



DIPLOMARBEIT

Titel der Diplomarbeit

Feasibility Study of Parallaxes with CoRoT

angestrebter akademischer Grad

Magister der Naturwissenschaften (Mag. rer. nat.)

Verfasser:	Gabriel Stöckle
Matrikel-Nummer:	a0047233
Studienrichtung:	Astronomie A413
Betreuer:	Univ.-Prof Dr. Maria G. Firneis

Wien, 6. November 2009

Erklärung

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst habe, andere als die angegebenen Quellen nicht benutzt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Wien, am 6. November

2009

Abstract

Ziel dieser Arbeit war es, die Messbarkeit von trigonometrischen Parallaxen mit dem Weltraumteleskop CoRoT zu untersuchen.

Gegeben waren Simulationen der Punktverteilungsfunktion (PSF) der CoRoT-Optik für je 4 verschiedene Sterntemperaturen und für 4 Positionen in der optischen Ebene. Zunächst wurde die Qualität verschiedener ESO-MIDAS-Algorithmen untersucht. Dies geschah indem man die Veränderungen der 16 PSFs modellierte und diese Modelle auf Nachvollziehbarkeit untersuchte. Dabei zeigte sich, dass die Center/Gauss Methode sehr ungenaue Positionsergebnisse lieferte. Zwischen den beiden MIDAS Algorithmen Center/Moment und Center/IQE konnte kein signifikanter Qualitätsunterschied festgestellt werden. Erste Abschätzungen für die erreichbare Messgenauigkeit der Position wurden gemacht, mit dem Ergebnis, dass eine Genauigkeit von der Größenordnung einer Millibogensekunde mit der gegebenen PSF und den verwendeten Positionsalgorithmen numerisch erreichbar wäre.

Desweiteren wurden eigene Algorithmen zur Positionsdefinition einer PSF entwickelt und getestet. Es handelt sich dabei um einen Schwerpunktsalgorithmus, ähnlich dem Center/Moment Algorithmus, um eine Symmetrie-Algorithmus, der die Symmetrien der PSF zur Positionsdefinition nutzt, sowie den Versuch die PSF durch ein Dreieck zu fitten.

Diese Untersuchungen hatten das Ergebnis, dass der Schwerpunktsalgorithmus der numerisch stabilste und auch derjenige Algorithmus ist, der die meiste Information der PSF verarbeiten kann. Der Symmetriealgorithmus und der Dreiecksfit scheiterten vor allem an der ungewöhnlichen Topologie der PSF. Die Anzahl der verwendbaren Datenpunkte war einfach zu gering um die Vorteile der Algorithmen zur Geltung zu bringen.

Nachdem wir vom CoRoT-Team namentlich Michel Auvergne Aufnahmen von CoRoT-Feldern erhalten hatten, wurden diese auf ihre Auswertbarkeit für astrometrische Zwecke untersucht. Die Aufnahmen enthielten unter anderem zwei Aufnahmen des gleichen Feldes zu verschiedenen Zeitpunkten. Zunächst wurden von Hand 5 Sterne in den beiden Aufnahmen dieses Feldes mit dem Schwerpunktsalgorithmus vermessen. Der Vergleich der gemessenen Positionen ergab, dass die Positionen eine Schwankungsbreite von ca. 0.1 Pixeln besitzen, dies entspricht grössenordnungsmässig einer Positionsgenauigkeit von einer Bogensekunde. Diese Messung wurde automatisiert, dazu wurden automatische Sternerkennungsalgorithmen geschrieben. Die automatisierte Auswertung mit 20 Sternen ergab eine Genauigkeit von ca. einer halben Bogensekunde. Eine direkte Korrelation der beiden Felder ergab, dass aufgrund des Hintergrundrauschens eine höhere Genauigkeit als 0.1 Pixel also eine viertel Bogensekunde nicht möglich ist. Kurz wurde die Frage geklärt ob eine höhere Anzahl der Messungen des gleichen Sternfeldes zwangsläufig die Genauigkeit der gemessenen Positionen erhöht, was nicht zwangsläufig der Fall ist. Wichtig für einen Genauigkeitserfolg ist das Vorhandensein von Referenzpunkten, ohne die würde eine erneute Messung keinen Informationsgewinn bedeuten. Der Genauigkeitserfolg hängt von der Genauigkeit der Einordnung in ein Referenzsystem, also auch von der Messung der Referenzpunkte und deren Genauigkeit ab. Weiter enthält

die Arbeit eine Beschreibung der geometrischen Effekte der Projektion und eine exakte mathematische Beschreibung einer zu messenden parallaktischen Ellipse. Zuletzt werden noch Methoden zur Reduktion von Sternpositionen auf Fotoplatten dargestellt, und auf das CoRoT-Feld angewendet. Das Ergebnis dieser Untersuchung ist, dass bei der korrekten Verwendung von Referenzsternen sehr wohl eine Genauigkeit im Bereich von Millibogensekunden möglich scheint.



Artist concept of CoRoT (image credits: CNES/D. Ducros.)

Contents

1	Introduction	1
1.1	Distance Measurement	1
1.1.1	Standard Candles:	2
1.1.2	How does trigonometric parallax measurement work?	6
1.2	Astrometry	8
1.2.1	Stellar Parallaxes	8
1.2.2	Astrometry today	13
1.3	The CoRoT mission	19
1.3.1	What makes high precision photometry so interesting?	19
1.3.2	How is CoRoT's observing?	20
1.3.3	Why in space?	24
1.3.4	What are the demands to the instrumentation?	24
1.3.5	How can the pulsations be classified?	24
1.3.6	What are the main scientific goals of CoRoT and what is the question the author is working on?	25
1.3.7	The Optical System	26
1.3.8	The point spread function (PSF) of the Seismological Channel	27
1.3.9	The CCDs	27
1.3.10	The simulated data:	28
2	Tests with all PSFs	33
2.1	First thoughts about optical effects and centroiding algorithms	33
2.1.1	Effects changing the shape of the PSF	33
2.1.2	ESO-MIDAS Centroid Search Routines:	34
2.1.3	Positions derived with different methods	35
2.2	Tests of the accuracy of the MIDAS methods	38
2.2.1	Grouping and trends	38
2.2.2	Coefficient of determination	43
3	Tests with one shifted PSF	49
3.1	Creation of shifted Originals	50
3.1.1	Rebinning of the simulations	50

CONTENTS

3.1.2	Creation of shifted images using spline interpolation	51
3.2	Some centroiding algorithms and their quality	52
3.2.1	The center of gravity algorithm	52
3.2.2	Data analysis of the centroid results for rebinned images	54
3.2.3	Symmetry	56
3.2.4	Triangle fit	60
3.2.5	Jitter Simulations	60
4	Astrometry with "real" CoRoT data	63
4.1	The Observations	63
4.1.1	First data analysis, star count trials, identification of stars	65
4.2	Centroid positions for bright stars in field LRA01 for the two observations of 17:38 and 18:47 from October 16, 2007	68
4.2.1	Selection of stars	68
4.2.2	Starlist filter	68
4.2.3	Determination of star position and shifts between the two fields	69
4.3	Overlay of shifted Originals	72
4.3.1	Does increasing the number of observations of the same field also increase the astrometrical accuracy?	76
5	Parallax measurement, data reduction - a theoretical view	79
5.1	Gnomonic Projection	80
5.2	Displacement due to annual parallax	81
5.3	Methods to reduce star positions of photographic plates	82
5.3.1	Deriving of celestial coordinates of stars in the CoRoT fields us- ing reference stars	87
6	Conclusion	91
7	Danksagung	93
A	Glossary	101
B	The PSFs and their derived positions	103
C	IDL Routines	109
C.0.2	Rebinning the images	109
C.0.3	Spline shifting	110
C.0.4	Centroid algorithm	111
C.0.5	Symmetry algorithm	112
C.0.6	IDL Spline Routine	114
C.0.7	IDL Shift Routine	118
C.0.8	How to count the number of stars	121

CONTENTS

C.0.9	How to find the edge	122
C.0.10	Creating a list of stars	122
C.0.11	Creating a list of filtered stars	123
C.0.12	Creating list of stars with their positions	125
C.0.13	Determining the shift between two images	126
C.0.14	Subpixel shifts , "Overlay" method, parts 2 to 5	127
C.0.15	Turner method	130
D	Centroid results for all rebinned PSFs	133

Chapter 1

Introduction

In the first section some astrometric details are given along with a short introduction to trigonometric parallaxes. A short overview on the CoRoT mission, its aims, the optical system, the CCD's and the simulated point spread function are described.

The second section offers an overview on tests done with all simulated PSFs with standard positional algorithms from the ESO MIDAS package.

In the third section new tests to derive positional information from the simulated point spread function are developed. In section four some astrometric results from “real” observations with the COROT satellite are shown.

1.1 Distance Measurement

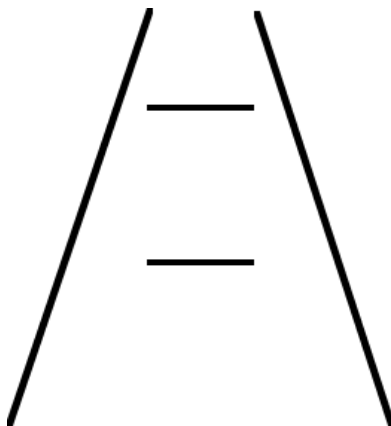


Figure 1.1: Ponzo effect

Its very important to know the distance to an object to judge its properties. Just as an illustration, let us consider the Ponzo effect (Mario Ponzo, Italian psychologist, 1882–1960) which shows convincingly that a wrong interpretation of the distance of an object leads to a improper judgment of its properties.

In Figure 1.1 we get the impression that the vertical lines are leading into the depth of the image, thus leading us to the interpretation that the two horizontal lines are lying in different distances. This leads us to the wrong interpretation that the two lines have different length. Our mind tries automatically to reconstruct a third dimension.

The direct estimation of interstellar distances via triangular parallaxes is one of the fundamental techniques in astrophysics. The entire astronomical distance scale is based on direct positional measurements. Without knowledge of distances we would still discuss, whether the stars were all just different colored

light sources on a sphere above our heads.

1.1.1 Standard Candles:

With the help of modern physics we can say which kind of mechanism can cause a certain color and we can simulate, what kind of star can cause certain physical effects.

We have models which allow us to calculate the properties of the flux emitted by a star with a certain temperature and consistency of its stellar atmosphere. One of the main models is the so called ATLAS Code which in our case has been used to simulate the stars observed by CoRoT. To test those models we still must measure the distance to some stars directly, because only when we know their distances we can reduce the apparent magnitude of a star to its absolute “real” magnitude.

Only if we know the real magnitude of a star we can test if the properties in brightness and color of the observed star really fit to the models. The main idea behind this is the so called “standard candles” -method. If we know the amount of light, the flux emitted by a “candle” in a certain time period, or from our point of view, the amount of light that reaches us in that time period, the so called magnitude of that “candle”. We can use the magnitude of this candle to derive its distance.

