

Echtzeit-Regelung eines optischen Systems für Cherenkov-Teleskope

(Automated mirror control for Imaging Atmospheric Cherenkov Telescopes)

von

Jan Henning Drewes

geboren am

03. Dezember 1986

Bachelorarbeit im Studiengang Physik
Universität Hamburg

2010

1. Gutachter: Prof. Dr. Dieter Horns
2. Gutachter: Dr. Martin Tluczykont

Kurzfassung

Echtzeit-Regelung eines optischen Systems für Cherenkov-Teleskope

Die vorliegende Arbeit beschreibt die Entwicklung eines Regelkreises zur Echtzeit-Regelung des Spiegelsystems von Cherenkov-Teleskopen. Hierzu wurde ein Teststand errichtet, bei dem die relative Position eines Spiegelsegments zu der, durch eine LED-Matrix simulierten, Hauptkamera mit einer am Spiegelsegment angebrachten CCD-Kamera überprüft wird. Hierzu wurde ein Programm in Python geschrieben und getestet, das die Position einer LED auf den Aufnahmen der CCD-Kamera bestimmen kann. Mit Hilfe einer berechneten Transformationsmatrix kann der Spiegel wie gewünscht gesteuert werden. Bei der Errichtung des Teststandes und der Steuerung der Spiegelmotoren wurde auf Komponenten des H.E.S.S. Projekts zurückgegriffen. Abschließend wurde die Spiegelsteuerung testweise direkt in die Bildauswertung eingebettet und so ein Regelkreis eingerichtet.

Abstract

Automated Mirror Control for Imaging Atmospheric Cherenkov Telescopes

The present thesis describes the development of a control circuit for the automated mirror alignment of Imaging Atmospheric Cherenkov Telescopes (IACTs). At first a experimental set-up was built where the telescopes main camera is simulated by an LED-matrix. The position of the main camera relative to a mirror segment is checked with an CCD-camera, mounted directly on the mirror segment. Therefor an image analysing programm was written in Python to locate the LED. The mirror segment can be driven with a calculated transformation matrix. The mirror segment and the programm for mirror control were taken from the H.E.S.S. project. Finally the motor control was implemented in the image analysing programm to establish a control circuit.

Inhaltsverzeichnis

1	Einleitung	7
2	Grundlagen	9
2.1	Cherenkov-Teleskope	9
2.2	Justierung der Teleskope bei H.E.S.S.	10
2.3	Regelkreis	11
3	Aufbau des Experiments	13
3.1	Aufbau	13
3.2	Spiegelhalterung	14
3.3	USB-CCD-Kamera DMK 31AU03.AS	15
3.3.1	Bestimmung des Gains der Kamera bei den verwendeten Einstellungen	17
3.3.2	Objektiv und Auflösung	19
3.4	LED-Matrix	20
4	Programm zur Bildanalyse	22
4.1	Struktur	22
4.2	Ablauf der Bildanalyse	22
4.3	Erreichte Genauigkeit der Positionsbestimmung	24
4.4	Langzeitmessung zum Verhalten des Teststandes	25
5	Spiegelsteuerung	27
5.1	Berechnung der Transformationsmatrix	27
5.1.1	Winkelauflösung der Motorsteuerung	29
5.2	Ansteuerung der Spiegelmotoren und Test der Transformationsmatrix	29
5.3	Benötigte Zeit für die Ansteuerung der Motoren	30
5.4	Einrichtung und Test eines Regelkreises	31
6	Zusammenfassung und Ausblick	34
A	Quelltext des Programms zur Bildanalyse	36
A.1	Quelltext des Hauptprogramms	36
A.2	Quelltext des Unterprogramms	36

Kapitel 1

Einleitung

Die Erforschung des Universums im Bereich der hochenergetischen γ -Strahlung, also Photonen mit Energien oberhalb von einigen MeV, umfasst zahlreiche grundlegende Fragen der Astro- und Teilchenphysik. Dazu gehört die Untersuchung von Phänomenen wie Aktiven Galaxien, Schwarze Löcher und nicht zuletzt die Suche nach der Dunklen Materie. Bei all diesen Phänomenen wird hochenergetische γ -Strahlung ausgesendet. Da diese Strahlung für das menschliche Auge nicht sichtbar ist, eignen sich Teleskope im optisch sichtbaren Bereich nicht zur Beobachtung. Zudem erreicht die γ -Strahlung nicht die Erdoberfläche, sondern reagiert in Teilchenprozessen mit der Erdatmosphäre. Zum einen verwendet man daher satellitengestützte Teleskope für die Beobachtung im γ -Bereich.

Gerade die Teilchenprozesse in der Erdatmosphäre erlauben jedoch die zusätzliche Verwendung von erdgebundenen Teleskopen. Dazu wird Licht, das von Sekundärteilchen ausgeht, von sogenannten Cherenkov-Teleskopen aufgefangen. Die Analyse des Cherenkov-Lichts lässt Rückschlüsse auf Herkunftsrichtung und Energie der primären Teilchen zu.

Um mit dem Energiebereich der erdgebundenen Teleskope an den der Satellitenteleskope anzuschließen, und somit die Datenlage zu verbessern und zu erweitern, befinden sich neue Cherenkov-Teleskope derzeit in Planung¹. Diese Teleskope werden zum Teil doppelt so große Spiegelflächen und Fokallängen besitzen, wie bereits bestehende Systeme². Damit geht eine größere Belastung der Konstruktion durch das, von Schwerkraft- und Wettereinflüssen ausgeübte, Drehmoment einher. Dadurch nimmt die Qualität der Abbildungsfunktion ab. Dem kann entgegengewirkt werden, indem man Verzerrungen der Teleskopkonstruktion durch eine aktive Nachführung des Spiegelsystems ausgleicht³. Das Spiegelsystem besteht dabei aus einer Vielzahl von Facetten, die einzeln angesteuert werden können.

Ziel dieser Bachelorarbeit ist es ein System zur automatischen Regelung dieser Nachführung mit Hilfe eines Kamerasystems zu entwickeln.

¹darunter CTA und H.E.S.S. 2.

²Z.B. haben die H.E.S.S.-Teleskope einen Spiegeldurchmesser von 11,73 m und eine Fokallänge von 15,03 m. Das sich derzeit im Bau befindliche H.E.S.S 2 Teleskop wird einen Spiegeldurchmesser von ca. 25 m besitzen.

³Dies geschieht bereits bei den MAGIC-Teleskopen, siehe [Biland u. a. \(2008\)](#)

Dies umfasst die Programmierung einer Bildanalysesoftware, mit der die Ausrichtung der Spiegelfacetten anhand von Aufnahmen von CCD-Kameras überwacht werden und nachgeführt werden kann. den Aufbau eines Teststandes, an dem die Genauigkeit der Bildanalyse überprüft, sowie die Steuerung der Motoren und die automatische Regelung der Nachführung getestet werden können. Teilweise konnte dabei auf Komponenten zurückgegriffen werden, die für das H.E.S.S. Observatorium in Namibia entwickelt wurden.

Anmerkung

Im Laufe der Arbeit hat sich gezeigt, dass die Zeit nicht ausreicht einen ausgereiften Regelkreis umfassend zu testen und an die Erforderlichkeiten von CTA anzupassen. Es wurde jedoch ein erster Entwurf eines Regelkreises entwickelt und getestet, auf dem aufgebaut werden kann.

Kapitel 2

Grundlagen

2.1 Cherenkov-Teleskope

Aktuell gibt es weltweit mehrere Projekte¹, die abbildende Cherenkov-Teleskope zur Untersuchung hochenergetischer, kosmischer γ -Strahlung verwenden. Der Nachweis der kosmischen Photonen geschieht mit der Abbildung des Lichts von Luftschauern. Dieses entsteht, wenn ein γ -Photon der Kosmischen Strahlung auf Teilchen der Erdatmosphäre trifft und einen Teilchenschauer auslöst. Geladene Sekundärteilchen, die sich mit der Geschwindigkeit $v = \beta \cdot c$ fortbewegen, können sich in der Erdatmosphäre mit der Brechzahl $n > 1$ schneller als mit der Phasengeschwindigkeit des Lichts² ausbreiten.

Durch die Polarisierung von Atomen entlang ihres Weges rufen die Sekundärteilchen die Aussendung eines Lichtkegels mit dem Öffnungswinkel Θ hervor. Dies wird als Cherenkov-Effekt bezeichnet.

$$\cos(\Theta) = \frac{1}{n\beta} \quad (2.1)$$

Abbildende Cherenkov-Teleskope³ nutzen diesen Effekt, indem sie das Cherenkovlicht eines Teilchenschauers mit einem Spiegelsystem sammeln und auf eine Anordnung von Photomultipliern (die Hauptkamera) fokussieren. Um die lichtschwachen Luftschauber abzubilden, wird eine möglichst große Sammelfläche benötigt. Da technische und finanzielle Gründe die Größe einer zusammenhängenden Spiegelfläche beschränken, wird eine Vielzahl einzelner Spiegelfacetten zu einem Reflektor angeordnet. So werden zum Beispiel die Reflektorflächen der vier H.E.S.S.-Teleskope [Abb. 2.1] aus jeweils 380 Spiegelfacetten gebildet⁴.

Die untere Grenze des abgedeckten Energiebereichs ist abhängig von der Größe der Reflek-

¹besonders erfolgreich sind hierbei MAGIC auf La Palma und H.E.S.S. in Namibia.

²Die Phasengeschwindigkeit $c' = c/n$ von Licht in Materie ist geringer als die Vakuumlichtgeschwindigkeit c , die die Obergrenze für Teilchengeschwindigkeiten bildet.

³engl. Imaging Atmospheric Cerenkov Telescopes (IACT's).

⁴[H.E.S.S.-Kollaboration \(2010\)](#)

torfläche eines Teleskops, während die effektive Fläche des gesamten Teleskopsystems die obere Grenze des Energiebereichs bestimmt. Letzteres resultiert aus der geringen Anzahl hochenergetischer γ -Ereignisse. Durch die stereoskopische Verwendung mehrerer Teleskope kann die Einfallsrichtung des Lichtkegels und damit der primären γ -Photonen bestimmt werden.



Abbildung 2.1: Zwei der insgesamt vier H.E.S.S.-Teleskope in Namibia. Bildquelle: H.E.S.S.-Kollaboration.

Die nächste Generation von Cherenkov-Teleskopen soll die Empfindlichkeit für höhere Energien erhöhen und gleichzeitig die Energieschwelle zur Detektion von γ -Strahlung niedriger Energie absenken. Zusätzlich soll die Winkelauflösung verbessert werden. Um all dies zu erreichen wird bei der Planung des Cherenkov Telescope Array (CTA) der Ansatz eines sich ergänzenden System aus Teleskopen mit unterschiedlichen Spiegelflächen verfolgt. Dabei sollen Einzelteleskope in drei verschiedenen Größen zum Einsatz kommen, deren Aufbau sich an bestehenden Cherenkov-Teleskopen orientieren soll: 1. viele kleine Teleskope, die über eine große Fläche verteilt sind, 2. in einem Bereich um das Zentrum herum Teleskope mittlerer Größe (in etwa den H.E.S.S. Teleskopen entsprechend) und 3. in der Mitte vier sehr große Teleskope mit einer geplanten Fokallänge zwischen 20 und 30 m⁵.

