level (or logical *false*) for led switched off. The diagrams in fig. A.1 show the timing and meaning of the controlling signals.
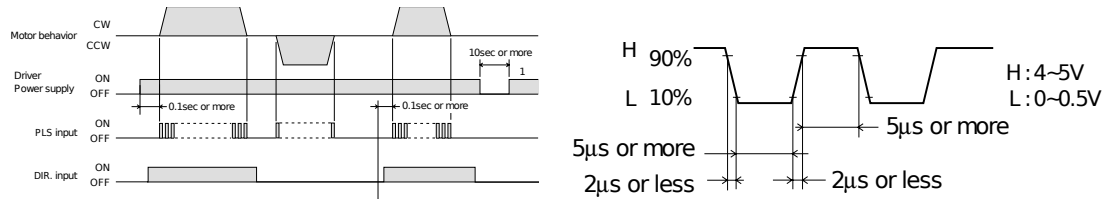

Figure A.1.: timing diagrams for controlling the driver board (adopted from [38])
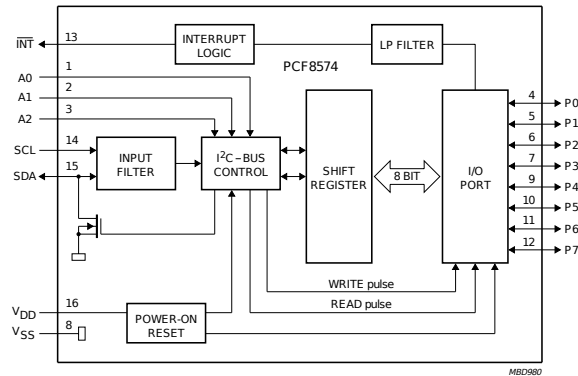
A.3. I²C Communication

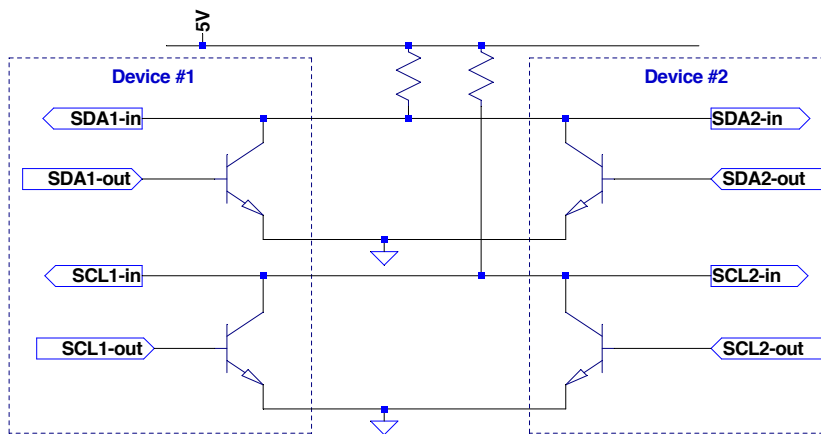**Figure A.2.:** block diagram of the PCF8574 I²C expander [39]

Figure A.3.: Circuit diagram of an I²C bus. It uses two open-collector data lines *SCA* and *SCL*.

A.3.1. PCB Design

To produce an I²C data line between the camera and the I²C expander, two PCBs have been developed (Fig. A.4). The first board is placed at the camera side. The connection to between the board an the camera uses the pins described in Tab. A.1. Between the two boards, there is a four wire cable used as the I²C interface, with the wires described in Tab. A.2.