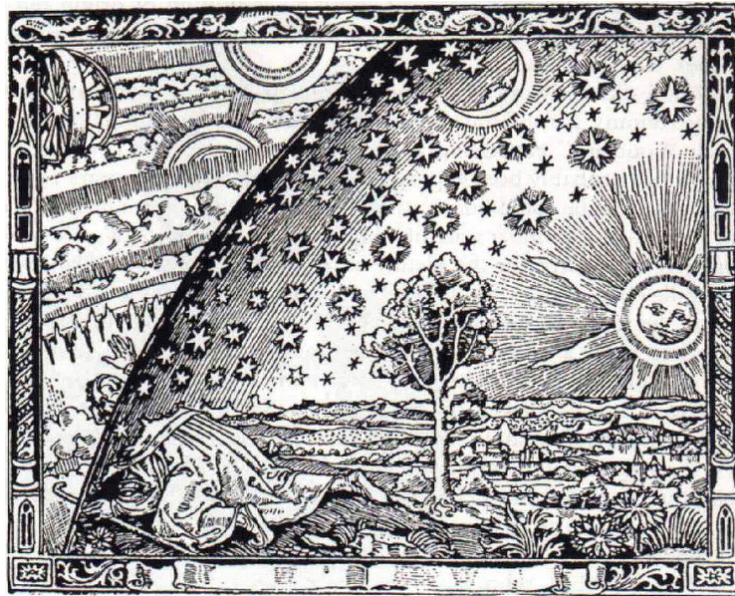


Figure 1.2: Camille Flammarion: L'atmosphère météorologie populaire. Paris 1888

apparent magnitude (δm): The first classification of stars was performed by the Greek astronomer Hipparchos, who tried to classify the stars in 6 classes, with the faintest stars being in the sixth class.

$$\Delta m = m_1 - m_2 = -2,5 \lg s_1/s_2 \quad (1.1)$$

Here, m_1 and m_2 are the apparent magnitudes of two stars . s_1 and s_2 are the amount of detected light coming from these two stars [42]. This scale is relative, the zero-point is set by the apparent magnitude of Vega which is set to 0. A logarithmic measure is appropriate because the sensitivity of the eyes to light is also logarithmic. This definition

of the apparent magnitude is the reason why the naked eye is sensitive for stars down to a magnitude of 6.

absolute magnitude (M): The absolute magnitude of a star is defined as the apparent magnitude a star would have if its distance to the observer would be 10 parsec.

$$m - M = -5 + 5 \log_{10} r = -5 + 5 \log_{10} \pi \quad (1.2)$$

How do we know the distance of an object that is so far away as the stars? The idea of using magnitudes to classify the stars is as old as the beginning of astronomy. But the idea that all stars are relatively similar objects with a similar intrinsic brightness only came up with the Copernican revolution (“stars are distant suns”, James Gregory wrote in 1668). To evaluate the Copernican model by means of parallax measurements and not only planetary orbits and Kepler’s laws, it was necessary to measure the distance to a star directly. A model by means of parallax measurement and not only planetary orbits and Kepler’s laws was necessary to measure the distance to a star.

Already in the 1630 Galileo Galilei lets the person Salviati (who represents Galilei’s own ideas) state in his *Dialogo* an interesting idea ([18]).

This idea is based on two fundamental views of the world which were not that obvious for the people of that time.

He, Salviati proposes that the distance to the stars can be measured by measuring the movement of a **brighter**, thus nearer star in front of to a **fainter** star, which is further away. The idea is a simple geometric idea, but it is based on two facts.

First, and that is still in question at that time, Salviati assumes that the earth is not the center of the world and moves around the sun. If the earth moves during one year from one side of the sun to the other side, the angle under which distant stars are seen compared to a set of background stars should change during one year. Salviati’s second fundamental theory is that brighter stars must be closer to the sun than fainter stars, but this assumption was perhaps more a stumbling block than a good idea concerning the search for parallaxes.

As we know now the absolute magnitude of Sirius is 25 times that of the sun, that means if we put the sun in the same position as Sirius of 2,64 pc or 8.6 light years, and we make the same assumption as Salviati, we would get a distance for Sirius 5 times the distance of the sun.

The direct and fundamental way to find out how far away our object really is, is by triangular distance measurement. If we measure the distance to a star by a triangular distance measurement, the so called annual parallax, we know the “real” distance. If we now calculate their absolute magnitude, we see that stars do not have the same absolute magnitude at all.

In table 1.1 we see that Sirius is only as bright compared to other stars because he is so close to us. It thus is not that simple to judge the distance of a star by just comparing its relative magnitude.

1.1. DISTANCE MEASUREMENT

Table 1.1: some prominent stars

name	m	M	distance
Sun	$-26^m, 73$	$+4^M, 84$	$4,851 \cdot 10^{-6}$ pc
Sirius	$-1^m, 46$	$+1^M, 43$	2,64 pc
Vega	$+0^m, 03$	$+0^M, 58$	7,75 pc
Pollux	$+1^m, 15$	$+1^M, 08$	10,34 pc
Spica	$+1^m, 04$	$-3^M, 51$	81,3 pc
Rigel	$+0^m, 12$	$-6^M, 78$	240 pc

The astronomers had to wait until 1830 when the first distance to a neighboring star was measured directly by astrometric means, and it was possible to show if Galileo's idea was working. It was possible to prove the Copernican theory of a movement of the earth around the sun directly by observing the positions of the surrounding stars.

If we know something about the absolute magnitude of some stars we can classify them and then try to find a star of the same type and if this star is brighter he should be nearer. The distance correlates directly to the magnitude of a star, but this only works if the star has the same type, the same age, the same chemical properties, etc. Nowadays, the well know method of "standard candles" is based on the direct trigonometric distance measurement of "the" standard candles. And every increase in our knowledge of distances increases also the accuracy of all astrophysical theories based on the absolute magnitude of stars, this is what makes astrometry so fundamental. The increase in astrometric knowledge is worth the effort. Compare the picture of Camille Flammarion (figure 1.2), with figure 1.3, showing the solar surrounding up to a distance of 50 pc and at table 1.4, a list of those stars including their parallaxes and a short description. This is what we know now, 400 years after the invention of the telescope and Galileo's theories how to determine the position of the Earth in the Universe.

CHAPTER 1. INTRODUCTION

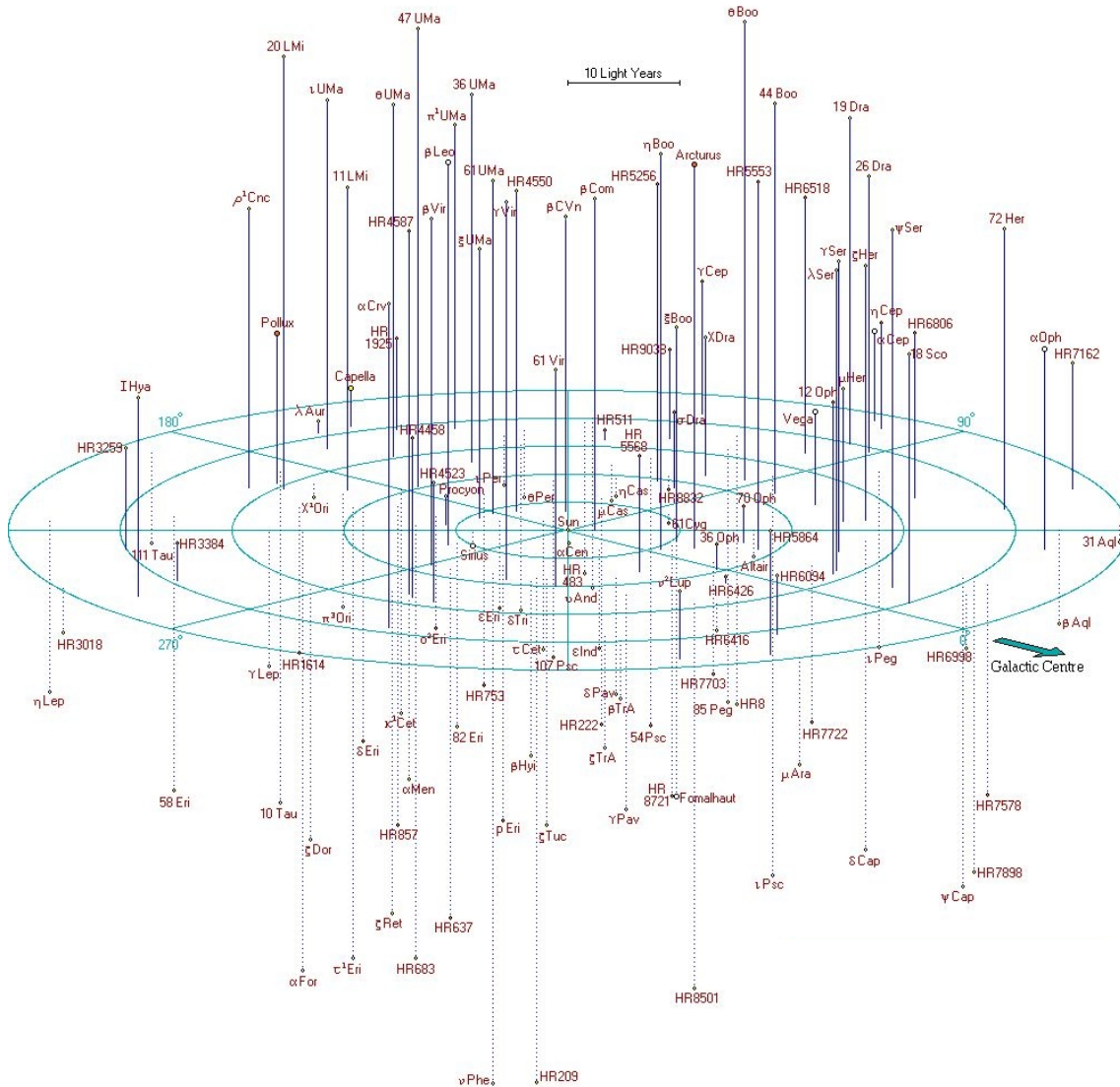


Figure 1.3: Map of the solar surrounding of 50 ly from <http://www.atlasoftheuniverse.com/>

1.1. DISTANCE MEASUREMENT

1.1.2 How does trigonometric parallax measurement work?

Historic distance measurement in geography

To measure the length of a line a fundamental measuring unit is needed to compare the measured length with this fundamental unit and to indicate numbers to this fundamental length. If we consider geographical distances, direct measurement already is a problem.

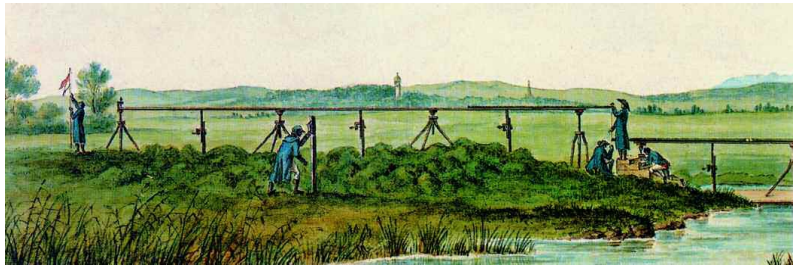


Figure 1.4: Measurement of fundamental lines, Switzerland, ca. 1790

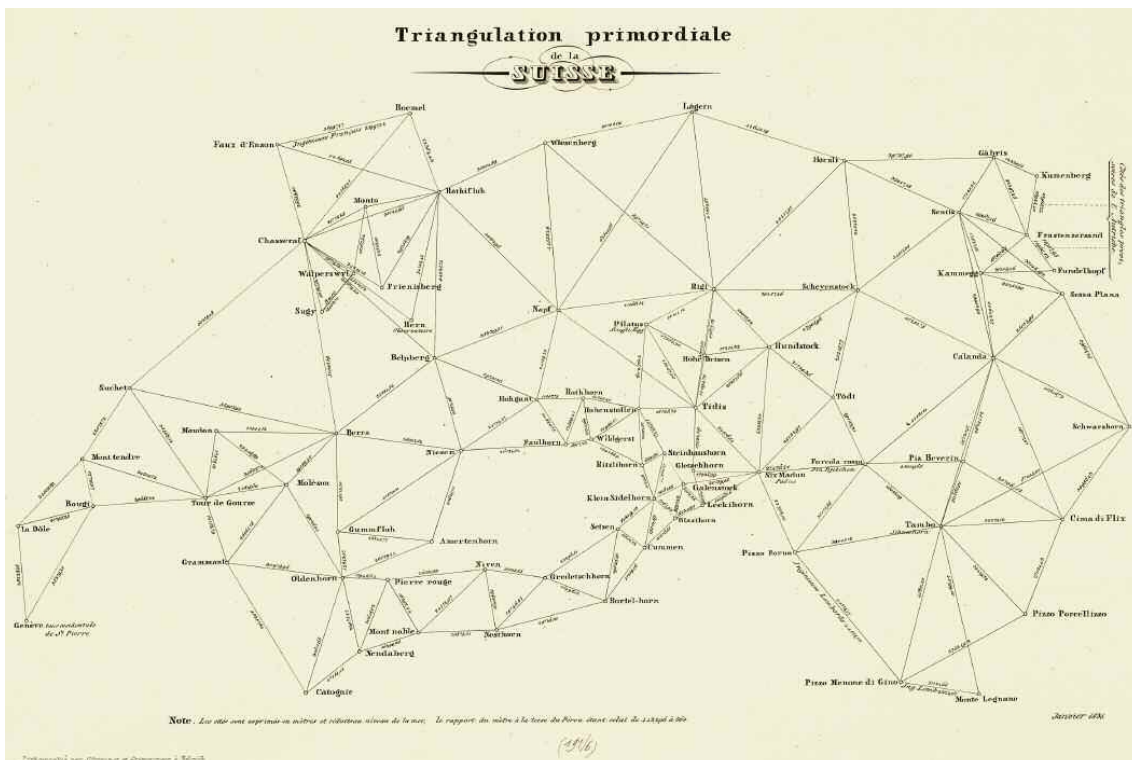


Figure 1.5: Fundamental lines of Switzerland, ca. 1790

In the beginning of map making and without the availability of optical instruments to measure angles with a good accuracy, the direct measurement of distance was the way how the fundamental distances were measured. Just as an illustration a picture (Figure

1.4) from the measurement of fundamental lines in Switzerland from 1790 is considered. Here the researchers are using iron sticks to measure distances between some fundamental points. It was important to measure a direct line and one problem was to reduce errors caused by deviations between connection points between two sticks. Often the terrain was not easy to handle. A flat and stable underground would be needed and in Switzerland it was not always available. The goal of these measurements was the creation of a map of Switzerland (Figure 1.5) with realistic dimensions. As one can easily see, some points were (and are) not easily reachable.