2.2 Justierung der Teleskope bei H.E.S.S.

Bei den Teleskopen des H.E.S.S. Projekts wird eine automatische Justierung mit Hilfe eines Kamerasystems durchgeführt. Dabei werden Justiersterne ausgewählt, deren Abbild mit den Spiegelfacetten auf den Deckel der Hauptkamera reflektiert wird. An jedem Teleskop ist eine Kamera, die sogenannte LidCCD⁶, angebracht, die die Sternabbilder beobachtet. Das Mirror Alignment Control System (kurz: MACS) verfährt nun jeden Spiegel einzeln bis das Abbild des Sterns in einem Punkt fokussiert ist. Dazu ist jedes Spiegelsegment an drei Punkten aufgehängt, von denen zwei beweglich sind und mit Hilfe

⁵Wagner u. a. (2009)

⁶Dreyling-Eschweiler (2010)

von Motoren aus der Automobilindustrie betrieben werden. Die Spiegelfacetten besitzen also jeweils zwei Freiheitsgrade. Diese Methode nimmt, je nach Wetterlage, mehrere Beobachtungsnächte in Anspruch. Durch die LidCCD ist eine Überprüfung der Justierung jederzeit möglich. Die Erfahrung hat gezeigt, dass sich bei H.E.S.S auch nach längerer Zeit die Abbildungsfunktion nicht gravierend verschlechtert.

Bei einem wesentlich größeren Teleskop, stellt sich die Situation jedoch anders dar. Mit einer größeren Fokallänge wird bei den Teleskopen von CTA zwangsläufig eine höhere gravitative Belastung der Konstruktion einhergehen. Der Grad der mechanischen Verzerrungen der Teleskopstruktur hängt dabei natürlich im speziellen von deren Konstruktion und den verwendeten Materialien ab. Aus Kostengründen wird jedoch keine beliebig hohe Stabilität zu erreichen sein. Deshalb ist damit zu rechnen, dass die Qualität der Abbildungsfunktion während des Betriebs abhängig von der Elevation des jeweiligen Teleskops abnehmen wird. Es kann davon ausgegangen werden, dass sich die Abbildungsfunktion in größerem Maße verschlechtern wird als dies bei kleineren Teleskopen, wie zum Beispiel bei denen von H.E.S.S, der Fall ist. Dies motiviert die Entwicklung einer aktiven Nachregelung des Spiegelsystems um die auftretenden Verzerrungen durch die Nachführung einzelner Spiegelfacetten zu kompensieren⁷.

Die Grundidee ist, die Position der Hauptkamera bezüglich jedes einzelnen Spiegels nach der ersten Justierung als Sollwert zu speichern und während des Betriebs fortwährend zu kontrollieren. Eine Rückkopplung mit den Motoren an den Spiegelsegmenten soll gewährleisten, dass Positionsänderungen der Hauptkamera durch Verstellungen der einzelnen Spiegelsegmente entgegengewirkt werden kann. Die beschriebene Idee lässt sich durch einen Regelkreis verwirklichen. Zur optischen Kontrolle der Position der Hauptkamera könnte an jeder Spiegelfacette eine CCD-Kamera angebracht sein. Als Referenzpunkte für die Positionsbestimmung können, wie es bereits bei H.E.S.S. der Fall ist, LEDs um die Hauptkamera herum angeordnet werden, deren Bewegung von einem Programm anhand von Kontrollbildern analysiert wird. Dieses würde dann nötigenfalls eine Nachführung der Spiegel veranlassen, indem es mit der Spiegelsteuerung kommuniziert.

2.3 Regelkreis

Das Ziel dieser Bachelorarbeit ist die Entwicklung einer automatisierten Steuerung eines Spiegelsystems für Cherenkov-Teleskope.

Aus diesem Grund soll an dieser Stelle kurz die allgemeine Funktionsweise eines Regelkreises erklärt werden. An späterer Stelle wird darauf eingegangen, wie der entwickelte Aufbau als Regelkreis verwendet werden kann.

Als **Regelkreis** bezeichnet man einen geschlossenen Wirkungskreis zur Steuerung einer physikalischen Größe. Im wesentlichen besteht ein Regelkreis aus den drei Komponenten

⁷Die MAGIC-Teleskope auf La Palma besitzen bereits ein solches System zur Regelung ihres Spiegelsystems. Quelle: www.magic.mppmu.mpg.de

Regler, Regelstrecke und Rückführung (Messglied):

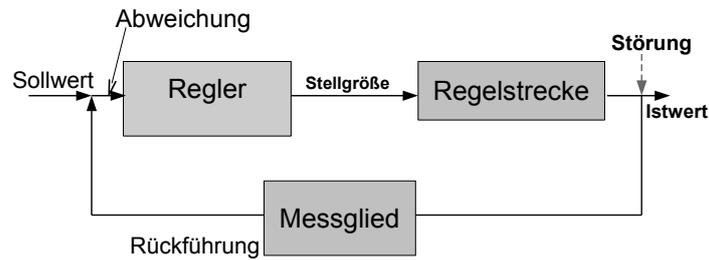


Abbildung 2.2: Flussdiagramm eines Regelkreises.

- Die **Rückführung** dient zur ständigen Kontrolle der zu regelnden Größe (hier die Stellung eines Spiegelsegments). Dies wird dadurch gewährleistet, dass die Größe ständig gemessen wird und ihr Istwert an den Regler übermittelt wird.
- Der **Regler** ist in der Lage die zu regelnde Größe selbstständig zu beeinflussen. Dazu vergleicht er den Istwert der Größe mit deren Sollwert und berechnet daraus die **Regelabweichung**, die er in eine **Stellgröße** (hier: Hallcounts, die angeben wie weit ein Spiegelmotor gefahren werden soll) überführt.
- Auf der **Regelstrecke** wird die Stellgröße in eine Beeinflussung der zu regelnden Größe umgesetzt. Im vorliegenden Fall bedeutet dies eine Ansteuerung der Motoren, die zu einer Veränderung der Spiegelposition führt.

Kapitel 3

Aufbau des Experiments

3.1 Aufbau

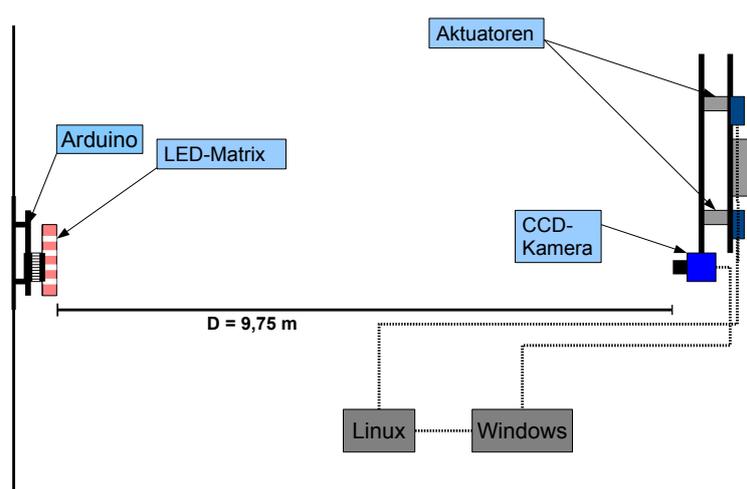


Abbildung 3.1: Aufbau des Teststandes. Bei dem Arduino, der mit einer Halterung an der Laborwand angebracht ist, handelt es sich um einen Mikrocontroller, der zur Steuerung der LED-Matrix dient. Auf der rechten Seite befindet sich der Spiegelprototyp. Dieser ist auf einer Metallplatte verschweißt die ebenfalls an der Wand angebracht ist. Die Spiegelplatte selbst ist an drei, in Kugelgelenken gelagerten Haltepunkten aufgehängt. Zwei dieser Haltepunkte sind über Motor betriebene Aktuatoren verstellbar, so dass der Spiegel in zwei Richtungen bewegt werden kann. An der unteren Kante des Spiegels befindet sich eine CCD-Kamera, die zur Positionskontrolle auf die LED-Matrix ausgerichtet ist. Die Steuerung der Motoren, sowie die Analyse der Aufnahmen der CCD-Kamera übernimmt ein Linux-Notebook, während die CCD-Kamera von einem Windows-Notbook gesteuert wird.

Zum Test des Bildanalyseprogramms, sowie der Spiegelsteuerung anhand von Bilddaten, wurde ein Teststand errichtet [Abb. 3.1]. Auf der einen Seite des Labors befindet sich an der Wand ein Spiegelprototyp. An der Unterseite der als Spiegelerersatz dienenden Platte ist eine CCD-Kamera montiert. Gegenüber des Spiegels befindet sich in 9,75 m Entfernung

eine LED-Matrix auf einem programmierbaren Arduino-Board, die als Attrappe für die Hauptkamera des Teleskops dient. Die CCD-Kamera wird von einem Windows-Notebook gesteuert, das die Aufnahmen auf einem Linux-Notebook speichert, auf dem das eigentliche Analyseprogramm läuft und mit dem die Spiegelfacetten gesteuert werden kann. Die Verwendung zweier unterschiedlicher Betriebssysteme wurde notwendig, da für die CCD-Kamera keine Linux-Treiber zur Verfügung standen.

Verwendung des Aufbaus als Regelkreis

Die Verwendung des Aufbaus als Regelkreis [Abb. 3.10] wird dadurch ermöglicht, dass die CCD-Kamera bzw. das Analyseprogramm die Positionsänderung der LED überwacht. Die Einheit CCD-Kamera und Analyseprogramm bildet also das Messglied und die Rückführung des Regelkreises. Das Analyseprogramm dient gleichzeitig auch als Regler, da es die Abweichung zwischen Ist- und Sollwert der Position der LED mit Hilfe der in Kapitel 5.1 berechneten Transformationsmatrix in eine Stellgröße (Hallcounts) überführt und diese an die Motorsteuerung übergibt, die die Regelstrecke darstellt.

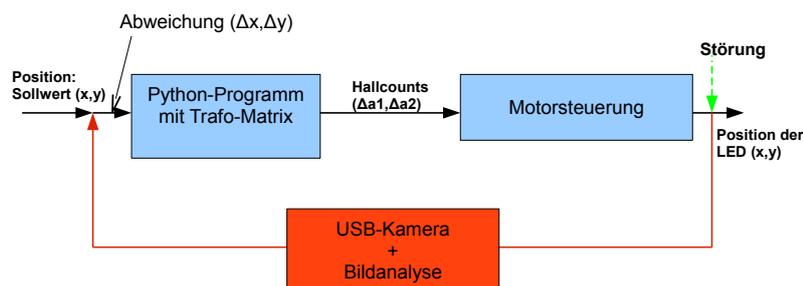


Abbildung 3.2: Schematische Darstellung der Funktion der CCD-Kamera und der Softwarekomponenten des Aufbaus in dem entwickelten Regelkreis.

3.2 Spiegelhalterung

Bei der verwendeten Spiegelhalterung [Abb. 3.3] handelt es sich um einen Prototypen, der ursprünglich für H.E.S.S entwickelt wurde. Der Spiegel selbst wird hier durch eine Metallplatte ersetzt, die aber im Folgenden als „Spiegel“ bezeichnet wird. Diese ist an drei Aufhängepunkten mit der übrigen Konstruktion verbunden. Zwei der Punkte sind durch Aktuatoren höhenverstellbar. Die Aktuatoren werden mit Gleichspannungsmotoren aus der Automobilindustrie bewegt. Durch eine Kugelgelenklagerung aller drei Haltepunkte ist eine spannungsfreie Bewegung des Spiegels in allen Raumrichtungen möglich. Informationen über die Zahl der gefahrenen Umdrehungen der Motoren liefern Hall-Sensoren, die auf der Motorwelle angebracht sind. Die Angabe der Umdrehung erfolgt in Hallimpulsen

(Hall-Counts)¹.

Die Ansteuerung der beiden Aktuatormotoren erfolgt über eine Steuereinheit wie sie bei H.E.S.S verwendet wird. Diese ist an einen Rechner (VMEbus) angeschlossen, der wiederum in einem lokalen Netzwerk mit dem Rechner für die Bildauswertung verbunden ist. Die Steuerung der Motoren kann von letzterem übernommen werden. Dafür ist auf dem Rechner die Steuerungssoftware *Mirror Alignment Control System* (MACS) installiert, die bei H.E.S.S zur automatischen Justierung der Teleskope dient². In dieser Arbeit wird sie nur zur Steuerung der Aktuatormotoren verwendet.