The second PCB is placed at the mirror side of the test setup. This board receives the I²C commands on the 8-bit expander. The data wires *SDA* and *SCL* are directly routed to that chip. The board then converts the high- and low-level signals of the expander in
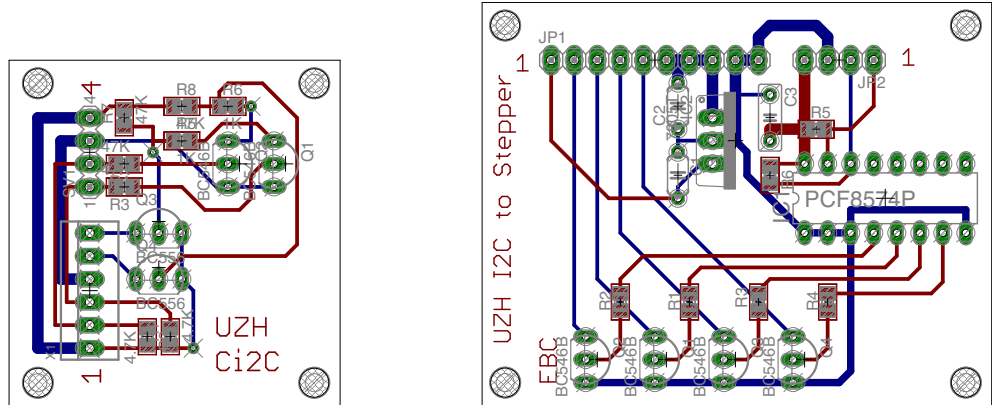
Figure A.4.: PCBs for I²C transmission. The left board is placed at the camera, the right board at the stepper drivers. The red lines are on top, the blue lines on the bottom of the board.

Table A.1.: Pin description of the camera.

Pin #	Type
1	5 V power supply
2	Output 1
3	Output 2
4	I/O Ground
5	Input 1
6	Input 2

such a way that the opto-coupler LEDs of the stepper driver boards are switched on or off. Also, the voltages needed for the driver boards (i.e. 5 V for the logic and 24 V for the motor coils) are provided by that board.

Table A.2.: I²C signal wires used in the cable between the two boards

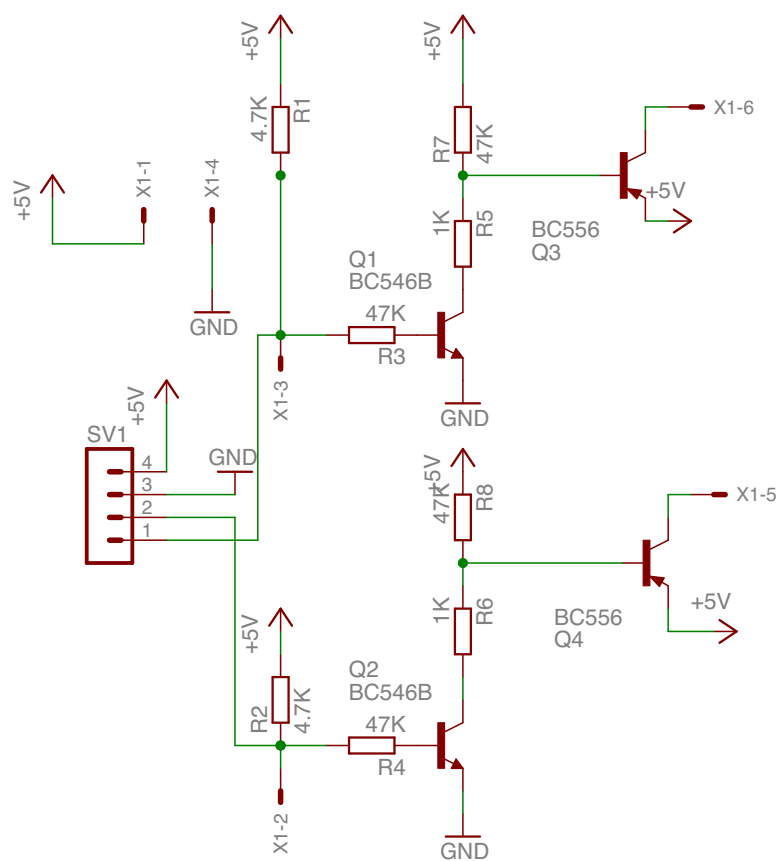| Pin/Wire #	Type
1	SDA
2	SCL
3	GND
4	+5 V