Trigonometry But some simple geometry can help us to measure the distance between these points (see figure 1.6). To measure the height of a mountain some trigonometric calculations and a good optical system can solve this problem.

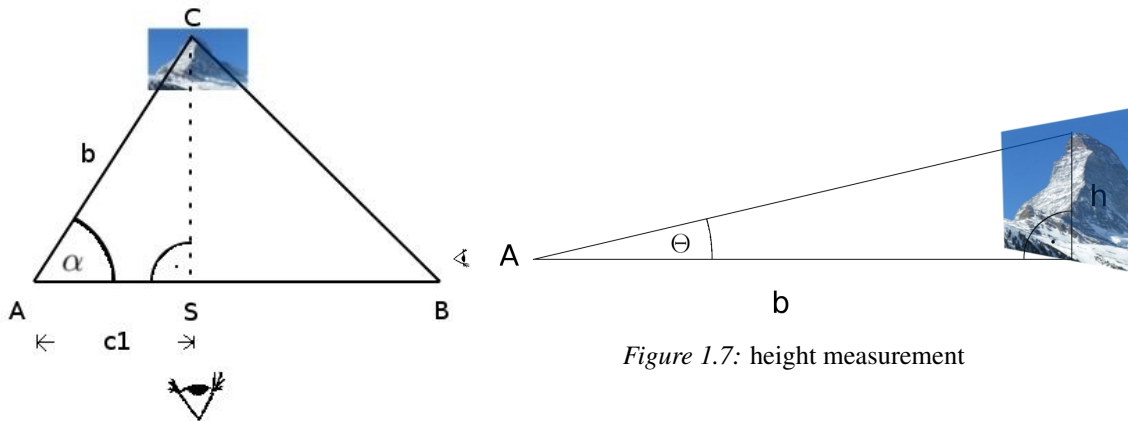


Figure 1.6: distance measurement

A, B, C are the corners of a triangle where C is unreachable. S is the point where the perpendicular line from C subtends the line AB. We assume the length of c1 to be known.

We know that

$$\cos \alpha = \frac{c1}{b} \Rightarrow b = \frac{c1}{\cos \alpha}$$

Thus we obtain the distance b to the base of the mountain.

When we have this distance we only have to measure the angle between the horizon and the mountain top along a perpendicular line and thus we can deduce the height of the mountain directly by the formula

$$\tan \theta = \frac{h}{b} \Rightarrow h = b \tan \theta$$

This only works if we assume that the difference between the distance to the mountain top and the distance to the mountain base is negligible.

The reason why we only need to measure the position of the mountain top twice, from the two points A and S, is that it is sufficient for us to know the distance c_1 , and the angle α . All other distances can be derived from this distance.

To measure parallaxes, two different points of view are needed. To formulate it in philosophical terms: we can say that without changing our point of view we would not get any deeper “insight” into the real distance scales of our world.

1.2 Astrometry

To come back, changing our point of view sounds easy on human scales, but when we talk about faraway stars it is not that easy to change our point of view as far as to gain sufficient significance for an angle measurement.

1.2.1 Stellar Parallaxes

Our aim is to determine the distance r to a star using observations of positions of stars in the sky. Parallax measurement is based on the measurement of the angular distance between a star **A** and a set of background stars B_i . We assume that the principle star **A** is close to the sun compared to the background stars B_i , for which we assume they are lying at infinite distance.

Because the angle under which the constellation is observed from earth changes during the year with the position of the earth along its orbit, the distance $A - B_i$ changes too.

If we assume that the background stars B_i are far away and thus do not show any intrinsic movement, then the trace of **A** looks like a small picture of the motion of the earth around the sun, depending on the position of the constellation relative to the plane of the Ecliptic.

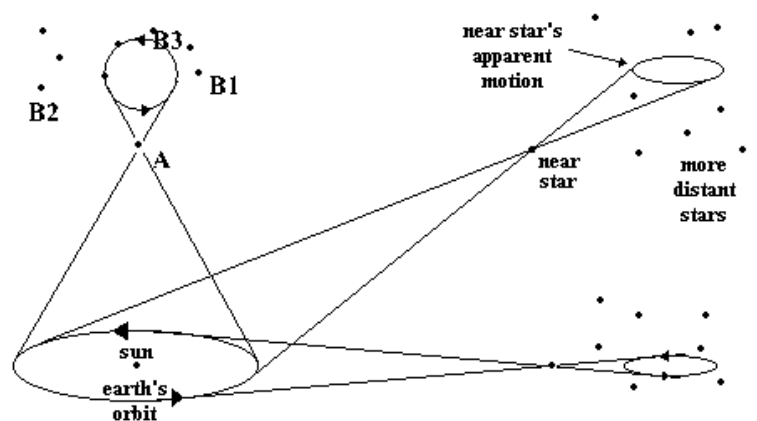


Figure 1.8: parallax measurement

If the constellation is situated in the plane of the ecliptic the star shows only a movement along a line from left to right and back. If earth is on the “left side” of the sun, the

star is on its furthestmost right position. If earth is on the “right side” of the sun, the star is on its furthestmost left position.

If the constellation lies at the north pole of the plane of the ecliptic, a perfect copy of earth’s ellipse around the sun is achieved.

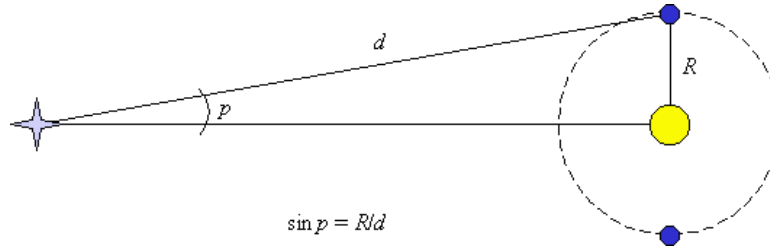


Figure 1.9: parallax angle

Annual parallax:

The semi major axis of this ellipse is called **annual parallax p** . The annual parallax is also the angle under which the radius of earth’s orbit around the sun can be seen from a star. We see that for the largest measurable parallax angle can be measured when the triangle earth-sun-star is rectangular, which is the case at least two times a year.

If we assume that the difference between the hypotenuse and the adjacent leg of this triangle is negligibly small, we have the simple trigonometric relation:

$$\sin(p) = R/r$$

, where **R** is the distance sun-earth the so called Astronomical Unit(AU , $1AU = 1.496 * 10^{11}m$ [42]), **r** the distance between sun and star, and **p** is half the parallax angle. So we get the well known formula for the distance r :

$$r = 1 / \sin p$$

. The unit of r gives the definition of the AU, The smaller a star’s annual parallax, the farther away the star lies from earth (see figures 1.8 and 1.9).

The distance to the closest star e.g. (Proxima Centauri) with a parallax of 742.12 milliarcseconds expressed in Astronomical Units is $206265 * 0.74212 = 153073, 3818AU$, the measured unit is the parallactic angle p which is measured in arc seconds ("), arc minutes (') and degrees (°) so, to be able to work with more handy numbers in distance scaling, the unit parsec was defined:

If the parallactic angle has an amount of $1'' = 1/60' = 1/60/60^\circ$ during one year, the star A lies in a distance r of **1 parsec** to the solar system. This distance corresponds to 206265 or $\frac{360}{2 * \pi} * 60 * 60$ times the distance of Earth to the sun. Calculated in SI Units: $1pc = 3.086 * 10^{16}m = 206265AU$ [42].

Parallax determination Parallax determination its not as simple as it may sound. First there are some effects adding up to the positional shifts of a star.

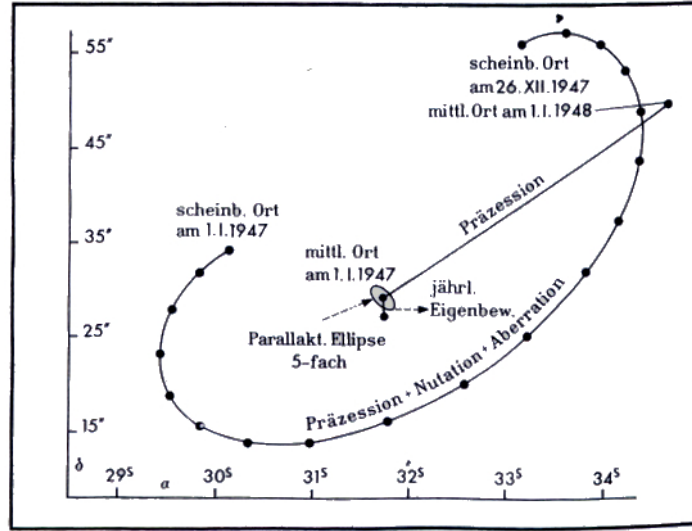


Figure 1.10: Influences on the spherical position of a star [2]

In plot 1.10 we see how the rotation of earth (precession, nutation and aberration) and of the sun relative to the stars (proper motion, “Eigenbew.”) change our “point of view”, also we see the relative size of these effects compared to the parallax ellipse caused by the revolution of the earth around the sun. The parallax ellipse which is in this image 5 times enlarged to be visible. The star depicted is Ross 248 with a parallax of 3.18 pc, so the parallax ellipse has a radius of only 1/3.18 ”.

Reduction of data First we have to reduce the position of our star to geocentric coordinates (see [2], page 54). If we know the inclination of the ecliptic ϵ , the radius vector of the Earth \vec{r} and the longitude of the sun L we can derive the following relations between the heliocentric coordinates α, δ and the geocentric coordinates α' and δ' :

$$\alpha' - \alpha = r\pi(\cos \alpha \cos \epsilon \sin L - \sin \alpha \cos L) \sec \delta \quad (1.3)$$

$$\delta' - \delta = r\pi(\cos \delta \sin \epsilon \sin L - \cos \alpha \sin \delta \cos L - \sin \alpha \sin \delta \cos \epsilon \sin L) \quad (1.4)$$

To measure a parallax Π it is necessary to have at least one second star (coordinates: α_v, δ_v) in the field with a very small parallax. Measured is the periodic shift relatively to this second star.