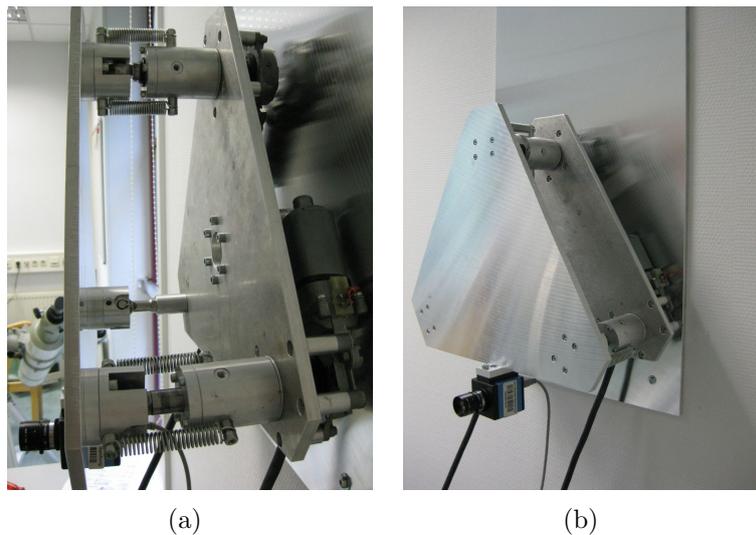


Abbildung 3.3: Spiegelhalterung: Zu sehen ist an die Laborwand montierte Prototyp der Spiegelhalterung. (b) An der Unterseite des Spiegels ist die CCD-Kamera angebracht. (a) Drei Aufhängepunkte halten den Spiegel. Der Obere und der Untere sind bewegliche Aktuatoren, mit denen der Spiegel verstellt werden kann. Hinter der zweiten Platte befinden sich die beiden Motoren.

3.3 USB-CCD-Kamera DMK 31AU03.AS

Im folgenden soll ein tabellarischer Überblick über die Eigenschaften der verwendeten USB-Kamera DMK 31AU03.AS [Abb. 3.4] gegeben werden [Tab. 3.1], soweit diese vom Hersteller The ImagingSource angegeben wurden.

Ein Nachteil der Kamera ist, dass der als Freeware erhältliche Linux-Treiber, mittlerweile das Gerät nicht mehr erkennt³. Aus diesem Grund ist es nötig die Kamera mit einem Notebook anzusteuern auf dem ein Windows-Betriebssystem installiert ist. Dieses speichert die aufgenommenen Bilder in einem freigegebenen Ordner auf einem zweiten Notebook. Auf

¹siehe [Dreyling-Eschweiler \(2010\)](#)

²siehe Kapitel 4 in [Cornils \(2001\)](#)

³In der Dokumentation des Treibers war die Kamera zum Zeitpunkt des Aufbaus des Experiments noch aufgeführt. Die Nachfrage beim Hersteller der Kamera ergab, dass die Deviceklasse der Kamera geändert wurde.



Abbildung 3.4: CCD - Kamera DMK 31AU03.AS mit USB-Verbindung. Die Abbildung zeigt die Kamera bereits an den Spiegel montiert. Die Stromversorgung der Kamera erfolgt über das USB-Kabel.

Eigenschaft	Wert der Eigenschaft
Product Code	DMK 31AU03.AS
Type	monochrome USB camera without IR cut filter
Connection	USB
Sensor	CCD
Type	progressive scan
Sensor specification	Sony ICX204AL PDF
Format	1/3"
Resolution	1024 x 768 pixel
Dynamic range	8 bit
Exposure time	1/10000 s to 60 minutes
Gain	0 dB to 36 dB
FPS @ Max resolution	30 fps
Video formats @ Frame rate	1024 x 768, Y800 @ 30, 15, 7.5, 3.75 fps
Dimensions (H x W x L)	50.6 mm x 50.6 mm x 50 mm
Mass	265 g

Tabelle 3.1: Auszug aus dem Datenblatt der DMK 31AU03.AS [Quelle: The Imaging Source]

diesem ist ein Linux-Betriebssystem installiert unter dem das Bildanalyseprogramm läuft. Im Hinblick einer Verwendung im tatsächlichen Experiment ist diese Lösung mit verschiedenen Betriebssystemen sicherlich zu umständlich, es gibt jedoch genügend Kameras für die Linux-Treiber verfügbar sind.

Ansteuerung der Kamera

Zur Bildnahme mit der Kamera wird das mitgelieferte Programm IC.Capture verwendet⁴. Hier lassen sich die Kameraparameter einstellen. Das Programm verfügt über einen Bildfolge-Timer. Mit diesem kann man wahlweise alle Bilder der Kamera im Videomodus speichern oder ein Intervall festlegen, in dem Bilder aufgenommen werden. Der kleinste

⁴Die von The Imaging Source auf CD mitgelieferte Software umfasst Windows-Treiber und das Programm IC.Capture

mögliche Intervall ist dann die Aufnahmen eines Bildes pro Sekunde. Im hier verwendeten Videomodus lässt sich die Rate mit der Bilder gespeichert werden über die Einstellung der Bildwiederholungsrate der Kamera regeln [Tab. 3.1]. Im Hinblick auf die begrenzte Geschwindigkeit des Bildanalyseprogramms (etwa 2 *fps*⁵) wird eine Bildwiederholungsrate von 7.5 *fps* gewählt. Damit ist auf jeden Fall bei jedem Neustart des Bildanalyseprogramms mindestens ein Bild zur Analyse vorhanden. Probleme kann eine zu langsame Übertragung der Bilder vom Windows Rechner an den Linux Rechner bereiten⁶.

3.3.1 Bestimmung des Gains der Kamera bei den verwendeten Einstellungen

Bei der Positionsbestimmung des Schwerpunktes des hellen Punktes auf den Aufnahmen werden die einzelnen Pixel mit ihrem Helligkeitswert A_i gewichtet. Hierbei wird jedoch nicht der in der Bilddatei gespeicherte Wert in ADU (Analog-to-Digital Unit. Hier der Graustufenwert in Stufen von 0 bis 255) verwendet, sondern die Anzahl der entsprechenden im Halbleiter ausgelösten Photoelektronen n_i , da deren Fehler mit Hilfe einer Poisson-Verteilung abgeschätzt werden kann. Um aus dem Wert in ADU die Anzahl der Photolektronen zu erhalten wird zunächst der Verstärkungsfaktor der Kamera, der Gain $g = n_i/A_i$, bestimmt.

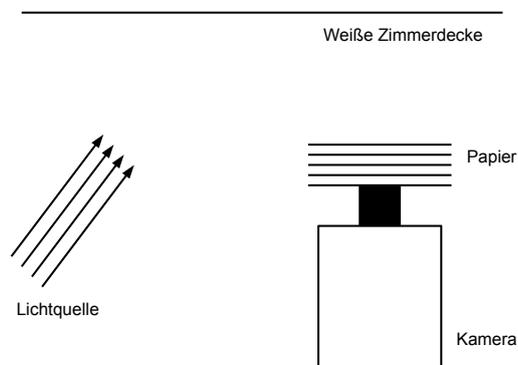


Abbildung 3.5: Schematischer Versuchsaufbau zur Bestimmung des Gains.

Die Messung zur Bestimmung des Gains wird ähnlich einer Anleitung⁷ der Firma Apogee durchgeführt [Abb. 3.5]. Das Objektiv wird von der Kamera entfernt und die Kamera so platziert, dass die Vorderseite des CCD-Chips zur Decke weist. Damit der gesamte CCD-Chip homogen ausgeleuchtet wird, werden auf der Kameraöffnung fünf Lagen wei-

⁵fps (engl. frames per second), bezeichnen die Anzahl der pro Sekunde von einer Kamera aufgenommenen Bilder.

⁶Wenn beim Start des Programms das ausgewählte Bild unvollständig ist, kommt es zu einer Fehlermeldung. Solange dies nicht zu häufig auftritt ist eine Analyse dennoch möglich.

⁷Auf der Webseite von Apogee findet sich unter der Rubrik CCD University ein Kapitel CCD Test Methods, in dem dieses Verfahren beschrieben wird [Apogee \(2010\)](#)

ßes Papier platziert. Als diffuse Lichtquelle dient die Zimmerdecke, die durch ein Fenster von außen beleuchtet wird. Es wird eine Bildfolge mit unterschiedlicher Stapeldicke (5 bis 1 Bögen) aufgenommen [Abb. 3.6], um eine unterschiedliche Erzeugungsrate von Photoelektronen zu erhalten. Die Integrationszeit bleibt bei allen Aufnahmen konstant bei $1/30s$.

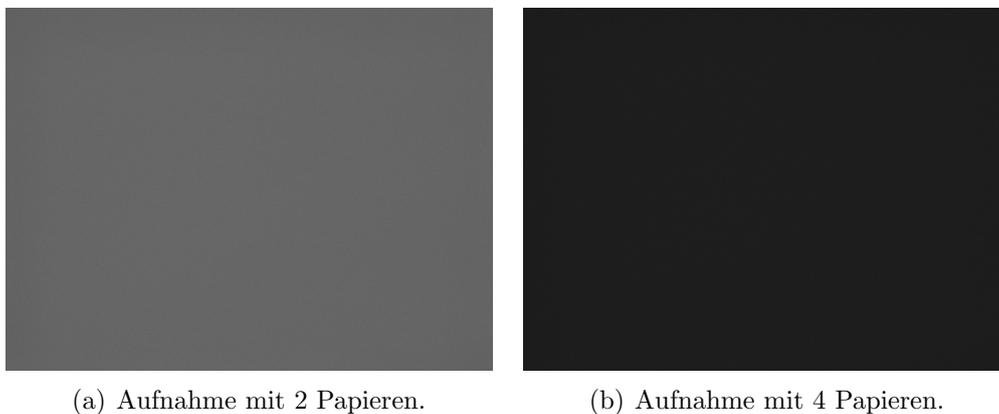


Abbildung 3.6: Die abgebildeten Aufnahmen (a),(b) dienen zur Bestimmung des Verstärkungsfaktors (des Gains) der Kamera. Um eine unterschiedliche Rate von Photoelektronen zu erreichen wird die Anzahl der, die Kameraöffnung bedeckenden, Lagen Papier erhöht. Es wurden fünf Aufnahmen mit 1 bis 5 Lagen Papier verwendet.

Zur Bestimmung des Gains wird von allen Aufnahmen ein Histogramm erstellt. Aus diesem kann sowohl der Mittelwert der Helligkeit A , als auch deren Varianz $var(A)$ berechnet werden [Abb. 3.7, Tab. 3.2].

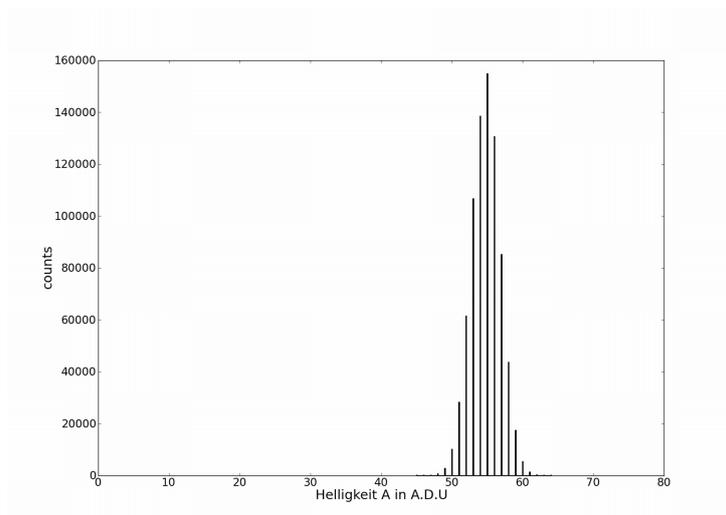


Abbildung 3.7: Exemplarisches Histogramm bei einer Stapeldicke von 3 Bögen Papier.