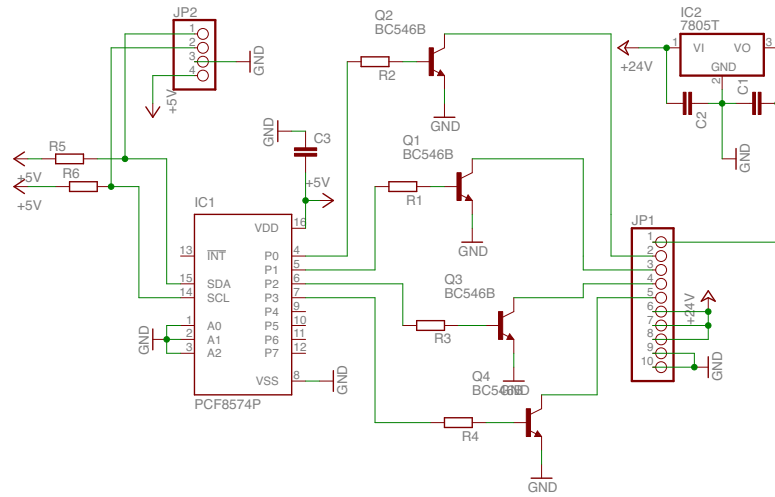**Figure A.5.:** Schematics of the PCB placed at the camera