In the following relation, we define the coefficient of π to be N , where

$$N = r(\cos \alpha \cos \epsilon \sin L - \sin \alpha \cos L). \quad (1.5)$$

This gives us for two stars, where one (p) has a measurable parallax and therefore is near and a second background star (v) is so far away that his parallax cannot be measured, the following relations:

$$\alpha'_p - \alpha_p = \Delta\alpha_p = N\pi \sec \delta_p, \text{ and} \quad (1.6)$$

$$\alpha'_v - \alpha_v = \Delta\alpha_v = N\pi \sec \delta_v \quad (1.7)$$

The distance between the two stars $m = \alpha'_p - \alpha'_v$ is now measured on a plate or as in our case on the CCD:

$$m = \alpha_p + \Delta\alpha_p - (\alpha_v + \Delta\alpha_v). \quad (1.8)$$

As $\Delta\alpha_v$ is defined to be negligible we have

$$m = \alpha_p - \alpha_v + N\pi \sec \delta. \quad (1.9)$$

For a second observation at a different time of the year we have

$$m_1 = \alpha_p - \alpha_v + N_1\pi \sec \delta. \quad (1.10)$$

Finally we have the relation for the measured values of m and m_1 :

$$\pi = \frac{(m_1 - m) \cos \delta}{N_1 - N}. \quad (1.11)$$

By measuring multiple times we get observations like the one shown in figure 1.11. These positions allow the fitting of an ellipse.

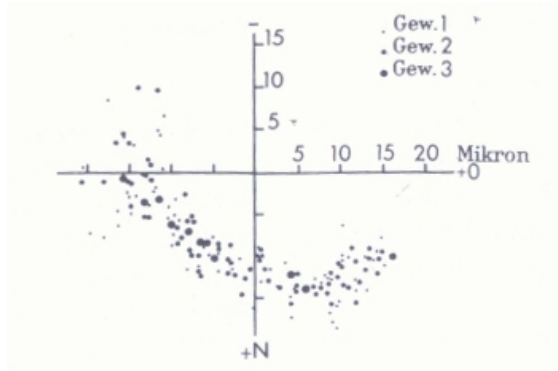


Figure 1.11: Measurements of Ross 248



Figure 1.12: Interpolated parallactic ellipse

The measured parallactic displacement is for the star Ross 248 with a parallax of $0''318$ taken in the time intervall between 1937 and 1946, it is nicely visible in the plot, that the star cannot be observed during the time from February to June. These data are good enough to fit an interpolated parallactic ellipse (figure 1.12).

If we compare the strength of the effects in angular size and the effort needed to measure parallaxes with photo plates, we can understand that the distance measured is really enormous. It shows how difficult it is and also why it took until 1837 that the first parallax for one of the nearest stars (61 Cygni) was measured by Friedrich Wilhelm Bessel.

1.2. ASTROMETRY

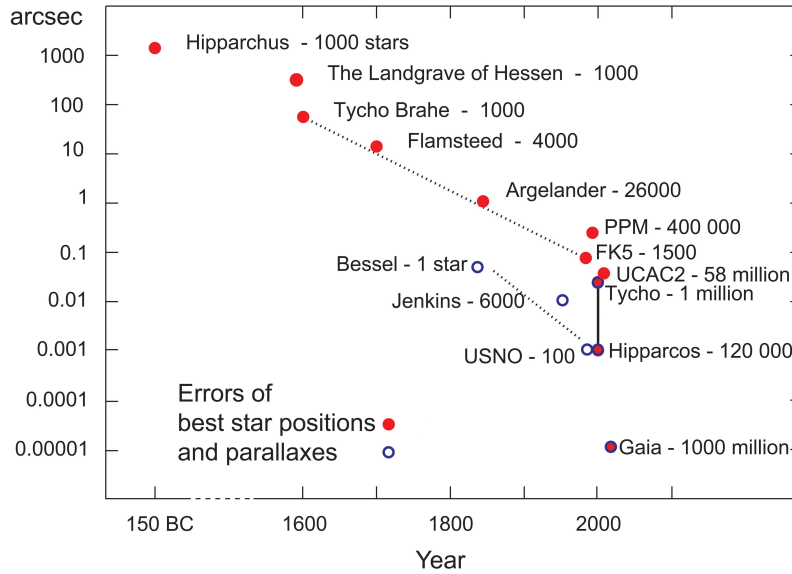


Figure 1.13: Historic Development of astrometric accuracy [5]

Historical development It is very interesting to compare the development of astrometry and to take evaluate the increase in sensitivity (see figure 1.13). We begin with Hipparchus, who measured star positions with the naked eye, and one can say that he reached an accuracy of some degrees, we pass on to the maximum that was reachable with systematic measuring techniques but still without using a telescope (Tycho Brahe, Landgrave of Hessen, both around 1600), reaching an accuracy of some arcseconds. For the famous “Bonner Duchmusterung”, reaching an accuracy of one arcsecond, Friedrich Wilhelm August Argelander used already a telescope with an entrance pupil of 78 mm and a focal length of 630 mm, comparable to modern small field glasses. Modern ground based catalogs like the FK5 catalog or the UCAC2, which is right now one of the most complete catalogs available, are reaching accuracies of around 0.1 arcseconds. For some selected stars special astrometrical research has been done to determine trigonometric parallaxes. First we have the measurement of Bessel, and nowadays we have around 10000 measured parallaxes measured from the ground (in 1952 Louise F. Jenkins, published 6000 trigonometric parallaxes in the General catalogue of trigonometric stellar parallaxes, in 1995 the Yale Trigonometric Parallaxes catalog includes in it is fourth edition 15,994 parallaxes for 8,112 stars [1]). This also shows us, how slowly the number of measured parallaxes increased. For about 100 stars we reach with very sophisticated techniques an astrometric accuracy from the ground of one milliarcsecond. This value was also reached by the extremely important HIPPARCOS mission in 1989.

Hipparcos

The original detection technique was based on a **transit instrument** with a simple photo diode detecting the angular distance between the observed stars. The time distance be-

tween a transit of a star and the next star was measured. The measurement accuracy depended on the constancy of the satellite's rotation period. In this case a sharp point spread function (PSF, see Glossary A) to have sharp time signals of the transit, was the goal of the optical design. Comparing that with current possibilities from space astronomy, HIPPARCOS (High Precision Parallax Collecting Satellite, 1989) still reached an astrometric accuracy of up to one milliarcsecond, measured the parallaxes of 120 000 stars and reached an accuracy of **0.64 milliarcseconds**.

1.2.2 Astrometry today

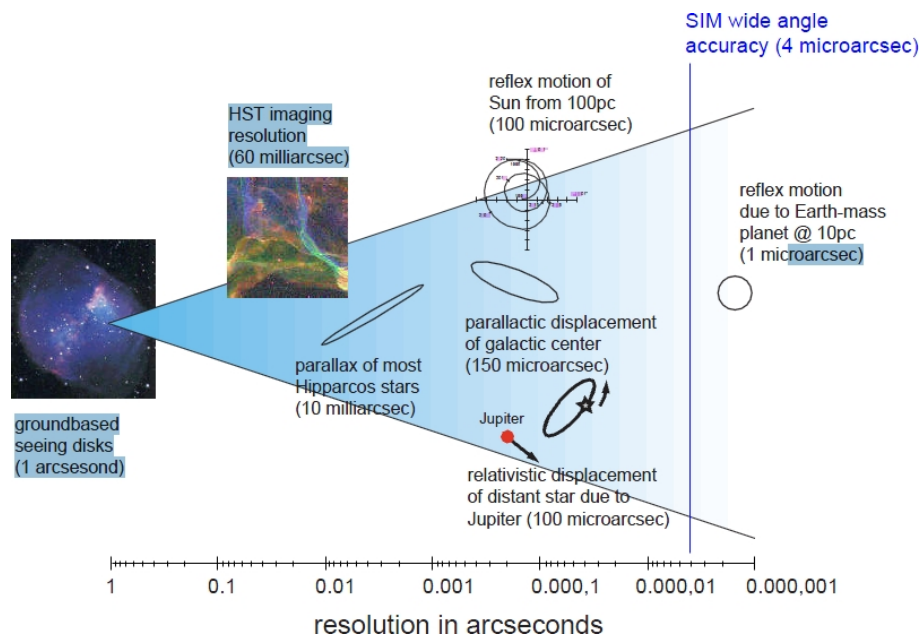


Figure 1.14: Comparison of astrometric accuracies [36]

Actual and future missions The next major improvement in Astrometry was expected with the launch of the U.S. SIM PlanetQuest mission (Space Interferometry Mission) which was planned for 2015 but is actually deferred to unknown time [37] and the European Cornerstone Mission GAIA (2011) [5] both with an expected accuracy of **1 μ asec**.

In picture 1.14 we see a comparison of some astrometrical effects, error circles and the sensitivity of the SIM PlanetQuest Mission. The main interest for us in this figure is the relative size of e.g. the effect of parallactic displacement of the galactic center with a size of around 150 microarcseconds versus the sensitivity of ground based astrometry which is around 1 arcsecond for classical astrometry obtained from photo plates and astrographs. Also we see the effect of relativistic displacement caused by Jupiter with a size of 100 microarcseconds. As a comparison we see the error circle for a modern astrometrical mission like SIM with 4 microarcseconds. This accuracy would even allow us to measure the reflex motion of a solar like star caused by a planet with one earth mass in a distance

1.2. ASTROMETRY

of 10 parsec. Also we see the resolution reached by high resolution space missions like the Hubble Space Telescope of 60 milliarcseconds. When we include in this figure the HIPPARCOS error circle of 0.64 milliarcseconds (figure 1.15) it becomes evident why an increase in astrometric knowledge is so important.

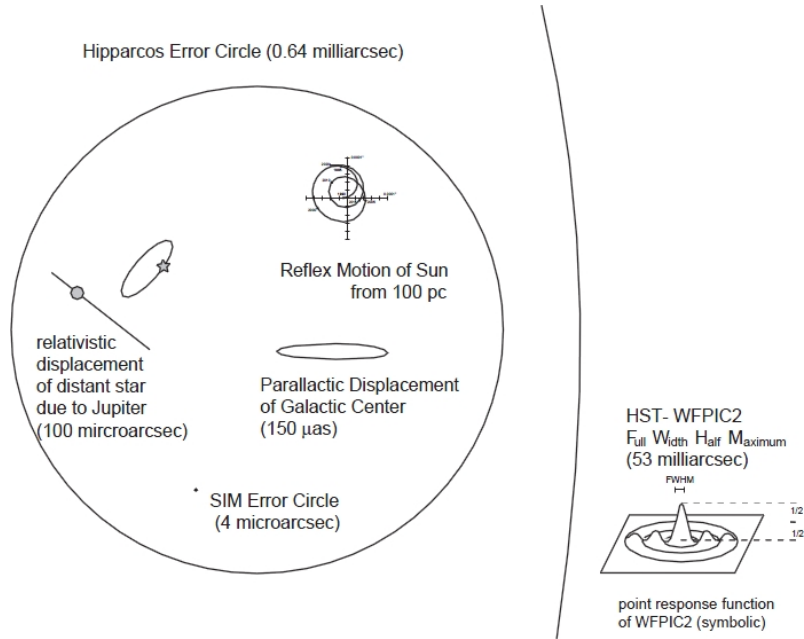


Figure 1.15: Astrometric effects with HIPPARCOS error circle [37]

Today the main source of astrometrical information is the result of the HIPPARCOS mission from 1989. The next direct increase in positional knowledge especially in the astrometrical accuracy will be achieved if the astrometrical GAIA [5] mission is successfully delivering results which can be expected for 2011 or later. There is also the very promising SIM Planetquest Mission [37] concept, which is actually deferred indefinitely by the NASA headquarters [38].

Astrometry with photometric telescopes:

The actual development of new Satellites dedicated to Astroseismology and Photometry like CoRoT and MOST with their defocused optics and long time observations of stars on one hand and the need for better astrometrical data on the other hand lead to the question whether it is possible to use the results of these missions to derive positional information.

It is obvious that there is always a certain positional information if a star is observed by a telescope, but the main question is how accurate our telescopes can measure positions. For a defocused optical system it is also interesting what kind of algorithm is useful to derive positional information from a defocused star field. In the actual case of CoRoT we do not only have a simple defocused optical system but even an asymmetrical system of mirrors and lenses. How do the different algorithms react to such a system?

CHAPTER 1. INTRODUCTION

The question is: “*Is it feasible to determine parallaxes with the CoRoT (CONvection et ROTation et Transits) satellite, and which accuracy can be achieved?*”. We will try to answer this question in the following chapters.