Trägt man nun A gegen $var(A)$ auf, so erhält man eine Gerade, deren Steigung gerade der Gain ist. Mit linearer Regression erhält man einen Gain von $10,08 e^-/A.D.U.$.

var(A)	A
23,23	238,46
7,42	102,4
4,13	54,77
2,68	29,95
1,93	16,88

Tabelle 3.2: Messwerte zur Bestimmung des Gains der Kamera. Die entsprechenden Werte wurden einem Histogramm entnommen. Mit A wird hier der mittlere Graustufenwert der Aufnahme in ADU bezeichnet.

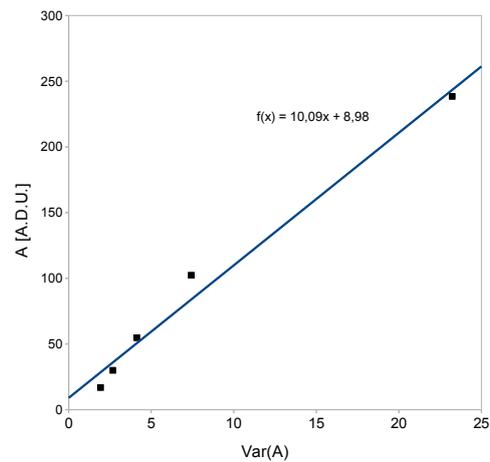


Abbildung 3.8: In dem Diagramm ist die mittlere Helligkeit A aufgetragen gegen deren Varianz $\text{var}(A)$. Mittels linearer Regression wird aus den Messwerten der Gain berechnet.

3.3.2 Objektiv und Auflösung



Abbildung 3.9: 16-mm-Objektiv der Firma Pentax.

Verwendet wird ein 16 mm Objektiv der Firma Pentax [Abb. 3.9]. Die Brennweite wurde dabei so gewählt, dass der Durchmesser einer LED ($G = 3 \text{ mm}$) in etwa einem Pixel ($B = 4,65 \text{ }\mu\text{m}$) auf der Aufnahme der CCD-Kamera entspricht. Der Zusammenhang ist,

$$f = \frac{B \cdot d}{G + B} . \quad (3.1)$$

Dabei ist $d \approx 9,75$ m der Abstand zwischen LED-Matrix und Objektiv. Um o.g. Kriterium für CTA zu erfüllen wird für das größte geplante Teleskop mit $d = 20$ bis 30 m ein 50 mm Objektiv benötigt.

3.4 LED-Matrix

Bevor die optische Bestimmung der Position eines Objekts, hier der Hauptkamera des Cherenkov-Teleskops, in Angriff genommen werden kann, stellt sich zwangsläufig die Frage, woran ein Analyseprogramm die Position schnell und zuverlässig erkennen soll. Eine Idee, die sich schon bei den H.E.S.S.-Teleskopen findet, ist die Verwendung eines Rahmens aus LED, die um die Hauptkamera herum angeordnet sind. Für diese Arbeit wird zunächst kein Rahmen verwendet. Vielmehr wird die Hauptkamera durch eine einzelne LED repräsentiert, die anhand ihrer Helligkeit von einem Programm als solche identifiziert wird. Diese Methode besitzt keine unbegrenzte Toleranz gegenüber der Umgebungshelligkeit, da gewährleistet sein muss, dass die LED auf den Aufnahmen das hellste Objekt ist. Da die Beobachtungen bei Dunkelheit durchgeführt werden stellt dies jedoch kein Problem dar. Dennoch wurden Tests hinsichtlich des Verhaltens bei veränderlicher Umgebungshelligkeit durchgeführt.

Die in dieser Arbeit verwendete LED-Matrix ermöglicht grundsätzlich auch einen Rahmen oder ein Gitter aus LEDs zu bilden und andere Verfahren zur Positionsbestimmung zu entwickeln, die weniger die Helligkeit als vielmehr die bekannte Anordnung der LEDs zueinander zur Identifikation nutzen. Eine Kombination beider Verfahren könnte Vorteile hinsichtlich der Zuverlässigkeit der Positionsbestimmung bieten.

Ein weiterer Grund für die Verwendung der LED-Matrix liegt in dem Ziel, einen Regelkreis zu entwickeln. Hierfür ist die Matrix auf einem programmierbaren Arduino-Board⁸ angebracht. Dabei handelt es sich um ein I/O-Board mit Mikrocontroller und digitalen und analogen Ein- und Ausgängen. Zur Programmierung des Mikrocontrollers steht eine auf *Processing*⁹ basierte plattformunabhängige Integrierte Entwicklungsumgebung zur Verfügung. Die Programmierung des Mikrocontrollers selbst erfolgt in der *Arduino Programming Language*, einer auf *Wiring*¹⁰ basierten Programmiersprache. Somit lassen sich zeitliche Abfolgen von eingeschalteten und ausgeschalteten LEDs erzeugen¹¹. Durch die-

⁸Arduino (2010)

⁹Dabei handelt es sich um einen Java-Dialekt.

¹⁰Dabei handelt es sich um einen C++ ähnlicher C-Dialekt.

¹¹In Odendahl u. a. (2009) findet man Schaltpläne und Quelltexte für die Ansteuerung einer LED-Matrix mit einem Arduino-Board. Die Programme und die Schaltung zur Ansteuerung der LED-Matrix wurden, diesem Buch folgend, von Robert Eichler entwickelt.

se Abfolgen lässt sich eine Bewegung der Hauptkamera gegenüber der kontrollierenden CCD-Kamera simulieren. Die entsprechenden Programme werden per USB-Verbindung auf das Arduino-Board geladen und ausgeführt solange eine Stromversorgung besteht. Eine ständige Verbindung zu einem Rechner wird dazu nicht benötigt.

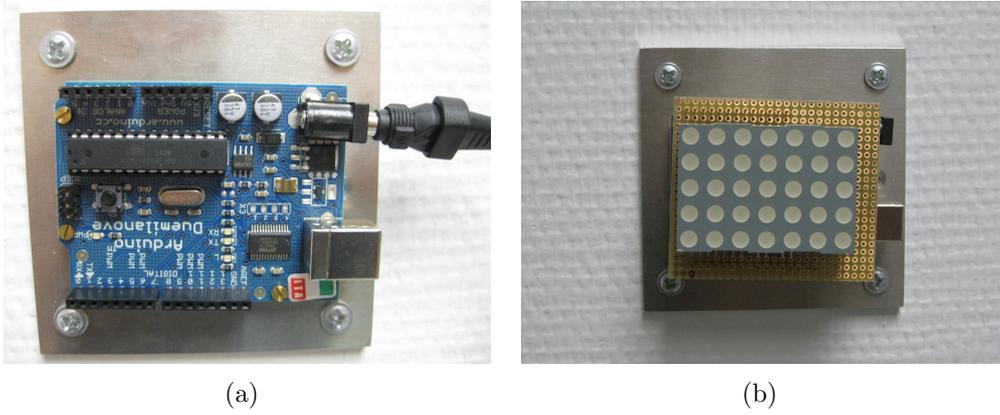


Abbildung 3.10: Wandhalterung mit(a) Arduino-Board und (b) LED-Matrix.

Kapitel 4

Programm zur Bildanalyse

Im Folgenden soll die Struktur und die Funktionsweise des entwickelten Bildanalyseprogramms zur Bestimmung der Position der LED (also der „Hauptkamera“) dargestellt werden. Es wurde die Skript basierte Programmiersprache *Python*¹ verwendet. Unter den verwendeten Modulen besonders hervorzuheben sind *Numpy*², für das Rechnen mit Feldern³, sowie die *Python Imaging Library*⁴, die viele nützliche Funktionen zur Bildbearbeitung zur Verfügung stellt.

4.1 Struktur

Die Bildanalyse besteht aus zwei Komponenten. Die erste Komponente ist das Hauptprogramm (siehe Anhang A.1), welches manuell gestartet wird. Seine Funktion ist damit erschöpft, das Unterprogramm immer dann aufzurufen, wenn dieses nicht ausgeführt wird. Nachdem das Unterprogramm initialisiert worden ist sucht es in seinem Zielordner nach Bildern zur Auswertung. Findet es ein Bild, werden darin die Schwerpunktkoordinaten der LED bestimmt. Anschließend beendet sich das Unterprogramm. Falls der Zielordner leer ist, wird das Unterprogramm sofort wieder beendet. Daraufhin wird es sofort vom Hauptprogramm erneut gestartet. Der Absturz des Unterprogramms wird durch einige Abbruchbedingungen verhindert, die im nächsten Abschnitt erläutert werden.

4.2 Ablauf der Bildanalyse

In diesem Abschnitt soll der Ablauf des Unterprogramms erläutert werden, das auf den Aufnahmen der CCD-Kamera nach einer LED sucht und deren Position bestimmt. Die Erweiterung dieses Programms, die die Verwendung als Echtzeit-Regelung ermöglicht,

¹Dokumentation und Tutorial siehe www.python.org

²Für Dokumentation und Tutorial siehe www.scipy.org

³Felder ermöglichen (wie Listen) den Zugriff auf Daten über einen Index. Obwohl im vorliegenden Programm nur eindimensionale Felder verwendet werden ist ihre Verwendung gegenüber Listen vorzuziehen, da Rechenoperationen mit Feldern in Python deutlich schneller sind als solche mit Listen.

⁴Dokumentation siehe www.pythonware.com

wird später in Kapitel 5.4 vorgestellt. Der Quelltext für das Unterprogramm ist in Anhang A.2 zu finden.

1. Zunächst wird eine Schleife über den Ordner, in dem sich die Aufnahmen befinden, ausgeführt (`os.walk()`). Diese hat das Ziel, das aktuellste Bild zu finden. Hierzu wird genutzt, dass die Aufnahmen beim Abspeichern durch IC.Capture mit einer fortlaufenden Nummer benannt werden können. Die höchste aktuell vorhandene Nummer entspricht also dem neuesten aufgenommenen Bild. Dieses wird für die weitere Analyse geladen (`Image.open()`), während sämtliche anderen im Ordner vorhandenen Dateien gelöscht werden.
2. Wegen der Größe der Bild
3. Zwei Listen werden angelegt. In der ersten ist dem jeweiligen Pixel seine Helligkeit zugeordnet. Die zweite ist ein Histogramm. Aus dem Histogramm wird die mittlere Helligkeit a , sowie die maximale Helligkeit a_{max} bestimmt. Damit wird eine Helligkeitsschwelle $a_{schwelle} = a_{max} + (a_{max} - a)/4$ für die Auswahl von Pixeln festgelegt, die als LED in Frage kommen. Diese Art die Helligkeitsschwelle zu definieren hat den Vorteil, dass sie dynamisch ist und damit in gewissen Grenzen unabhängig von der Umgebungshelligkeit. Natürlich muss gewährleistet sein, dass die LED das hellste Objekt auf der Aufnahme ist.
4. Es wird eine Schleife über die erste Liste gebildet, in der Pixel gefunden werden, deren Helligkeit über dem Schwellwert liegt. Diesen Pixeln werden nun Koordinaten zugeordnet. Um auszuschließen, dass es sich um blinde Pixel handelt, also solche die zufällig hell sind, werden nur solche Pixel als LED in Betracht gezogen, die mehrere helle Nachbarpixel besitzen. Das erste Abbruchkriterium ist, dass kein heller Pixel (oder keiner mit hellen Nachbarn) gefunden wird.
5. Der vorläufige Schwerpunkt der LED wird aus den Koordinaten aller hellen Pixeln, gewichtet mit der Anzahl der ausgelösten Elektronen pro Pixel N_i , berechnet.

$$x = \frac{\sum N_i x_i}{\sum N_i} \quad (4.1)$$

Ein weiteres Abbruchkriterium kommt hinzu: Wenn die Streuung um den Schwerpunkt zu groß ist, muss davon ausgegangen werden, dass die LED nicht zweifelsfrei identifiziert wurde. Die Streuung um den Schwerpunkt kann zum Beispiel dadurch groß werden, dass zwei helle Spots das Kriterium aus 4. für die LED erfüllen. Der errechnete Schwerpunkt läge dann zwischen diesen beiden Spots. Ein zweiter Spot kann zum Beispiel durch Reflexionen an Metallteilen auftreten.