Figure A.6.: Schematics of the PCB placed at the stepper driver

B. Simulated Telescope Pictures
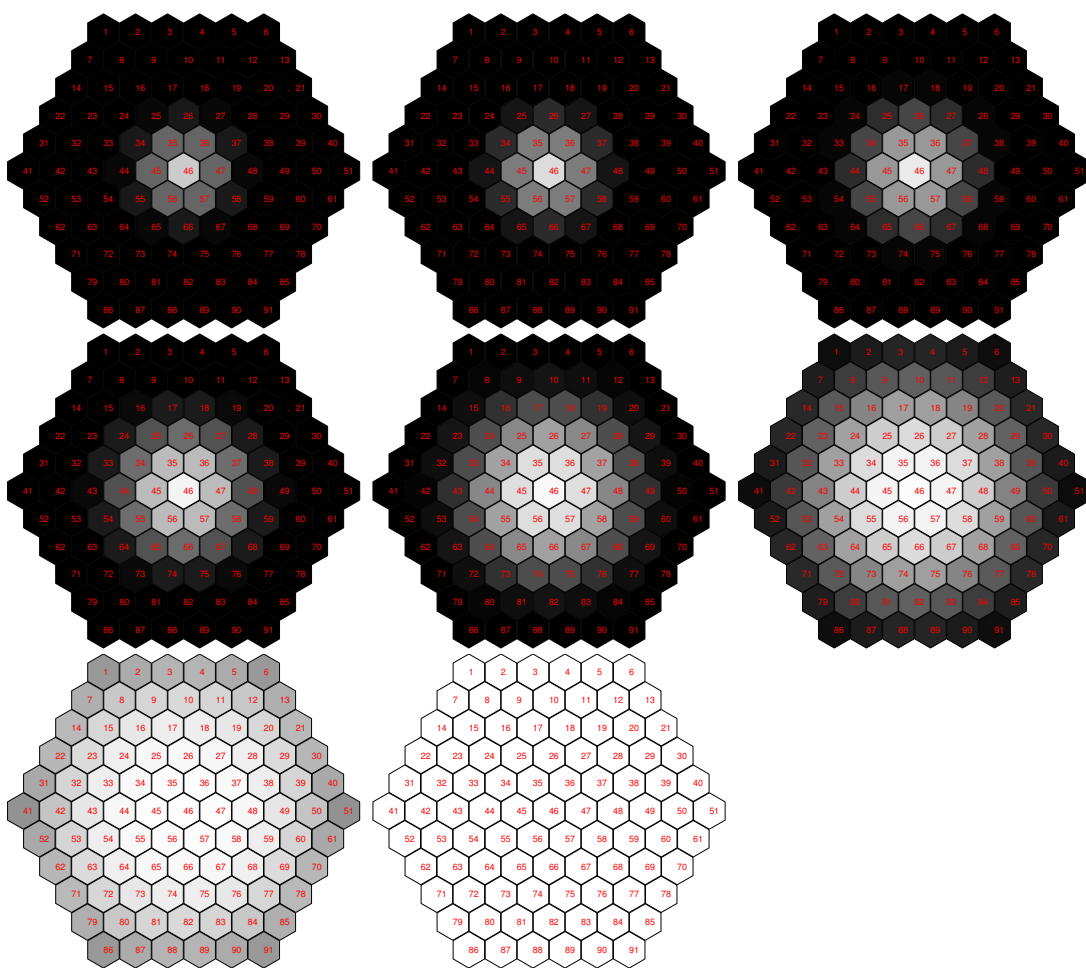
B.1. Moving the Lens along Optical Axis

Figure B.1.: Simulated pictures of the camera. The lens is moved along the telescope's optical axis. The steps in between every picture is 1.4 cm