1.2. ASTROMETRY

Table 1.2: Solar neighborhood: Stars up to 50 ly distance to the sun, part I

Catalog Name	MV	MAbs	Prllx mas	Dist ly	Proper Name	Notes
Sun	-26.8	4.83	–	0.00	Sol	Has 8 planets
Alpha Centauri	-0.29	4.06	742.12	4.39	Rigil Kentaurus	
Alpha Canis Majoris	-1.46	1.43	379.21	8.60	Sirius	
Epsilon Eridani	3.72	6.18	310.75	10.50		Has a planet
Alpha Canis Minoris	0.36	2.64	285.93	11.41	Procyon	
61 Cygni	4.79	7.07	285.42	11.43		
Epsilon Indi	4.69	6.89	275.76	11.83		
Tau Ceti	3.49	5.68	274.17	11.90		
Omicron Eridani	4.43	5.92	198.24	16.45	Keid	
70 Ophiuchi	4.03	5.50	196.62	16.59		
Alpha Aquilae	0.77	2.21	194.44	16.77	Altair	
Sigma Draconis	4.67	5.87	173.41	18.81	Alsafi	
HR5568	5.72	6.86	169.32	19.26		
Eta Cassiopeiae	3.46	4.59	167.99	19.42	Achird	
36 Ophiuchi	4.33	5.46	167.08	19.52		
HR7703	5.32	6.41	165.24	19.74		
82 Eridani	4.26	5.35	165.02	19.76		
Delta Pavonis	3.55	4.62	163.73	19.92		
HR8832	5.57	6.50	153.24	21.28		
Xi Bootis	4.54	5.41	49.26	1.85		
HR753	5.79	6.50	138.72	23.51		
HR6426	5.91	6.61	138.2	23.6		
HR222	5.74	6.38	134.04	24.33		
107 Piscium	5.24	5.87	133.91	24.36		
Beta Hydri	2.82	3.45	133.78	24.38		
Mu Cassiopeiae	5.17	5.78	132.40	24.63		
HR8721	6.48	7.07	130.94	24.91		
Alpha Piscis Austrini	1.15	1.72	130.08	25.07	Fomalhaut	0.87 ly from Fomalhaut
Alpha Lyrae	0.03	0.58	128.93	25.30	Vega	Surr. by a ring of dust
Pi Orionis	3.19	3.67	124.60	26.18	Tabit	
Chi Draconis	3.55	4.02	124.11	26.28		
p Eridani	5.07	5.52	122.75	26.57		
Xi Ursae Majoris	3.79	4.18	119.7	27.2	Alula Australis	
Beta Canum Venaticorum	4.24	4.63	119.46	27.30	Chara	
Mu Herculis	3.42	3.81	119.05	27.40		
61 Virginis	4.74	5.09	117.30	27.81		
Zeta Tucanae	4.23	4.56	116.38	28.03		
Chi Orionis	4.39	4.70	115.43	28.26		
HR6416	5.47	5.75	113.81	28.66		
HR1614	6.22	6.49	113.46	28.75		
HR7722	5.73	6.00	113.33	28.78		
Gamma Leporis	3.59	3.83	111.49	29.25		
Delta Eridani	3.52	3.74	110.58	29.50	Rana	
Beta Comae Berenices	4.23	4.43	109.23	29.86		
HR4550	6.42	6.61	109.21	29.87		
Kappa Ceti	4.84	5.03	109.18	29.87		
Gamma Pavonis	4.21	4.39	108.50	30.06		
HR4523	4.89	5.06	108.23	30.14		
HR4458	5.96	6.06	104.84	31.11		
61 Ursae Majoris	5.31	5.41	104.81	31.12		High velocity star

CHAPTER 1. INTRODUCTION

Table 1.3: Solar neighborhood: Stars up to 50 ly distance to the sun, part II

Catalog Name	MV	MAbs	Prllx mas	Dist ly	Proper Name	Notes
12 Ophiuchi	5.77	5.82	102.27	31.89		
HR511	5.63	5.64	100.24	32.54		
HR5256	6.49	6.47	98.97	32.96		
Alpha Mensae	5.08	5.05	98.54	33.10		
Beta Geminorum	1.14	1.07	96.74	33.72	Pollux	Nearest giant star, has a planet
HR857	6.05	5.97	96.33	33.86		
Iota Persei	4.05	3.94	94.93	34.36		
HR9038	6.36	6.19	92.68	35.19		
Zeta Herculis	2.81	2.64	92.63	35.21	Rutilicus	
Delta Trianguli	4.84	4.66	92.20	35.38		
Beta Virginis	3.59	3.40	91.74	35.55	Zavijava	Has a planet
HR637	6.12	5.93	91.63	35.60		
Beta Leonis	2.14	1.92	90.16	36.18	Denebola	
HR6806	6.38	6.15	90.11	36.20		
54 Piscium	5.88	5.65	90.03	36.23		Has a planet
Gamma Serpentis	3.85	3.62	89.92	36.27		
11 Leonis Minoris	5.40	5.16	89.45	36.46		
Theta Persei	4.10	3.85	89.03	36.63		
Alpha Bootis	-0.05	0.31	88.85	6.71	Arcturus	Most luminous star in 50ly
Eta Bootis	2.68	2.41	88.17	6.99	Muphrid	
HR5553	6.00	5.69	86.69	37.62		
Zeta Doradus	4.71	4.38	85.83	38.00		
Lambda Serpentis	4.42	4.07	85.08	38.34		
Iota Pegasi	3.77	3.42	85.06	38.34		
Delta Capricorni	2.85	2.49	84.58	38.56	Deneb Algedi	
Gamma Virginis	2.74	2.38	84.53	38.59	Porrima	
Zeta Reticuli	4.62	4.21	82.65	39.46		
Zeta Trianguli Australis	4.90	4.49	82.61	39.48		
HR3384	6.38	5.95	82.15	39.70		
HR1925	6.21	5.77	81.69	39.93		
Beta Trianguli Australis	2.83	2.38	81.24	40.15		
85 Pegasi	5.80	5.33	80.63	40.45		
Rho Cancri	5.96	5.47	79.80	40.87		Has 4 planets
HR3259	5.95	5.45	79.48	41.04		Has 3 planets
HR483	4.96	4.45	79.09	41.24		
Lambda Aurigae	4.69	4.18	79.08	41.24		
HR683	6.33	5.81	78.88	41.35		
44 Bootis	4.83	4.30	78.39	41.61		
HR6518	6.44	5.90	78.14	41.74		
36 Ursae Majoris	4.82	4.28	77.82	41.91		
HR6094	5.37	4.82	77.69	41.98		Has a planet
HR4587	5.54	4.99	77.48	42.10		
Alpha Aurigae	V 0.07	-0.49	77.29	42.20	Capella	2 giant stars + 2 red dwarfs 0.14ly away
HR6998	5.85	5.28	77.02	42.35		
58 Eridani	5.49	4.87	75.10	43.43		

1.2. ASTROMETRY

Table 1.4: Solar neighborhood: Stars up to 50 ly distance to the sun, part III

Catalog Name	MV	MAbs	Prllx mas	Dist ly	Proper Name	Notes
Upsilon Andromedae	4.10	3.45	74.25	43.93		Has 3 planets
Theta Ursae Majoris	3.17	2.52	74.15	43.99	Al Haud	
HR8501	5.36	4.69	73.47	44.39		
HR8	6.07	5.39	72.98	44.69		
Beta Aquilae	3.71	3.03	72.95	44.71	Alshain	
10 Tauri	4.29	3.60	72.89	44.75		
Iota Piscium	4.13	3.43	72.51	44.98		
Gamma Cephei	3.21	2.51	72.50	44.99	Errai	Has a planet
Tau? Eridani	4.47	3.74	71.56	45.58		
18 Scorpii	5.49	4.76	71.30	45.74		
47 Ursae Majoris	5.03	4.29	71.04	45.91		Has 2 planets
26 Draconis	5.23	4.49	70.98	45.95		
Alpha Fornacis	3.80	3.05	70.86	46.03	Fornacis	
HR7578	6.22	5.46	70.34	46.37		
Pi Ursae Majoris	5.63	4.86	70.07	46.55		
Alpha Ophiuchi	2.08	1.30	69.84	46.70	Rasalhague	
Eta Cephei	3.41	2.63	69.73	46.77		
72 Herculis	5.38	4.59	69.48	46.94		
Nu Lupi	5.65	4.83	68.70	47.48		
Theta Bootis	4.04	3.22	68.63	47.52	Asellus Primus	
Iota Ursae Majoris	3.12	2.29	68.32	47.74	Talita	
HR7898	6.36	5.53	68.28	47.77		1.68ly from Psi Capricorni
111 Tauri	5.00	4.17	68.19	47.83		
Psi Serpentis	5.86	5.03	68.16	47.85		
Psi Capricorni	4.13	3.30	68.16	47.85		
Alpha Corvi	4.02	3.17	67.71	48.17	Alchibah	
I Hydrae	4.93	4.07	67.19	48.54		
20 Leonis Minoris	5.37	4.50	67.14	48.58		
HR209	5.80	4.93	66.92	48.74		
Alpha Cephei	2.45	1.58	66.84	48.80	Alderamin	
HR7162	5.20	4.32	66.76	48.86		
Eta Leporis	3.71	2.82	66.47	49.07		
Nu Phoenicis	4.97	4.08	66.43	49.10		
19 Draconis	4.88	3.99	66.28	49.21		
31 Aquilae	5.17	4.27	66.01	49.41		
HR3018	5.36	4.45	65.79	49.58		
HR5864	6.01	5.09	65.60	49.72		
Mu Arae	5.12	4.23	65.46	49.83		Has 4 planets

1.3 The CoRoT mission



Figure 1.16: Artist concept of CoRoT (image credits: CNES/D. Ducros.)

CoRoT (Convection et Rotation et Transits) is mainly designed to detect photometric intensity changes. The goal is the measurement of physical parameters of the target stars like pulsations and rotation periods. Furthermore it is planned to detect extrasolar planets down to the size of the Earth by observing the intensity change of a star caused by the transit of planets.

1.3.1 What makes high precision photometry so interesting?

We want to increase our knowledge about the physical properties of stars. One of the main techniques to obtain information about the internal structure and inner mechanisms is the observation of the variations in the flux we receive from that star.

Distance measurement using Cepheids

One important example for the importance of the observation of periodic flux changes via so called “light curves”, was shown for the first time by Henrietta Swan Leavitt in 1912. She found that yellow stars with a fluctuation period of three days have a luminosity of about 800 times that of the Sun. Stars with a thirty-day period are 10000 times as bright as the Sun. She classified these stars as “Cepheids” because the first variable star of that type was δ Cephei, a star whose variability was already known since 1784 (by John Goodricke). This scale has been calibrated using nearby Cepheid stars, for which the

1.3. THE COROT MISSION

distance could be measured directly by the use of trigonometric parallaxes. Every increase of the astrometrical knowledge about the distance of Cepheids therefore increases the accuracy of the period-luminosity relationship. As a result of the HIPPARCOS mission the relation

$$M_v = -2.81 \log(P) - 1.43 \pm 0.1$$

(M_v stands for the absolute magnitude and P for the period in days) was found in 1997 [13]. A precise knowledge of this relationship is very important because Cepheids are used as “standard candles”. Cepheids are relatively bright and can be observed even in other galaxies. Assuming that these Cepheids follow the same physical laws as “our” Cepheids in our galactical neighbourhood we can say that a Cepheid with a certain period MUST have a certain absolute magnitude. In order to have a measured apparent magnitude, this star must lie within a certain distance. If we now can associate the object to which this star belongs, we can also measure the distance to this object with the period-luminosity relationship.