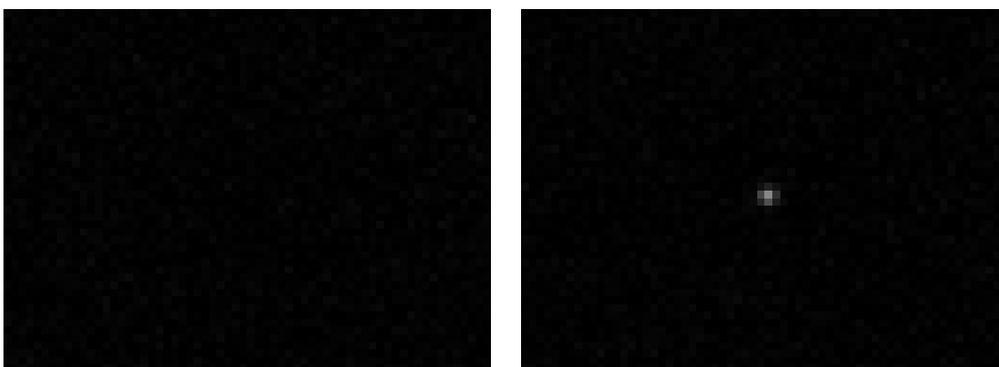
6. Um unabhängiger von zufälligen Helligkeitsschwankungen zu werden, wird nun ein Kasten um die Schwerpunktkoordinaten gezogen und über alle Pixelkoordinaten

darin wie in 5. gewichtet. Damit wird erreicht, dass auch solche Pixel deren Helligkeit zwar unter der Schwelle liegt, die aber dennoch zur LED gehören, mit berücksichtigt werden.

- Abschließend ruft das Programm eine ASCII-Datei auf und fügt dieser eine Zeile mit den Ortskoordinaten der LED in Pixeln im Koordinatensystem der Kamera hinzu, der Nullpunkt entspricht dem Mittelpunkt des Pixels in der linken unteren Ecke des ausgewählten Bereichs der Aufnahme. Die ausgewertete Aufnahme wird nun gelöscht und das Programm beendet.

4.3 Erreichte Genauigkeit der Positionsbestimmung

Im Folgenden soll die erreichte Genauigkeit des Bildanalyseprogramms bei der Bestimmung der LED-Position ermittelt werden. Hierfür wird zunächst ein sogenanntes Darkframe aufgenommen, bei dem das Objektiv mit der dazugehörigen Verschlusskappe abgedeckt ist. Anschließend wird, wie im normalen Betrieb der Bildauswertung, ein Bild der LED aufgenommen.



(a) Darkframe, Aufnahme mit Verschluss vor dem Objektiv. (b) Bild mit unabgedecktem Objektiv und eingeschalteter LED (helles Kreuz in der Bildmitte).

Abbildung 4.1: Bei den Abbildungen handelt es sich um Ausschnitte aus den beiden Aufnahmen, die zur Bestimmung der Unsicherheit der Positionsbestimmung dienen. Beide Ausschnitte sind 64 x 48 Pixel groß. Der Mittelpunkt des Ausschnitts befindet sich jeweils bei $(x,y)=(801 \text{ pixel}, 372 \text{ pixel})$.

Um Fehler durch den Dunkelstrom zu vermeiden wird die Positionsermittlung der LED anhand des Differenzbildes beider Aufnahmen vorgenommen. Die Bestimmung der Position wird wie bereits in Kapitel 4.2 beschrieben durchgeführt. Die Unsicherheit σ_x der ermittelten Position $x = \frac{\sum N_i x_i}{\sum N_i}$ berechnet sich aus dem Fehler der ausgelösten Elektronen N_i mittels Fehlerfortpflanzung. Für die Elektronen wird hier eine Poisson-Verteilung angenommen. Mit

$$\sigma_x^2 = \frac{\sum N_i(x_i - x)^2}{\sum N_i^2} \quad (4.2)$$

erhält man,

$$\sigma_x = 58 \mu m, \sigma_y = 67 \mu m. \quad (4.3)$$

4.4 Langzeitmessung zum Verhalten des Teststandes

Um das Verhalten des Teststands und des Analyseprogramms über einen längeren Zeitraum zu testen, wird eine Messreihe über 70 Stunden aufgenommen. Damit insbesondere das Verhalten bei veränderlicher Helligkeit betrachtet werden kann, ist dabei das Labor nicht völlig abgedunkelt. Die wechselnden Tageszeiten führen also zu unterschiedlichen Helligkeitsverhältnissen im Raum.

In Abbildung 4.2(a) bzw. 4.2(b) erkennt man eine Periodizität der x- und besonders deutlich der y-Komponente der LED-Koordinaten⁵. Die Schwankungen der Position belaufen sich auf bis zu 0,35 Pixel in x-, bzw. 0,20 Pixel in y-Richtung. Die Ursache des periodischen Verhaltens erkennt man in Abbildung 4.2(c). Hier ist die mittlere Helligkeit der Aufnahmen gegen die Zeit aufgetragen. Der Entwicklung der Helligkeit entspricht dem Wechsel von Tag und Nacht. Die Periode der Position folgt der Periode der mittleren Helligkeit der Aufnahmen, die durch die Tageszeit bestimmt ist. Die Übereinstimmung der Perioden von Helligkeit und Position weist darauf hin, dass die Bestimmung der Position tatsächlich von der Umgebungshelligkeit abhängt. Dies erscheint plausibel, da zur Position der LED auch solche Pixel in einem gewissen Bereich um die LED hinzugezogen werden, deren Graustufenwert zwar höher als der Schwellwert ist, die aber dennoch im Vergleich mit dem hellsten Pixel, recht dunkel sind. Der Helligkeitswert dieser Pixeln kann durch einen Anstieg der Umgebungshelligkeit relativ zu der der Pixel im Zentrum durch Reflexion des Umgebungslichts stärker zunehmen, sodass sie mehr Gewicht bei der Bildung des Schwerpunktes erhalten. Im Ergebnis führt das zu einer scheinbaren Änderung der Position der LED. Insgesamt „schwankt“ die Position bei dieser Messreihe um nicht mehr als 0,35 Pixel. Das entspricht einer „Bewegung“ der LED um 0,98 mm.

Eine Quelle für kleinere und kurzfristige Änderungen der bestimmten Position findet sich im Bildrauschen der CCD-Kamera. Dieses ist verantwortlich für die, in Abbildung 4.2(d) erkennbare, breite Streuung der Helligkeit des jeweils hellsten Pixels. Die Begründung, wie dies zu einer scheinbaren Positionsveränderung führen kann, findet sich wiederum in

⁵wenn im folgenden von den LED-Koordinaten die Rede ist, so sind immer die Koordinaten in Pixeln im Koordinatensystem der Kamera gemeint

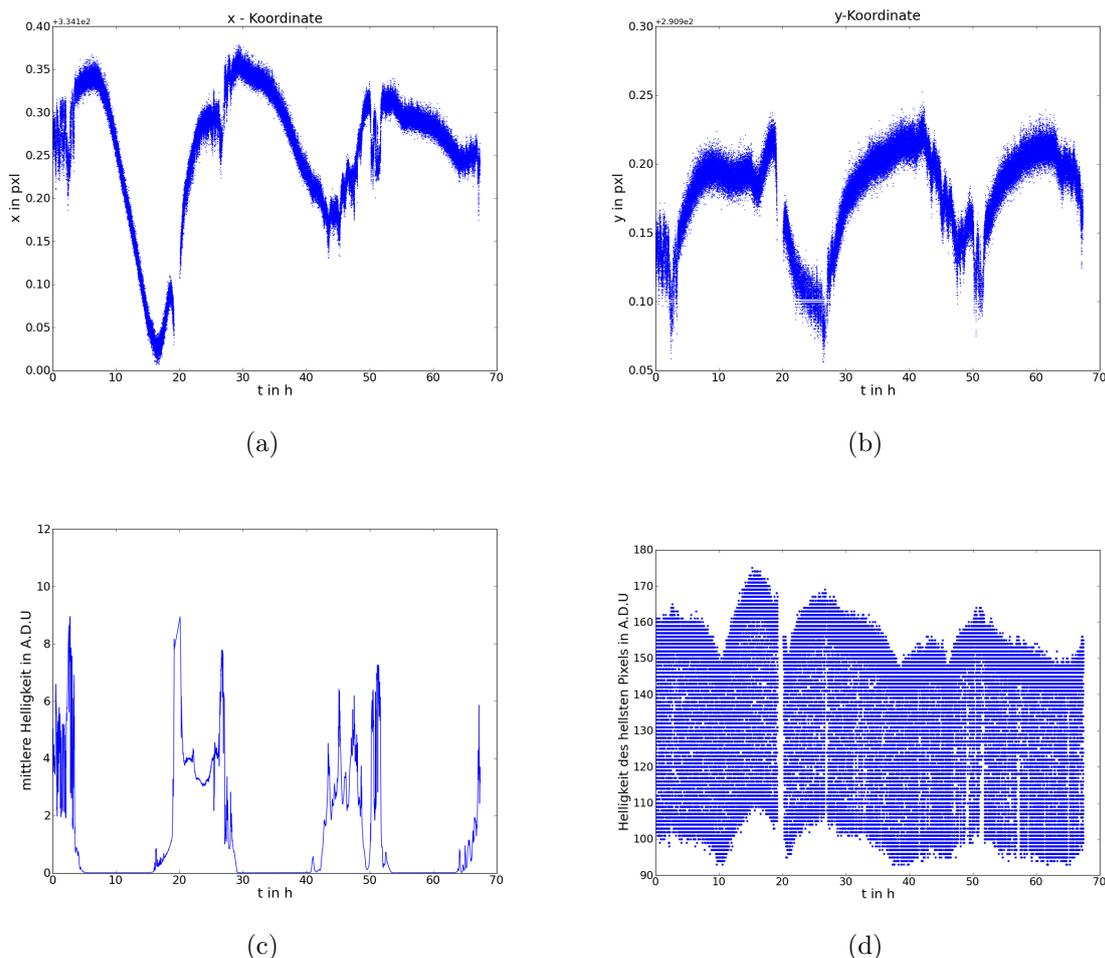


Abbildung 4.2: Die Abbildungen (a) und (b) zeigen die Entwicklung der x-, bzw. y-Koordinate der LED in einem Zeitraum von ca. 67 Stunden. Auffällig ist die Periodizität im Tagesrhythmus. (c) Um einen vermuteten Zusammenhang zwischen der bestimmten Position der LED und der Umgebungshelligkeit zu überprüfen, wurde gleichzeitig die mittlere Helligkeit der Aufnahmen gemessen. (d) Eine weitere Beeinflussung der gemessenen LED-Position kann das Rauschen des CCD-Chips hervorrufen. Daher wurden auch das Verhalten der Helligkeit des jeweils hellsten Pixels im selben Zeitraum betrachtet.

der Gewichtung der Position der einzelnen Pixel mit ihrer Helligkeit.