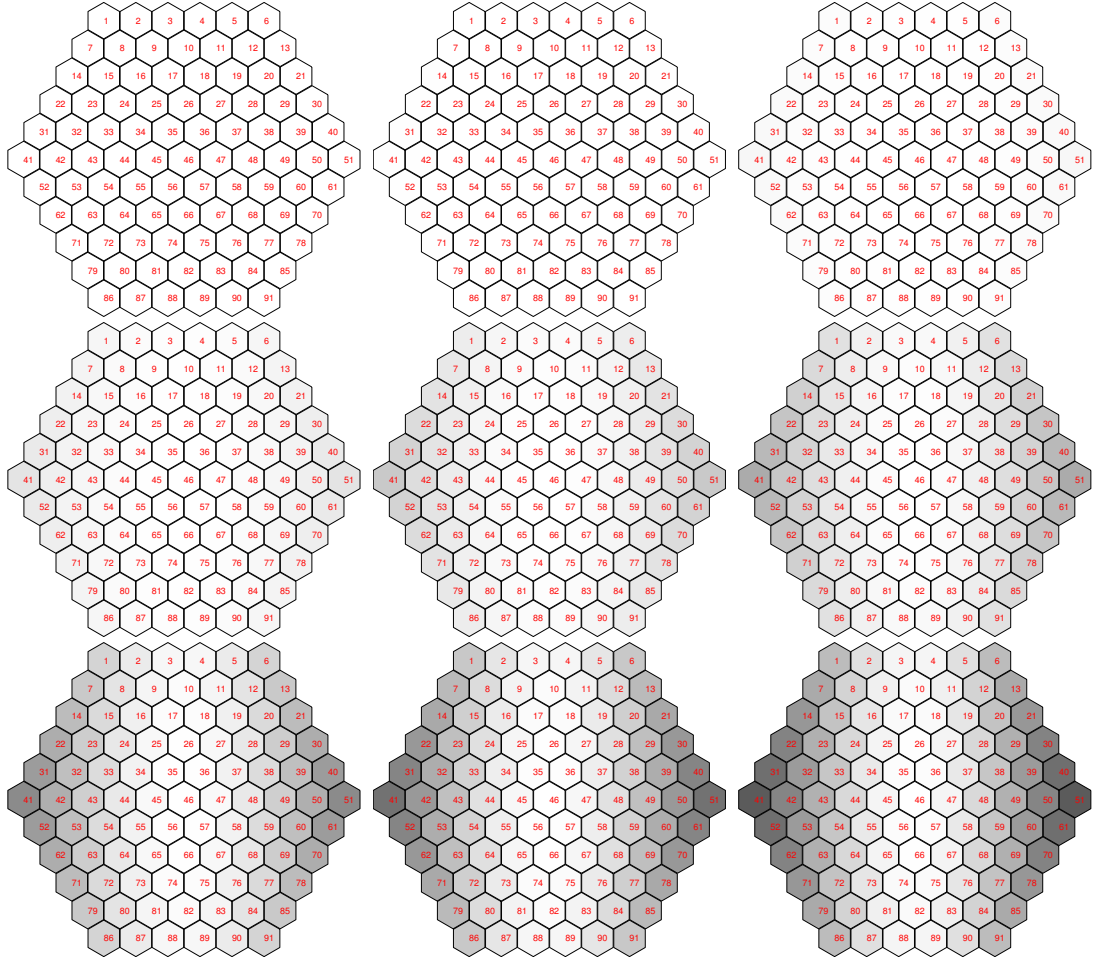## B.2. Tilting the Lens around One Axis

Figure B.2.: In this case, the lens is at the ideal position but the mirror facets are tiltd from 0 to 0.5 mrad

Bibliography

[1] R. L. Diehl, "Particle acceleration in cosmic sites," *The European Physical Journal D*, vol. 55, pp. 509–518, July 2009.

[2] "Astrophysics with H.E.S.S." `http://www.mpi-hd.mpg.de/hfm/HESS/pages/about/physics/`.

[3] "Homepage of the CTA Experiment." `http://www.cta-observatory.org`.

[4] "Homepage of the MAGIC Telescopes." `http://magic.mppmu.mpg.de/`.

[5] "Homepage of the VERITAS Experiment." `http://veritas.sao.arizona.edu/`.

[6] "Homepage of the H.E.S.S. Experiment." `http://www.mpi-hd.mpg.de/hfm/HESS/`.

[7] D. Michele, "CTA—A project for a new generation of Cherenkov telescopes," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 630, no. 1, pp. 285–290.

[8] F. Aharonian, J. Buckley, T. Kifune, and G. Sinnis, "High energy astrophysics with ground-based gamma ray detectors," *Reports on Progress in Physics*, vol. 71, p. 096901, Aug. 2008.

[9] R. M. Wagner, E. J. Lindfors, A. Sillanpää, and S. Wagner, "The CTA Observatory," *arXiv*, vol. astro-ph.IM, Dec. 2009.

[10] W. Hofmann, "Performance Limits for Cherenkov Instruments," *arXiv.org*, vol. astro-ph, Mar. 2006.

[11] T. Stanev, *High Energy Cosmic Rays*. Springer, 2nd ed., 2010.

[12] F. Schmidt, "Corsika shower images." `http://www.ast.leeds.ac.uk/~fs/showerimages.html`.

[13] D. Tescaro, "Timing analysis of the MAGIC telescope data after the installation of the ultra-fast 2 GSamples/s FADC readout," 2007.

[14] A. M. Hillas, "Cerenkov light images of EAS produced by primary gamma," in *International Cosmic Ray Conference* (F. C. Jones, ed.), vol. 3 of *International Cosmic Ray Conference*, pp. 445–448, Aug. 1985.

[15] "Homepage Statistical Methods and Analysis Techniques in Experimental Physics." `http://ihp-lx.ethz.ch/CompMethPP/main.html`.

[16] D. Eidemüller, "Hadron Suppression using a Cluster Finder Algorithm for the H.E.S.S. Telescope System," Diplomarbeit, Humbolt-Universität zu Berlin, 2002.

[17] A. Daum *et al.*, "First results on the performance of the HEGRA IACT array," *Astroparticle Physics*, vol. 8, no. 1-2, pp. 1–11.

[18] A. Kohnle *et al.*, "Stereoscopic imaging of air showers with the first two HEGRA Cherenkov telescopes," *Astroparticle Physics*, vol. 5, no. 2, pp. 119–131, 1996.