Physical properties of stars, the kappa mechanism

Next to the usability of this relationship for distance measurements there is also a physical mechanism behind these oscillations that gives us an extremely good insight into the inner life of these stars, e.g there is the so called kappa mechanism, describing intensity changes caused by changing the opacity “kappa”. The opacity of a plasma is dependent on its temperature and pressure. At the beginning of the cycle some exterior effect causes the plasma in a certain layer of the star to be compressed and therefore its temperature and its opacity increases. Due to the higher opacity, less radiation can pass this layer, the temperature due to radiation increases and therefore also the pressure. The high pressure causes the layer to expand. This expansion reduces also the temperature and opacity. Now the radiation can escape easily and the pressure drops, this leads to an infall of matter, which again causes the pressure and temperature in the layer to increase. This mechanism is very important because the result, the intensity change is measurable and the duration length of each cycle, the period of the intensity variation allows us to model the stars physical properties. It is as if we would be able to expand the star and let it react freely like an oscillating (nonlinear) pendulum. As we can determine the length of the pendulum observing its oscillation rate, we can also determine the size of a star, its temperature and its chemical properties measuring its oscillation rate.

1.3.2 How is CoRoT’s observing?

CoRoT was placed by a Soyuz rocket into a polar orbit at an altitude of 896 kilometers (see figure 1.17). CoRoT is pointing for a half year to the galactic anticenter direction and one half of the year to the galactic center direction. During the observation period the terminator of the earth will move due to the movement of the earth, so CoRoT is not

like MOST flying along the terminator, its orbit is always perpendicular to the viewing direction.

The orbit: This orbit allows a continuous observation of two fields for more than 150 days, plus due to straylight constraints by the sun which limits the observation period to less than 180 days, two additional short runs of around 30 days. The long run and the short run are always in the same CoRoT “eye”.

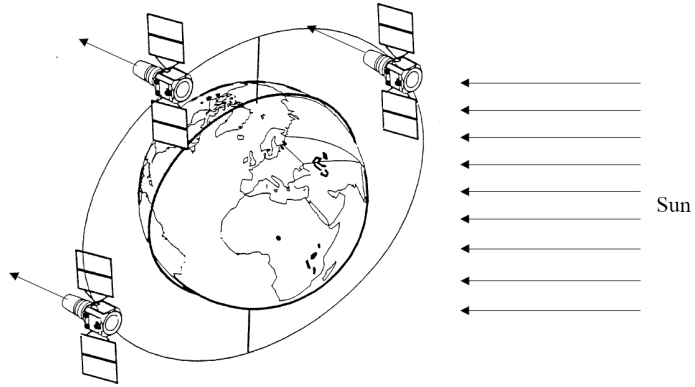


Figure 1.17: CoRoT’s orbit

Star fields (see CoRoT list of targets [44]) are observed over different time periods ranging from some weeks up to 120 days.

There are Long and Short Runs. CoRoT is always looking into the “night” direction, the telescope’s viewing direction is always pointing away from the sun. Due to the scientific question “Is there a difference between the stars of the galactic center direction and the stars lying in the opposite , the anticenter direction?” the Long Run fields were chosen to be pointing to the center and the anticenter of the milky way for a maximum time. Only to use the days when it is not possible to look at one of the two fields, because the sun comes too close to the minimal angular distance from the observational field two short run fields are observed for approximately 30 days (see list of all runs, table reftab:coordinates).

Its important to measure in intervals of relatively short time steps to have a reliable set of data and not to loose information by adding up light over a too long time period. Therefore it was decided to store images with a time rate of 30 seconds. Every second the content of one frame will be read out. This data frame is added up with 29 other frames for a virtual illumination time of 30 seconds per downloaded frame.

1.3. THE COROT MISSION

“COROT EYES”: COROT’s strategy is to observe two major fields the so called “eyes”. One eye is pointing directed towards the galactic center for the short and long runs SRC and LRc, and one eye pointing towards the anticenter with the short runs SRa and LRa fields.

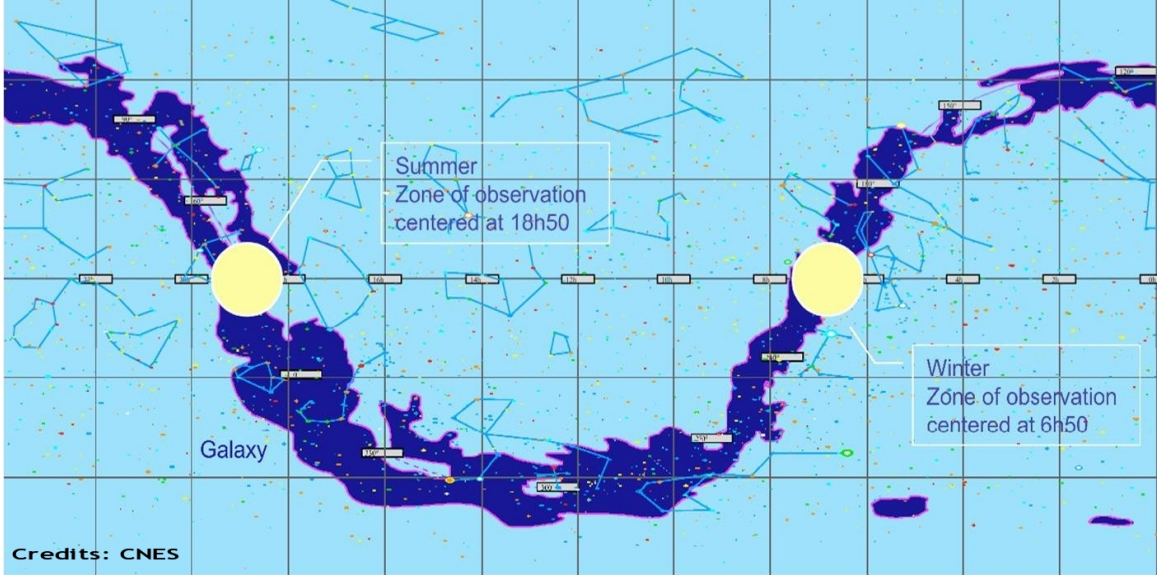


Figure 1.18: Corot "Eyes"

A detailed list of the coordinates of the already observed fields from [9] can be found in table 1.5,

Table 1.5: Coordinates of the observed long run fields

Run	Date start-end	duration [d]	RA	DEC
LRc01	05 2007 - 09 2007	152	19:23	00:27
LRa01	10 2007 - 02 2008	150	06:47	-00:12
LRc02	04 2008 - 09 2008	150	19:00	-03:20

For parallax measurements its very important to get a very long baseline between the two positions of the observer. The optimal time difference between two observations from earth of the same field would be close to half a year. When we take a look at the planned and now completed observational runs of CoRoT, we see that with an observational time of 150 days for the long runs COROT gives an ideal platform to derive parallaxes if the astrometric position of stars can be derived with sufficient accuracy. The short runs have only observational time intervals of around 30 days.

CHAPTER 1. INTRODUCTION

An overview of the position of the “eyes” directed to the galactic center LRc01 and to the galactic anticenter LRa01 can be seen in figure 1.18. A detailed view of the two fields is displayed in table 1.6. The circle shows the fields were CoRoT could point at. The small frame “Field of View” shows the place of the actual field. More details about the fields can be found in chapter 4.

The coordinates therein are the coordinates of the line of sight. To get the coordinates of the observed field its important to know the exact position of the corners of each of the four CCD’s. These coordinates can be found in the appendix.

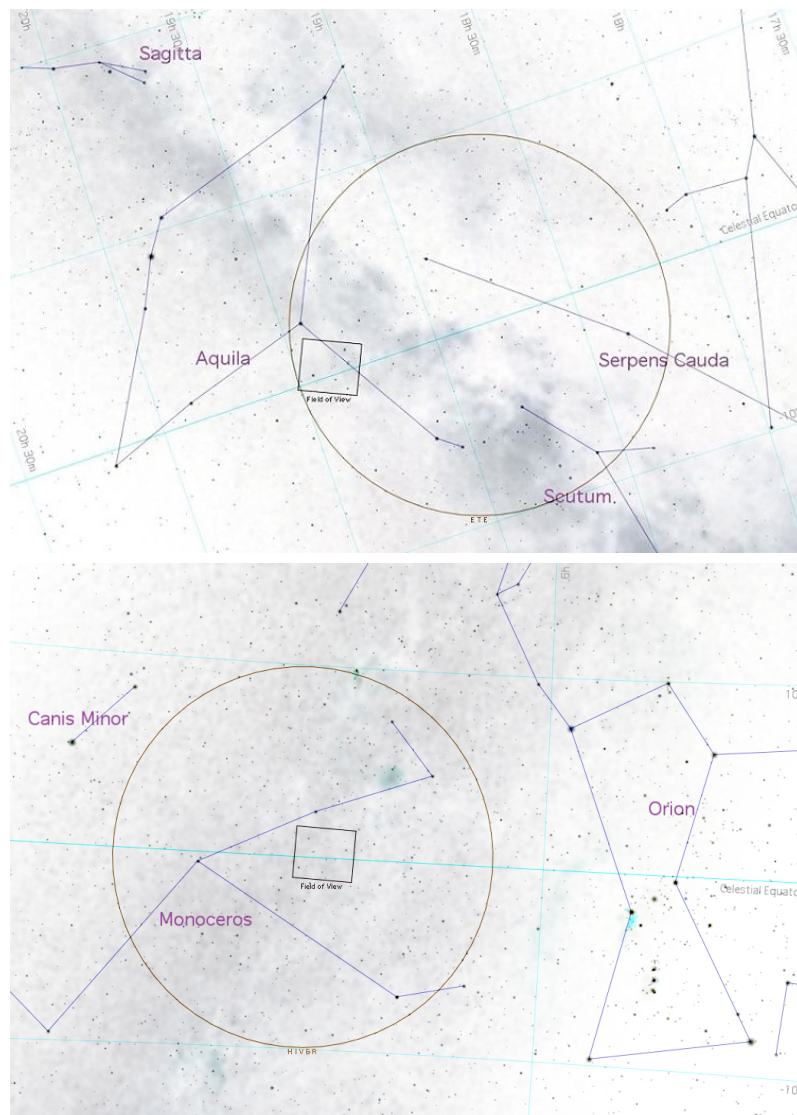


Table 1.6: CoRoT “Eyes” pointing to the galactic center (LRc01,top) and to the galactic anticenter (LRa01,bottom)

1.3.3 Why in space?

The problem of long time observations on earth is the limitation to 8 to 12 hours. Space observatories like [26] or CoRoT allow the measurement of light curves with a much longer observation time than on earth.

Even if networks are used and the stars are observed from different observatories, there is still a loss of accuracy caused by different instruments, different weather conditions and geographical factors. Due to the rotation period of the earth every light curve is biased by a 24-hours period, this period is not existing for space observations, here its the orbit period which can cause systematical errors in the flux measurement (straylight from earth, sun, moon).

1.3.4 What are the demands to the instrumentation?

A very high accuracy of the flux measurement

The idea to meet this demand is to use a defocused Point Spread Function. This means that the light from the observed star is spread over multiple pixels, the intensity change is measured on many pixels. First the fluctuations in the flux should have the same period for all pixels, so we reach a high accuracy in the measurement of light fluctuations.

Secondly we have a higher accuracy in the measurement of the whole flux from that star because we can distribute the light over more pixels. We can have a higher count number if only one pixel is limited with "Full Well Capacity". A higher number of counts allows a more exact evaluation of the whole amount of flux. Also the pixel is not saturated as fast as it would have been, if the star was completely focused on this pixel. This is useful for relatively bright stars.

Low straylight from earth, moon, etc. This is done by an especially constructed baffle. To reach the required straylight level it would have been inevitable to build a very long baffle, but in space flight, very "long" components are much more complicated to handle and therefore expensive. Small, compact components are preferred and cheaper. The solution was to build a folded optical system where the baffle and the optics could fit onto a standard ESA PROTHEUS platform [43].

This special design is also the main reason for the asymmetrical shape of a star, the so called Point Spread Function, which will be of interest for the further study.

1.3.5 How can the pulsations be classified?

There are two different CCD fields in the Field of View (FOV), in one part (the one we are also working with), the so called Seismological Field, there is just a simple defocused image of the star field. In front of the second field, the so called Exoplanetary field, a prism is installed, allowing to obtain a color image of the brighter stars. It is thus possible to distinguish stellar activity from a planetary transit, because flux changes caused by a

transit show the same relative level over the whole color spectrum unlike flux changes caused by flares, eruptions or solar spots.