Auffällig bei der Messreihe ist, dass um die zwanzigste Stunde herum die Auswertung für eine Stunde unterbrocht⁶, dann jedoch ohne Probleme fortgesetzt wird. Wahrscheinlich ist an dieser Stelle eine Abbruchbedingung erfüllt. Zum Beispiel erscheint es gut möglich, dass einfallendes Sonnenlicht an einer spiegelnden Oberfläche einen Reflex erzeugt, der durch das Analyseprogramm nicht von einer LED unterschieden werden kann. Für diesen Fall ist vorgesehen, dass das Unterprogramm beendet wird, ohne eine Position zu speichern. Nach einiger Zeit würde der Reflex dann verschwinden und das Programm liefere weiter, wie im vorliegenden Fall geschehen.

⁶Die Auswertung ist von 10:25:42 bis um 11:20:44 unterbrochen

Kapitel 5

Spiegelsteuerung

5.1 Berechnung der Transformationsmatrix

Um die Spiegelmotoren anhand der Bilddaten zu steuern, ist es notwendig den Zusammenhang zwischen einer Bewegung der Motoren und der entsprechenden Positionsänderung eines definierten Punktes (LED) auf den Bildern der mitbewegten Kamera zu kennen. Beide Größen, also die Motorposition a^1 , angegeben in Hallcounts, und die Position x^2 der LED im Koordinatensystem der Kamera, sind über eine Transformationsmatrix \hat{T} miteinander verknüpft, die nach der in Kapitel 3.1.3 [Cornils \(2006\)](#) beschriebenen Methode berechnet wird .

$$\Delta x = \hat{T} \Delta a, \quad \Delta a = \hat{T}^{-1} \Delta x \quad (5.1)$$

$$\hat{T} = \begin{pmatrix} \partial x_1 / \partial a_1 & \partial x_1 / \partial a_2 \\ \partial x_2 / \partial a_1 & \partial x_2 / \partial a_2 \end{pmatrix} \quad (5.2)$$

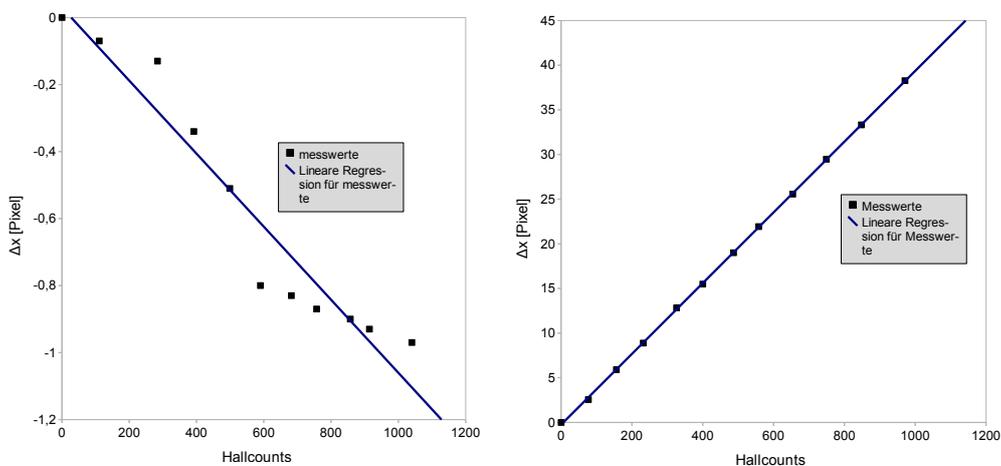
Um \hat{T} zu bestimmen wird mehrmals hintereinander ein Motor bewegt und die gefahrenen Hallcounts notiert. Nach jeder Bewegung des Motors wird mit der CCD-Kamera eine Aufnahme gemacht und mit dem Unterprogramm die resultierende Positionsänderung der LED bestimmt. [Abbildung 5.1](#) zeigt die Messwerte für beide Motoren. Die Spalteneinträge der Matrix werden mit linearer Regression aus den Messwerten bestimmt. Wie zu sehen ist, zeigt die erste Komponente von \hat{T} deutliche Abweichungen vom linearen Verlauf. Der Grund hierfür ist nach [Cornils \(2006\)](#), dass sich die Komponenten von \hat{T} nur in erster Näherung linear in a verhalten und tatsächlich \hat{T} selbst eine Funktionen

¹Bei a handelt es sich um ein Tupel (a_1, a_2) . Die Einträge beschreiben die Position des Motors 0 bzw. des Motors 1 in Hallcounts.

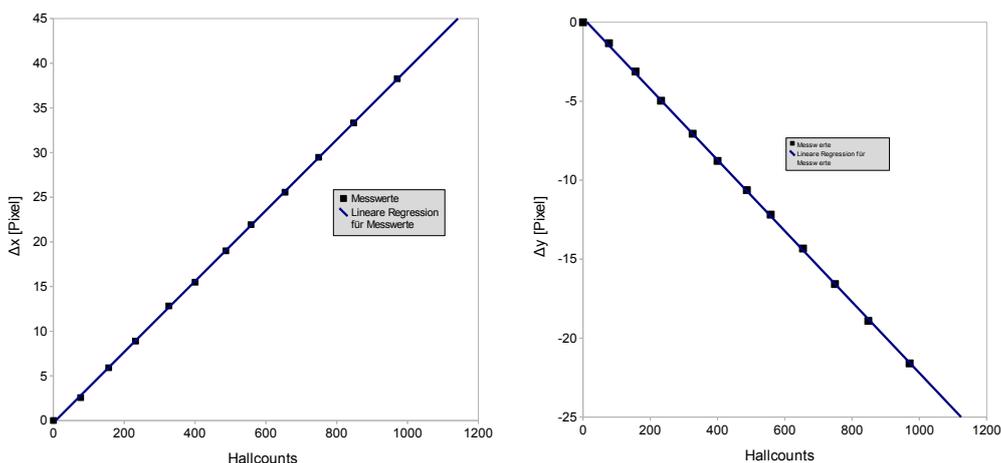
²Das Tupel $x=(x_1, x_2)$ beschreibt die x,y-Koordinate der LED-Position auf den Aufnahmen der CCD-Kamera in der Einheit Pixel.

der Aktuatorposition a ist. Um über große Aktuatorwege eine hohe Genauigkeit zu gewährleisten wäre deshalb ein iteratives Vorgehen nötig. Darauf wird an dieser Stelle, mit möglichen Einbußen in der Genauigkeit, verzichtet. Mit der linearen Näherung erhält man die Transformationsmatrix:

$$\hat{T} = \begin{pmatrix} -0,001 & 0,040 \\ 0,045 & 0,023 \end{pmatrix} \quad (5.3)$$



(a) Messreihe zur Bestimmung von $\partial x_1/\partial a_1$ mit linearem Fit. (b) Messreihe zur Bestimmung von $\partial x_1/\partial a_1$ mit linearem Fit.



(c) Messreihe zur Bestimmung von $\partial x_1/\partial a_2$ mit linearem Fit. (d) Messreihe zur Bestimmung von $\partial x_2/\partial a_2$ mit linearem Fit.

Abbildung 5.1: Die Abbildung zeigt die Messwerte zur Bestimmung der Elemente der Transformationsmatrix jeweils mit linearem Fit.

Zur Überprüfung der Transformationsmatrix kann man ausnutzen, dass die Spaltenvektoren nicht unabhängig voneinander sind. Vielmehr gilt:

$$\begin{pmatrix} \partial x_1 / \partial a_2 \\ \partial x_2 / \partial a_2 \end{pmatrix} = R_{120} \begin{pmatrix} \partial x_1 / \partial a_1 \\ \partial x_2 / \partial a_1 \end{pmatrix} = \begin{pmatrix} -1/2 & \sqrt{3}/2 \\ \sqrt{3}/2 & -1/2 \end{pmatrix} \begin{pmatrix} \partial x_1 / \partial a_1 \\ \partial x_2 / \partial a_1 \end{pmatrix} \quad (5.4)$$

Die Drehmatrix R_{120} gibt dabei den Winkel von ungefähr 120° zwischen den beiden Bewegungsrichtungen eines Punktes in der Fokalebene wieder und geht auf die geometrische Anordnung der Haltepunkte des Spiegels zurück. Zur Ansteuerung der Motoren mit den Bilddaten wird die Inverse Matrix der Transformationsmatrix \hat{T} benötigt. Diese berechnet sich wie folgt:

$$\hat{T}^{-1} = \frac{\text{adj}(\hat{T})}{\det(\hat{T})} = \begin{pmatrix} 17,72 & 22,54 \\ 26,01 & 0,58 \end{pmatrix} \quad (5.5)$$

5.1.1 Winkelauflösung der Motorsteuerung

Die kleinsten Verschiebungen, die korrigiert werden können, sind durch die Schrittweite der Spiegelmotoren, die einem Hallcount entspricht, begrenzt. Die Winkelauflösung der Motorsteuerung lässt sich aus der Positionsänderung pro Hallcount berechnen. Diese entspricht den Spaltenvektoren in der Transformationsmatrix \hat{T} . Aus dem Betrag der Verschiebungsvektoren $\Delta r = \sqrt{x^2 + y^2}$ lässt sich eine Verschiebung der LED in Metern berechnen. Für die Winkelauflösung gilt $\alpha_i = 2 \arctan\left(\frac{\Delta r}{2d}\right)$, dabei ist d der Abstand zwischen der LED-Matrix und der CCD-Kamera. Die Rechnung zeigt, dass für beide Motoren die Winkelauflösung in etwa gleich ist: $\alpha_{0,1} = 12,9 \pm 0,4 \mu\text{rad}$.

5.2 Ansteuerung der Spiegelmotoren und Test der Transformationsmatrix

Zur Ansteuerung der Spiegelmotoren wird ein Programm verwendet, das eine Liste aus anzusteuern den Koordinaten erstellt. Mit Hilfe der Transformationsmatrix wird diese in eine Liste aus Hallcounts für beide Motoren umgerechnet. Über die letztere Liste wird eine Schleife gebildet, sodass für jede Koordinate einmal beide Motoren angesteuert werden. Dies geschieht mit einem weiteren Programm, welches im Wesentlichen der H.E.S.S-Steuerungssoftware entnommen wurde. Auf die hierfür benötigte Zeit soll weiter unten noch eingegangen werden. Der Test der Transformationsmatrix besteht nun einfach darin, eine Bahn zu definieren, in diesem Fall ein Kreis, und zu überprüfen, ob diese Bahn der tatsächlichen Bewegung der Kamera entspricht. Wie man in Abbildung 5.2 erkennt ist

dies der Fall.

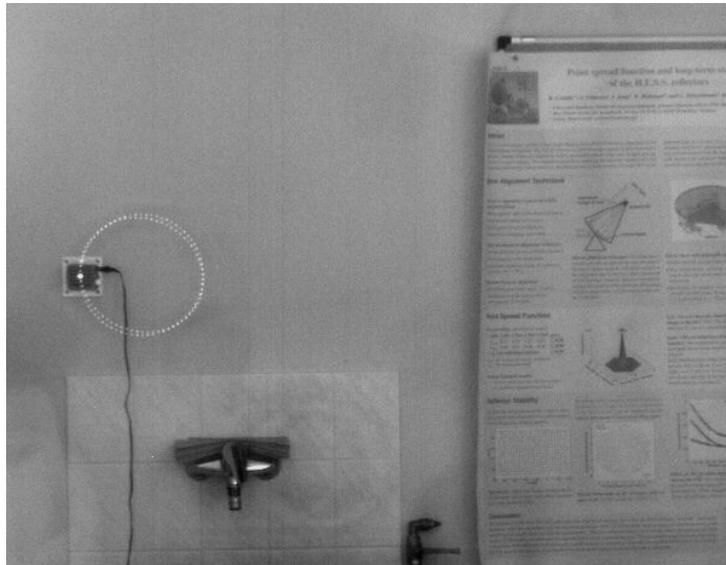


Abbildung 5.2: Zum Test der Transformationsmatrix und des Motorverhaltens wird ein Programm verwendet, das Koordinaten für einen Kreis liefert, die dann mit der Transformationsmatrix in Befehle zur Motorsteuerung umgerechnet werden. Die abgebildete Aufnahme entstand durch die Addition von 600 Einzelaufnahmen, die während der Bewegung des Spiegels mit einer Frequenz von 7,5 Bildern pro Sekunde aufgenommen wurden.