[19] V. Vassiliev, S. Fegan, and P. Brousseau, "Wide field aplanatic two-mirror telescopes for ground-based γ-ray astronomy," *Astroparticle Physics*, vol. 28, pp. 10–27, Sept. 2007.

[20] J. Knapp, "The Cherenkov Telescope Array -Status-." Presentation Slides, Jan. 2010. `http://hep.ph.liv.ac.uk/~green/cta/LiverpoolMeeting/TeVTalks/Johannes.pdf`.

[21] A. Schliesser and R. Mirzoyan, "Wide-field prime-focus imaging atmospheric Cherenkov telescopes: A systematic study," *Astroparticle Physics*, vol. 24, no. 4-5, pp. 382–390, 2005.

[22] J. Davies and E. S. Cotton, "Design of the quartermaster solar furnace," *Solar Energy*, vol. 1, no. 2-3, pp. 16–22, 1957.

[23] C. E. Max, B. A. Macintosh, S. S. Olivier, D. T. Gavel, and H. W. Friedman, "Observing techniques for astronomical laser guide star adaptive optics," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* (D. Bonaccini & R. K. Tyson, ed.), vol. 3353 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pp. 277–281, Sept. 1998.

[24] A. McCann, D. Hanna, J. Kildea, and M. McCutcheon, "A New Mirror Alignment System for the VERITAS Telescopes," *Astropart. Phys.*, vol. 32, pp. 325–329, 2010.

[25] R. Cornils *et al.*, "The optical system of the H.E.S.S. imaging atmospheric Cherenkov telescopes. Part II: mirror alignment and point spread function," *Astroparticle Physics*, vol. 20, pp. 129–143, Nov. 2003.

[26] A. Biland, M. Garczarczyk, H. Anderhub, V. Danielyan, D. Hakobyan, E. Lorenz, and R. Mirzoyan, "The Active Mirror Control of the MAGIC Telescope," *arXiv.org*, vol. astro-ph, Sept. 2007.

[27] S. Darula, R. Kittler, and C. A. Gueymard, "Reference luminous solar constant and solar luminance for illuminance calculations," *Solar Energy*, vol. 79, pp. 559–565, Nov. 2005.

[28] The CTA Consortium, "Design concepts for the Cherenkov Telescope Array CTA: an advanced facility for ground-based high-energy gamma-ray astronomy," *Experimental Astronomy*, pp. 1–124, Nov. 2011.

[29] R. Platzer, "Mirror layout (status 06/04/2010) as a physical perfect arrangement of mirrors in 3D-CAD-data," tech. rep., Deutsches Elektronen–Synchrotron, 2010.

[30] S. Steiner. Internally provided CAD Design Drawings.

[31] *Lensagon BHR16012S12 Data Sheet*, feb 2010. `http://www.lensation.de/images/PDF/BHR16012S12.pdf`.

[32] I. Micron Technology, *1/3-Inch Wide VGA CMOS Image Sensor*, feb 2007.

[33] "leanXcam project page." `http://leanxcam.origo.ethz.ch`.

[34] NXP Semiconductors, *UM10204 I^2C-bus specification and user manual.* `http://www.nxp.com/documents/user_manual/UM10204.pdf`.

[35] `http://www.virtualbox.org`.

[36] Bundesamt für Landestopografie, *Plan of Site.* `http://map.geo.admin.ch`.

[37] Aptina Imaging Corporation, *1/3-Inch Wide-VGA CMOS Digital Image Sensor MT9V032 Data Sheet*, May 2011. `http://www.aptina.com/products/image_sensors/mt9v032c12stm/`.

[38] ORIENTAL MOTOR (EUROPA) GmbH, *High-Torque 2-Phase Stepping Motor Unit; CSK Series.* `http://www.orientalmotor.de/Downloads/`.

[39] Philips Semiconductors, *PCF8574, Remote 8-bit I/O expander for I^2C-bus.* `http://www.nxp.com/documents/data_sheet/PCF8574.pdf`.