1.3.6 What are the main scientific goals of CoRoT and what is the question the author is working on?

Classification of star pulsations to distinguish pulsation caused by stellar eruptions and by convection on one hand and to derive rotation periods of the observed stars were the main scientific goals of CoRoT. As a very fascinating feature the team hoped to find extrasolar planets by flux changes caused by their planetary transits in front of their host star. As an additional feature the question came up whether it was possible to derive the positions of at least two stars in the field of view of CoRoT with sufficient precision to measure parallaxes? Can these measurements even be useful to allocate other measurements from Satellites like Hipparcos?

1.3.7 The Optical System

One of the great challenges the CoRoT instrument team had to face was the extremely low amount of stray light that was allowed to enter the pupil of the optical system. The required photometric sensitivity to measure intensity variations of the observed stars was set to 10^{-6} . That corresponds to a maximum stray light of 1 photon per second and pixel. Already from the earth, assuming an angle between earth's limb and the line of sight of 20° , stray light level would amount to 6×10^{12} photons per pixel and second. So the baffle must have a rejection capability of 10^{12} .

Normally for that kind of telescope an anastigmatic¹ optical system is favored, but this would have led to a baffle length of around 5 meters. A baffle of that size would cause many structural problems and would have increased the cost of the mission extremely.

The only way to build an optical system with such a long focal length was by modifying the design and by the use of lenses.

A non-symmetrical telescope with two mirrors folding the optical way (see figure 1.20) followed by a system of lenses was chosen. The folding was important in order that the telescope and the baffle fit onto a PROTHEUS platform and at the same time the baffle had enough stray light rejection capacity to fulfill the constraints.

The final optical design consists of a system of two parabolic mirrors, reducing the entrance pupil of 270 mm ([7]) by a factor of 3, followed by 6 dioptical lenses scaling the system to a focal length of 1.1 m.

The length of the telescope was in that way reduced and the stray light rejection capacity was even better then for the initially proposed TMA (Three-Mirror Anastigmatic, figure 1.19 and [43]) concept.

This special construction also leads to a very special image of stars in the field of view as we see in appendix B.

Image field Image Field Size: $2.8 \times 2.8^\circ$, aperture ratio: $f/3$. One half of the field is used for the "Seismology" and one half is used for the "Exoplanet Mission".

¹anastigmatic = not astigmatic, astigmatism: defect caused by the deviation of the lens surface from a spherical curvature, light beams are therefore not converging in one point of focus

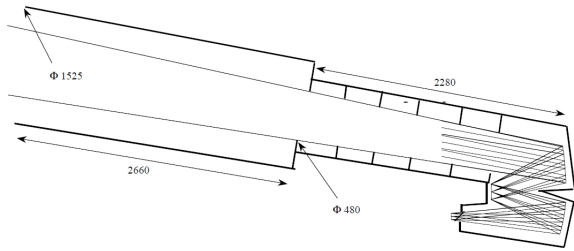


Figure 1.19: Preliminary Three Mirror Anastigmatic optical concept [43]

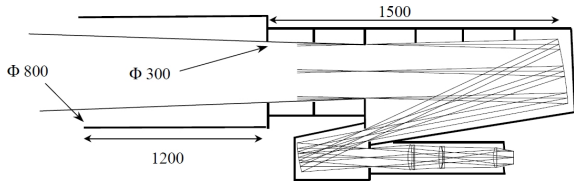


Figure 1.20: Final CoRoT-Optical System using two mirrors and 6 dioptical lenses [43]

1.3.8 The point spread function (PSF) of the Seismological Channel

The simulation of the optical system was performed by the French CoRoT team around Michel Auvergne, using the ZEMAX [52] optical system design software. To simulate the flux coming from a star of a certain temperature the Kurucz Stellar Atmospheres model code Atlas 9 [24](Kurucz, 1970, 1993) was used.

The PSF is especially designed for a high precision flux detection (see 1.3.4), so the illuminated area spreads over 10×13 "real" pixels. To reduce the risk of over-saturation the PSF is shaped like a small volcano (see figure 1.21), where the pixels inside the cone all have a similar intensity. One idea behind this shape is to have a good signal to noise ratio on all illuminated pixels. This is done by a PSF where the intensity on all illuminated pixels is in the same range. Like this no pixels are already overilluminated while others are not getting sufficient signal strength to reach a good signal to noise ratio.

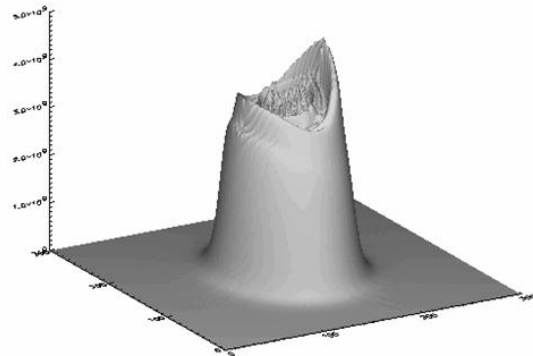


Figure 1.21: CoRoT PSF [43]

The second idea is to have a large number of pixels where the flux can be measured. A large number of pixels reduces the amount of error for the measurement of flux change compared to a single detector. All these pixels are exposed to a similar amount of light during one exposure time so all pixels reach a good signal to noise ratio. The number of under- or overexposed pixels is extremely reduced by that shape.

1.3.9 The CCDs

The CCDs used for the CoRoT mission are 42-80 back-illuminated Charge Coupled Devices from e2v technologies (former "Marconi Applied Technologies"). The 42-80 CCDs are especially useful for astronomical and scientific applications due to their low noise amplifiers and readout features which allow the reading of large charge packets. It also is designed for the construction of mosaic fields of more CCDs ([55]).

The optical detector consists of four CCD's in the optical plane. Two CCDs are used for the seismological program and two are used for the exoplanetary program. Each CCD has a size of 2048 x



Figure 1.22: CCD 42-80, back illuminated CCD from e2v technologies

1.3. THE COROT MISSION

2048 px. An impression of the arrangement of the four CCDs in the field of view is given in figure 4.1. The size of one pixel on the CCD corresponds to 2.5 " :

$$1,4^\circ = 60' + 24' = 84 * 60'' = 5040'' \sim 2000px$$

$$\Rightarrow 2px \sim 5'' \Leftrightarrow 2/5px \sim 1'' \Rightarrow 1masec \sim 0.0004px$$

. That means to reach an astrometrical accuracy comparable with the accuracy of HIP-PARCOS of 1 masec, we need to reach an accuracy of our position finding algorithm of 0.0004 px. Because our simulations of the PSF is four times oversampled, an accuracy of 1 px in the simulation corresponds to an accuracy of 0.25 px on the real CCD. That means we would need to reach an accuracy of **0.0016 px** for our simulated positional algorithms with our oversampled 160 x 160 px PSF.

1.3.10 The simulated data:

The simulations which were received from the French CoRoT Team around Michel Auger are consisting of sixteen point spread functions of the seismological channel.

They are simulations of the starlight from four stars with temperatures and therefore colors of 4000 K, 5000 K, 6000 K and 8000 K. For each star and temperature four different position angles -0.012, 0.662, 1.337 and 2.012 have been calculated. The overall intensity per simulated PSF has been set to 1, so the strongest illuminated pixel has an intensity of 0.00069.

These four positions² are lying along a line with increasing distance to the line of sight.

Temperatures	Positions
4000 K	-0.012
5000 K	0.662
6000 K	1.337
8000 K	2.012

Table 1.7: The positions of the simulated PSFs

If we compare the shape of the PSF (figure 1.23) we see from left to right the influence of the coma caused by the different distances to the line of sight, and from top to

²Naming Convention: "positions" and "coordinates" in the running text are describing derived position values for a star within a simulated frame, to describe different positions of the simulated frame this author uses the term "position angle".

bottom the influence of the chromatic aberration caused by the different temperatures and therefore colors of the simulated stars. We can already see that coma causes a stronger change in the shape of the PSF then the chromatic aberration.

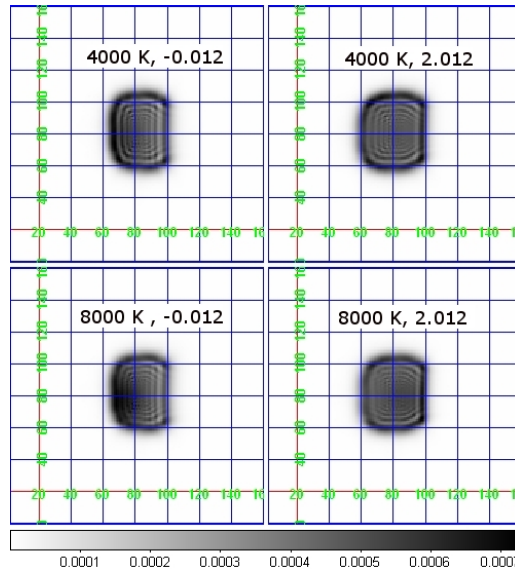


Figure 1.23: COROT point spread functions between 4000K and 8000K at angular positions -0.012 and 2.012

Filelist

The received files contain informations about the effective temperature and the position in the FoV which can be taken from the file names.

The name of the file contains:

- the position of the PSF in the Field of View measured in angular distance to the Line of Sight:
 $(-0.420/-0.012), (-0.420/0.0662), (-0.420/1.337)$ and $(-0.420/2.012)$
- the effective temperature of the target : 4000 K, 5000 K, 6000 K and 8000 K

Listing 1.1: filelist

```

-0.420_-0.012_4000.fits  -0.420_-0.012_5000.fits
-0.420_-0.012_6000.fits  -0.420_-0.012_8000.fits
-0.420_0.662_4000.fits  -0.420_0.662_5000.fits
-0.420_0.662_6000.fits  -0.420_0.662_8000.fits
-0.420_1.337_4000.fits  -0.420_1.337_5000.fits
-0.420_1.337_6000.fits  -0.420_1.337_8000.fits
-0.420_2.012_4000.fits  -0.420_2.012_5000.fits
-0.420_2.012_6000.fits  -0.420_2.012_8000.fits
    
```

1.3. THE COROT MISSION

The simulated frames are all positioned on a line with increasing distance to the line of sight.

Fits-Header

Each file contains a header file containing the following information:

Listing 1.2: Fits-header for simulated PSFs

```
SIMPLE = T
/ Written by IDL: Fri Apr 18 11:14:16 2003
BITPIX = -32 /
NAXIS = 2 /
NAXIS1 = 160 /
NAXIS2 = 160 /
DATE = '2003-03-06'
/ Creation date (CCYY-MM-DD) of FITS header
VISEE = '1.0460 microns at -0.4200, 0.6620 deg.' /
SPACING = 3.37500 / Data grid spacing
- microns
XAREA = 536.600000000 /
YAREA = 536.600000000 /
PUPIL = 'Pupil grid size: 256 by 256' /
IMAGE = 'Image grid size: 160 by 160' /
PXCENTER= 81
/ Image center - pixel unit
PYCENTER= 81
/ Image center - pixel unit
X_CENTER= 8.79474
/Origin of the monochromatic PSF at 650 nm
Y_CENTER= 2.24164
/Origin of the monochromatic PSF at 650 nm
DEFOC = 'f-0.8: ' /Defocus (mm)
TYPE = 'white '
/Values: white or monochromatic
VERSION = 17
/ version number of the optic
TRANS = 16.2000
/ transmission: version number of the optic
FLUX = 'Kurucz Atlas 9' /target Flux origin
QE = 'E2V '
/Quantum efficiency: measurements coming from
O_SOFT = 'Zemax '
/ Optic simulation software
```

CHAPTER 1. INTRODUCTION

```
T_TEMP = 20.0000
        /Telescope and OD temperature
ATM_V = 'Vacuum ' /Optics in vacuum
STAR_T = 4000.00
        /Target effective temperature
POS = '-0.420_0.662'
        /angular position in the sky
XIMIN = 8.52305 /xmin of the window (mm)
XIMAX = 9.06305 /xmax of the window (mm)
YIMIN = 1.96995 /ymin of the window (mm)
YIMAX = 2.50995 /ymax of the window (mm)
DX = 3.37500 /x step (micron)
DY = 3.37500 /y step (micron)
END
```


1.3. *THE COROT MISSION*

Chapter 2

Tests with all PSFs

2.1 First thoughts about optical effects and centroiding algorithms

To describe the behavior of the COROT optics for changing stars it was tried to find some trends in the distribution of the positions derived with different algorithms for all sixteen PSFs. The main goal is to find out, which algorithm delivers the best results for astrometry and should therefore be used for astrometric reductions of COROT fields.