5.3 Benötigte Zeit für die Ansteuerung der Motoren

Die für die Ansteuerung und Bewegung der Motoren benötigte Zeit ist für die Verwendung des Aufbaus als Regelkreis von kritischer Bedeutung. Der Zeitaufwand für die Nachführung bestimmt die Frequenz der kompensierbaren Störungen. Die mit Hilfe einer Stoppuhr ermittelte Zeit für die einmalige Ansteuerung eines Motors beträgt 11 s. Da um eine bestimmte Koordinate (x,y) im Koordinatensystem der Kamera anzufahren im allgemeinen beide Motoren angesteuert werden müssen, ergibt sich eine Gesamtzeit von 22 s. Der Ablauf besteht jedes mal aus dem Initialisieren der Anwendung zur Motorsteuerung, der Bewegung des Motors (auf einen Hallcount genau) und dem Beenden der Anwendung. Die Intervalle für die einzelnen Schritte wird ebenfalls bestimmt. Dabei fällt auf, dass die Zeit zum Initialisieren und Beenden der Anwendung jeweils ca. 4 bis 5 s, die benötigte Zeit zum tatsächlichen Fahren des Motors (bei 100 Hallcounts) jedoch nur ca. 1 bis 2 s beträgt. Zumindest die Zeit zum Initialisieren und Beenden sollte sich durch eine Optimierung der Steuersoftware noch verringern lassen.

5.4 Einrichtung und Test eines Regelkreises

Ein erster Versuch einen Regelkreis zur Steuerung der Aktuatoren einzurichten, soll hier vorgestellt werden. Dazu wird im Programm zur Positionsbestimmung ein Abschnitt eingefügt, in dem die aktuelle Position des Spiegels mit dem Sollwert verglichen wird. Als Sollwert dient die erste in einer Ascii-Datei gespeicherte Position der LED. Die Position der LED auf der aktuellen Aufnahme wird mit diesem Paar (x,y) verglichen. Falls die festgestellte Differenz einen Wert von 0,5 Pixeln (entsprechend einer Positionsänderung der LED um 1,4 mm) überschreitet, wird mit der, in Kapitel 5.1 berechneten, Transformationsmatrix ein Tupel aus Hallcounts für die beiden Motoren berechnet und an MACS übergeben, sodass der Spiegel eine Bewegung zur Kompensation der Abweichung ausführt. Der Wert von 0,5 Pixeln wurde relativ willkürlich gewählt und soll gewährleisten, dass nur eine tatsächliche „Bewegung“ der beleuchteten LED auf der LED-Matrix kompensiert wird und nicht etwa zufällige Schwankungen der ermittelten Position. Der Abstand zwischen zwei LED's auf der Diagonalen der Matrix beträgt 11 mm, entsprechend 3,9 Pixelbreiten auf den Aufnahmen der CCD-Kamera.

Um die Funktion des Regelkreises zu prüfen, wird ein kleines Programm zur Ansteuerung der LED-Matrix verwendet. Dieses Programm lässt in vorher festgelegten Intervallen die LEDs entlang der Diagonalen der Matrix aufleuchten.

Bei einer ersten Testmessung hat sich gezeigt, dass es wichtig ist, dass der zeitliche Abstand der Positionsänderungen kleiner ist, als die benötigte Zeit für die Nachführung. Da im Allgemeinen beide Motoren gefahren werden müssen, beträgt die zur Nachführung mindestens benötigte Zeit 22 s. Die Ansteuerung eines einzelnen Motors wird im Folgenden stets als „Schritt“ bezeichnet. In den meisten Fällen ist der Spiegel nach den ersten beiden Schritten zu weit gefahren. Falls die Abweichung gegenüber der Sollposition den kritischen Wert von 0,5 Pixeln noch immer überschreitet, wird mindestens ein weiterer Schritt zum genauen Anfahren der Sollposition nötig. Die Untergrenze für die Dauer jeder Nachregelung liegt damit bei ca. 40 Sekunden. Einer schnelleren Positionsänderungen kann die Nachführung also nicht folgen.

Um zu verhindern, dass die Motorsteuerung auch initialisiert wird, wenn keine oder nur wenige Hallcounts gefahren werden sollen, wurde eine Schwelle von 3 Hallcounts festgelegt unterhalb der das Steuerungsprogramm nicht aufgerufen wird. Dadurch verringert sich die Zeit pro Nachführung um etwa 11 s.

In Abbildung 5.3 erkennt man die Änderung der x, bzw. der y-Komponente der LED-Position, aufgetragen gegen die Zeit. Die jeweils beleuchtete LED wechselt alle 50 Sekunden entlang der Diagonalen der Matrix. Das mehrmalige Durchlaufen der Matrix führt zu dem beobachteten periodischen Verlauf. Die Spitzen treten auf, wenn die LED soeben die Position geändert hat. Zum erneuten Einstellen der Sollposition (hier: $x = 219,40$ Pixel, $y = 193,50$ Pixel) sind meist drei Schritte nötig. Diese Testmessung hat gezeigt, dass sich Positionsänderungen der LED mit einer kleineren Frequenz als 0,02 Hz mit dem

entwickelten Regelkreis nachregeln lassen. Dieses Ergebnis stimmt optimistisch für einen Einsatz bei Teleskopen wie denen des CTA. Zwar wird damit gerechnet, dass die Eigenfrequenz der Konstruktion im Bereich von einigen Hertz liegt und damit die resultierenden Störungen der Position, zumindest mit der hier entwickelten Regelung, nicht nachgeführt werden können. Die von der Elevation des Teleskops abhängigen Störungen der Position dürften sich jedoch im Zeitraum von einer halben Stunde bis mehreren Stunden abspielen und wären damit bereits nachregelbar³.

³Die Einschätzungen stammen aus einem Gespräch mit Dieter Horns

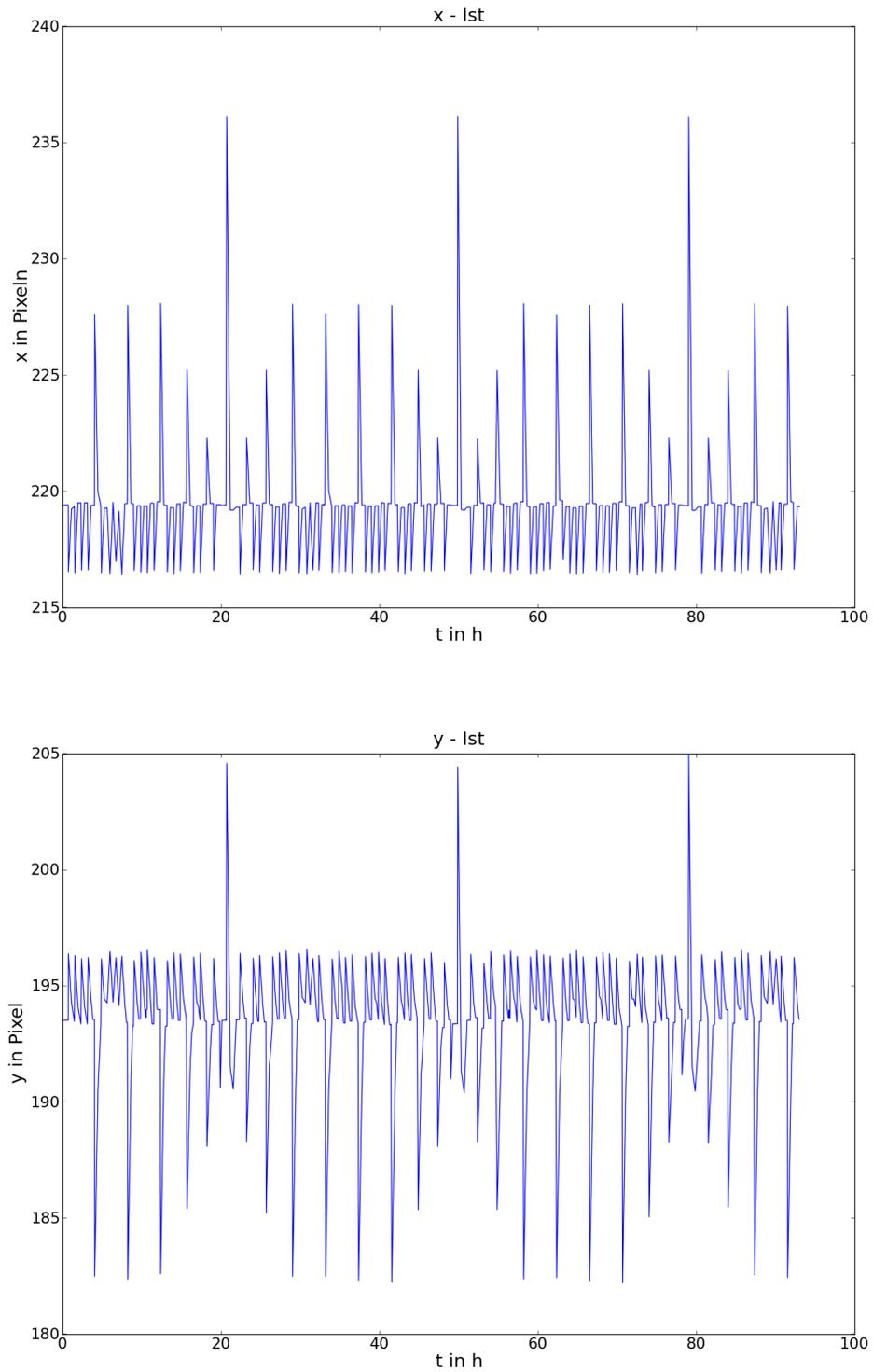


Abbildung 5.3: Veränderung der (a) x- und (b) y-Komponente der Position der LED beim Test des Regelkreises.

Kapitel 6

Zusammenfassung und Ausblick

In der vorliegenden Arbeit wurden umfangreiche Vorarbeiten für die Entwicklung einer Echtzeit-Regelung des Spiegelsystems einer neuen Generation von Cherenkov-Teleskopen geleistet. Zunächst wurde für den Test eines solchen Regelkreises ein Aufbau errichtet, bei dem eine Spiegelfacetten des geplanten Teleskops durch einen Prototypen, der ursprünglich für das Spiegelsystem der H.E.S.S.-Teleskope gebaut wurde, vertreten wird [Kapitel 3.2]. Die Hauptkamera, auf die jede Spiegelfacetten ausgerichtet sein wird, wurde mit Hilfe einer LED-Matrix simuliert. An dem beweglichen Spiegelprototypen wurde eine CCD-Kamera angebracht, sodass eine optische Kontrolle der Positionsänderungen von Spiegel oder LED möglich ist. Für die Analyse aufgenommenen Bilder der LED-Matrix wurde ein Programm geschrieben, das die LED anhand ihrer Helligkeit identifiziert und ihre Position bestimmt [Kapitel 4.2]. Bei einer Langzeitmessung zum Verhalten des Teststandes und des Analyseprogramms wurde festgestellt, dass die Umgebungshelligkeit einen Einfluss auf die bestimmte Position hat, sodass bei größeren Helligkeitsschwankungen die Größenordnung der erkannten Verschiebungen der LED auf ca. 1 mm beschränkt ist. Bei konstanter Helligkeit der Umgebung wurde eine weitaus bessere Empfindlichkeit von ca. 0,13 mm ermittelt. Dies bietet gute Aussichten für den Betrieb einer Echtzeit-Regelung [Kapitel 4.4].

Neben der Analyse-Software war die Steuerung des Spiegelprototypen anhand von Kameradaten ein wichtiger Aspekt dieser Arbeit und ist ebenfalls Voraussetzung für den Betrieb eines Systems zur Echtzeit-Regelung der Nachjustierung. Dafür wurde zunächst eine Transformationsmatrix berechnet, die den Zusammenhang zwischen Bewegungen der beiden Aktuatoren des Spiegelprototypen und Positionsänderungen der LED auf den Aufnahmen der CCD-Kamera herstellt. Die Transformationsmatrix wurde anschließend getestet, indem Bahnen in Koordinaten der CCD-Kamera vorgegeben und in Befehle zur Steuerung der Spiegelmotoren umgesetzt wurden. Die Bahn des Spiegels wurde überprüft, indem viele Einzelaufnahmen der LED addiert wurden. Für die Motorsteuerung selbst wurde dabei auf das Mirror Alignment Control System von H.E.S.S. zurückgegriffen [Kapitel 5.4].

Durch die Kombination des Programms zur Bildanalyse mit der Motorsteuerung wurde zum Abschluss der Arbeit ein Regelkreis eingerichtet. Dieser wurde erfolgreich verwendet um Positionsänderungen mit einer Frequenz von 0,02 Hz zu kompensieren [Kapitel 5.4].

Ausblick

Die Entwicklung eines Regelkreises für Spiegelsysteme von Cherenkov-Teleskopen liefert einen Beitrag für die Automatisierung des Teleskopbetriebs, wie sie bei CTA geplant ist. In einer weiteren Arbeit könnte daher der entwickelte Regelkreis verfeinert und weiter untersucht werden, um eine zuverlässige Echtzeit-Regelung zu ermöglichen, die die für CTA erforderliche Genauigkeit besitzt. So würde eine Reduzierung der für die Initialisierung der Motorsteuerung benötigten Zeit den Ausgleich höherfrequenter Störungen ermöglichen.

Die in dieser Arbeit verwendete Kamera hatte einen Preis von 490 Euro. Hinzu kommt noch das 16mm-Objektiv mit einem Preis von 160 Euro. Im Hinblick auf eine Reduzierung der Kosten ist es deshalb interessant, dass die Genauigkeit der hier verwendeten Kamera diejenige der Spiegelmotoren übersteigt. Deshalb liegt es nahe zu Prüfen, ob auch günstigere Kameras die erforderliche Genauigkeit für die Anwendung in einem Regelkreis wie dem vorgestellten besitzen. Konsequenterweise (die Kosten betreffend) kann man hierfür auch Webcams in Betracht ziehen. Da die benötigte Brennweite des Objektivs aufgrund der geringeren Pixelgröße von Webcams kleiner ist, sinkt gleichzeitig der Preis für das Objektiv, wodurch das Kamerasystem noch günstiger wird.

Man darf gespannt sein, wie weit die Automatisierung großer Teleskopsysteme in Zukunft gehen wird.

Anhang A

Quelltext des Programms zur Bildanalyse

A.1 Quelltext des Hauptprogramms

```
import os

while (True):
    os.system("python_echtzeit_untergeordnet.py")

    if os.path.isfile("stop"):
        break

os.remove("stop")
```

A.2 Quelltext des Unterprogramms

```
import math
import pylab
import Image
import numpy
import os
import glob
import matplotlib
import sys
from numpy import *
from datetime import datetime

ylist = []
xlist = []
zlist = []
rootdir = '/home/spiegel/samba/camera'
rechteck = 500, 250, 900, 700
savedir = '/home/spiegel/bilder'

####schleife ueber alle bilder
maximum = 0
bildpfad = ""
t = datetime.now()
for root, dirs, files in os.walk(rootdir):

    for name in files:
        o = os.path.splitext(name)
        bildnr = int(o[0]) - 2
        if bildnr > maximum:
            maximum = bildnr
            bildpfad = os.path.join(root, name)
```

```

for root, dirs, files in os.walk(rootdir):

    for name in files:
        o = os.path.splitext(name)
        bildnr = int(o[0])-2
        if bildnr < maximum:
            schlecht = os.path.join(root,name)
            os.remove(schlecht)

if bildpfad == "":
    sys.exit()

box = Image.open(bildpfad)
im = box.crop(rechteck)      #Verkleinerung des Bildes
sizeX,sizeY = im.size
pixel = im.getdata()
punkte = list(im.getdata())
hist = im.histogram()
histogram = array(hist)

nl = []
for i, n in enumerate(hist):
    b = i*n
    nl.append(b)

a = sum(nl)/float(sum(histogram))      #mittlere Helligkeit
print a

a_max = max(punkte)                  #maximale Helligkeit
a_schwelle = a + abs(a_max - a)/4     #dynamische Helligkeitsschwelle

helle_punkte = []
class punkt:
    def __init__(self,x,y,L):
        self.x = x
        self.y = y
        self.L = L
        self.gut = False

for i,L in enumerate(pixel):
    if L > a_schwelle:
        zeile = i/sizeX
        spalte = i%sizeX
        helle_punkte.append(punkt(spalte, zeile, L))

if len(helle_punkte) > 20:
    os.remove(bildpfad)
    sys.exit()

for l1 in helle_punkte: #liefert Anzahl der Nachbarpunkte
    N = 0 # N = Nachbarn des Pixels
    for l2 in helle_punkte:
        g = ((l2.x - l1.x)**2 + (l2.y - l1.y)**2)**(0.5)
        if g < 5:
            N = N + 1
    if N > 1 and N < 25:
        l1.gut = True

xwerte = []
ywerte = []
helligkeit = []

for l in helle_punkte:
    if l.gut == True:
        e = l.L * 10.03      # Anzahl Photoelektronen
        xwerte.append(l.x)
        ywerte.append(l.y)
        helligkeit.append(e)
#####
n = float(len(xwerte))

if n > 15:
    os.remove(bildpfad)
    sys.exit()

if n > 0:
    x_array = array(xwerte)
    y_array = array(ywerte)
    g = array(helligkeit)
    gx = multiply(x_array,g)
    gy = multiply(y_array,g)
    x = sum(gx)/float(sum(g))

```

```

y = sum(gy)/float(sum(g)) #Schwerpunkt des hellen Punktes

xabw = []
yabw = []
for xi in xwerte:
    s = (xi - x) ** 2
    xabw.append(s)
for yi in ywerte:
    d = (yi - y) ** 2
    yabw.append(d)

u_x = (sum(xabw)/(n-1))**0.5
u_y = (sum(yabw)/(n-1))**0.5
std = u_x + u_y

#####

if std > 2:
    os.remove(bildpfad)
    sys.exit()

kasten = int(x) - 2, int(y) - 2, int(x) + 2, int(y) + 2
cutout = im.crop(kasten)
size_x, size_y = cutout.size
pixel = cutout.getdata()
punkte = list(cutout.getdata())

gewichtung = []
x_list = []
y_list = []
for i,L in enumerate(pixel):
    zeile = i/size_x + int(y) - 2
    spalte = i%size_x + int(x) - 2
    h = (L*10.04-a*10.04)
    gewichtung.append(h)
    x_list.append(spalte)
    y_list.append(zeile)

weight = array(gewichtung)
x_array_new = array(x_list)
y_array_new = array(y_list)

w_x = multiply(x_array_new,weight)
w_y = multiply(y_array_new,weight)

x_new = sum(w_x)/float(sum(weight))
y_new = sum(w_y)/float(sum(weight))

ist_datei = open("istwert.asc", 'a')

xwert = str(x_new)
ywert = str(y_new)
twert = str(t)
awert = str(a)
amax = str(a_max)
zeile_ist = xwert + "    " + ywert + "    " + twert + "    " + awert + "    " + amax
#####"REGELKREIS"#####

x_datei = []
y_datei = []
for zeile in ist_datei:
    zeile = zeile[:-1].split()
    x_datei.append(float(zeile[0]))
    y_datei.append(float(zeile[1]))

i = len(x_datei)
if i > 0:
    delta_x = x_new - x_datei[0]
    delta_y = y_new - y_datei[0]

    r = (delta_x**2 + delta_y**2)**0.5

    if abs(delta_x) <= 0.5 and abs(delta_y) <= 0.5:

        soll_datei.write(zeile_ist + "\n")
        ist_datei.write(zeile_ist + "\n")
    else:
        a_0 = -(17.72*delta_x + 22.54*delta_y)
        a_1 = -(26.01*delta_x + 0.58*delta_y)

        if abs(a_0) > 3:

```

```
        os.system('/home/spiegel/macs/driving/obj/macsDriveMotor_Motor=0_Interval='+str(int(a_0
        )))
    if abs(a_1) > 3:
        os.system('/home/spiegel/macs/driving/obj/macsDriveMotor_Motor=1_Interval='+str(int(a_1
        )))

    ist_datei.write(zeile_ist + "\n")

else:
    ist_datei.write(zeile_ist + "\n")
#####

print "mittlere_Helligkeit:", a, "n:", n , "x:", x_new, "y:", y_new
os.remove(bildpfad)
```

Literaturverzeichnis

- [Apogee 2010] APOGEE: *Webseiten von Apogee Instruments Inc.* 2010. – URL www.ccd.com
- [Arduino 2010] ARDUINO: *offizielle Webpräsenz des Arduino-Projekts.* 2010. – URL arduino.cc
- [Biland u. a. 2008] BILAND, A. ; GARCZARCZYK, M. ; ANDERHUB, H. ; DANIELYAN, V. ; HAKOBYAN, D. ; LORENZ, E. ; MIRZOYAN, R.: *The Active Mirror Control of the MAGIC Telescope.* (2008)
- [Cornils 2001] CORNILS, Rene: *Untersuchungen zur automatisierten Justierung der Spiegelfacetten der H.E.S.S. Cherenkov-Teleskope*, Institut für Experimentalphysik, Universität Hamburg, Diplomarbeit, 2001
- [Cornils 2006] CORNILS, Rene: *Justierung und Abbildungsfunktion der H.E.S.S.-Reflektoren sowie Untersuchung der ultraleuchtkräftigen Infrarot-Galaxie Arp 220 mit dem H.E.S.S.-Teleskopsystem*, Institut für Experimentalphysik, Universität Hamburg, Dissertation, 2006
- [Dreyling-Eschweiler 2010] DREYLING-ESCHWEILER, Jan: *Aufbau und Test eines neuartigen CCD-Kamerasystems zur Kontrolle der Optik eines H.E.S.S.-Cherenkovteleskops*, Institut für Experimentalphysik, Universität Hamburg, Diplomarbeit, 2010
- [H.E.S.S.-Kollaboration 2010] H.E.S.S.-KOLLABORATION: *Offizielle Webseiten von H.E.S.S.* 2010. – URL www.mpi-hd.de/hfm/HESS
- [Horns 2010] HORNS, Dieter: *Private Kommunikation.* 2010
- [Odendahl u. a. 2009] ODENDAHL, Manuel ; FINN, Julian ; WENGER, Alex: *Arduino-Physical Computing für Bastler, Designer und Geeks.* O'Reilly, 2009
- [Python 2010] PYTHON: *Webseiten des Python-Projekts.* 2010. – URL www.python.org
- [Pythonware 2010] PYTHONWARE: *Webseiten der Python Imaging Library.* 2010. – URL www.pythonware.com

[scipy 2010] SCIPY: *Webseiten des Scipy-Projekts*. 2010. – URL www.scipy.org

[TheImagingSource 2010] THEIMAGINGSOURCE: *offizielle Webpräsenz des Kameraherstellers The Imaging Source Europe GmbH*. 2010. – URL ww.theimagingsource.com

[Wagner u. a. 2009] WAGNER, R. M. ; LINDFORS, E. ; SILLANPÄÄ, A. ; WAGNER, S.: *The CTA Observatory*. (2009)

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine weiteren Quellen und Hilfsmittel als die angegebenen verwendet habe. Mit einer universitätsinternen Veröffentlichung bin ich einverstanden.

Hamburg, den

Jan H. Drewes