2.1.1 Effects changing the shape of the PSF

Chromatic aberration:

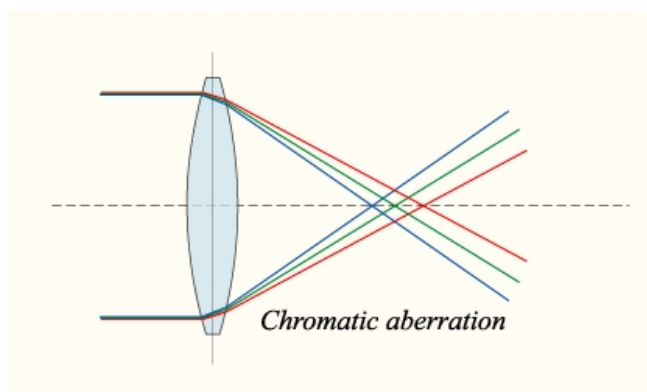


Figure 2.1: Chromatic aberration [53]

Stars emit light with a typical wavelength distribution, this corresponds to colors. With different temperatures the color of the stars change. Lenses have a different refraction index depending on the wavelength of the light passing through the lens. Therefore stars with a different temperature can be found on a slightly different position on the optical plane. The strength of this effect is depending of the optics used and on the centroiding algorithm used.

2.1. FIRST THOUGHTS ABOUT OPTICAL EFFECTS AND CENTROIDING ALGORITHMS

Coma and distortion effects:

Another effect influencing the position of a star or frame positions in the Field of View (FoV) is coma or comatic aberration.

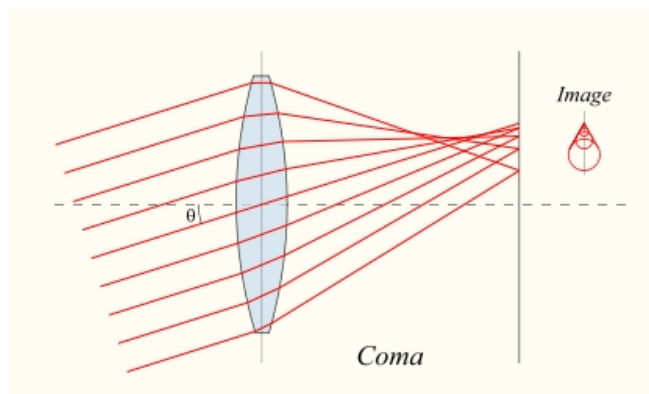


Figure 2.2: Coma [54]

Light beams coming from a source not lying directly in the line of sight focus on a different point than the focal point. This effect is usually described for parabolic mirrors or lenses. The further away from the optical axis a light source is positioned the stronger is this effect. In our case we do not have a simple "one mirror" optics, we have a more complicated optical system, with lenses and mirrors, but the change of the resulting position

of the centroid of a PSF caused by the different location relative to the optical axis can also be called comatic aberration.

So we have two effects which can be examined, the chromatic aberration and the coma effect. Both effects are influenced by the asymmetrical optical system which leads to a systematic distortion for different positions of a star in the Field of View.

2.1.2 ESO-MIDAS Centroid Search Routines:

ESO-MIDAS¹ (European Southern Observatory - Munich Image Data Analysis System) offers three methods to derive coordinates of stars from a star field.

To create an overview of the behavior of the 16 PSF's the positions of our stars (x,y) have been derived with three methods:

¹Trademark of the European Southern Observatory. The ESO-MIDAS system provides general tools for image processing and data reduction with emphasis on astronomical applications including imaging and special reduction packages for the ESO instrumentation at La Silla and the VLT at Paranal. In addition it contains applications packages for stellar and surface photometry, image sharpening and decomposition, statistics and various others.

The official name, ESO-MIDAS, is a registered trademark. ESO-MIDAS is available under the GNU General Public License (GPL), and can be implemented on OpenVMS and UNIX (Linux) systems. For more detailed information about ESO-MIDAS, the reader is referred to the following website: <http://www.eso.org/sci/data-processing/software/esomididas/>

Center/Moment: Compute the central position using an intensity weighted first moment (mean), the so called barycenter² of the pixel values.

By replacing mass with the pixel value (Intensity) $I_{(i,j)}$ and position(x,y) with pixel row i and column j this method delivers an easy way to calculate positions of stars on digital images.

$$x_c = \frac{\sum_{i,j} I_{(i,j)} * i}{\sum_{i,j} I_{(i,j)}}; \quad (2.1)$$

$$y_c = \frac{\sum_{i,j} I_{(i,j)} * j}{\sum_{i,j} I_{(i,j)}} \quad (2.2)$$

This simple Centroid algorithm is especially useful for blobs. [51]

Center/IQE: Compute the central position fitting a 2-dim Gaussian distribution to the marginal distributions in both the x- and y-directions. This method also returns the axis and position angle and is especially useful for elongated stars.

Center/GAUSS: Compute the central position of the specified object by fitting a Gaussian distribution to the marginal distributions in both the x- and y-directions (or just in x). The algorithm delivers also the FWHM (the full width at half maximum, the width of the Gaussian curve where the intensity is half of the maximum intensity) for each star.

2.1.3 Positions derived with different methods

The results of the different algorithms for the sixteen point spread functions are shown in table 2.1 and plotted in figure in 2.3. The main property of the three methods is seen by observing the plots and shows that the centroiding algorithms react differently to the change of the PSF properties. The distribution of the results for the Center/Gauss (Δ) algorithm compared to the results for the two other methods is clearly visible and leads to the impression that the Center/Gauss method is inaccurate in comparison with the two other methods.

In plot 2.4 only the results for the Center/IQE and the Center/Moment algorithms without the widespread Center/Gauss results are displayed. Here we observe that the Center/IQE (*) and Center/Moment (+) results have a similar distribution range but slightly different position values. Also we can see that the two methods react differently to the change of the shape of the PSF caused by a different star temperature and different position in the Field of View. This effect is worked out in greater detail in the following section.

If we take a closer look at the distribution of the results for Center/Moment (figure 2.5) and Center/IQE (figure 2.6) we see that the results can be assigned to different groups and there seems to be a trend with increasing star temperature and with increasing angular

²The Barycenter or center of mass \tilde{R} of a system of particles is defined as the average of their positions \vec{r}_i weighted by their masses m_i : $\tilde{R} = \frac{\sum m_i \vec{r}_i}{\sum m_i}$

2.1. FIRST THOUGHTS ABOUT OPTICAL EFFECTS AND CENTROIDDING ALGORITHMS

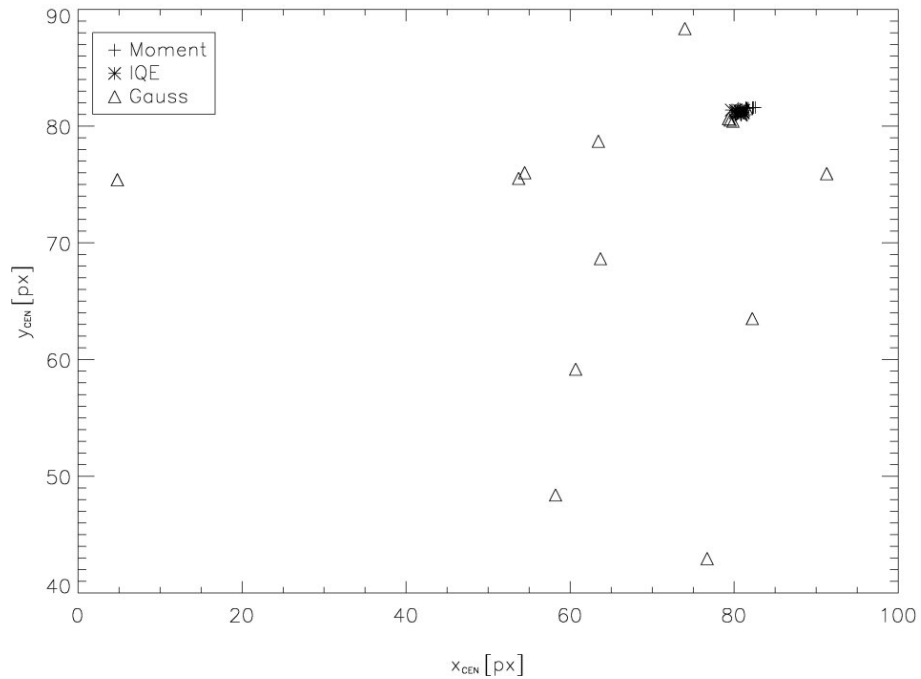


Figure 2.3: Coordinates for all three MIDAS algorithms

distance to the line of sight. This means that the physical effects of chromatic aberration and coma can be displayed with these algorithms. To display the change in the shape of the PSF also all 16 simulated PSF's were included (see page 103). It can be seen that the shape of the PSF changes slightly with temperature and position in the field of view.

CHAPTER 2. TESTS WITH ALL PSFS

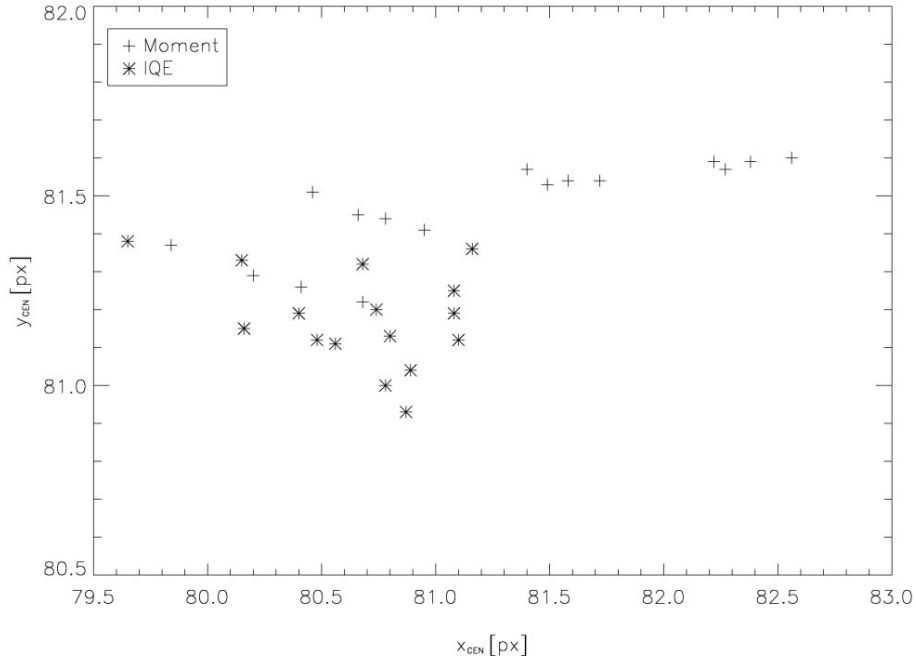


Figure 2.4: Coordinates for Center/Moment and Center/IQE

Method		Center/Moment		Center/IQE		Center/Gauss	
Angle	T	X_{CEN}	Y_{CEN}	X_{CEN}	Y_{CEN}	X_{CEN}	Y_{CEN}
-0,012	4000	82,22	81,59	81,16	81,36	81,5	81,5
0,662	4000	81,4	81,57	80,68	81,32	63,69	68,64
1,337	4000	80,46	81,51	80,15	81,33	81,5	81,5
2,012	4000	79,84	81,37	79,65	81,38	76,7	42,94
-0,012	5000	82,27	81,57	81,08	81,25	53,72	75,52
0,662	5000	81,49	81,53	80,74	81,2	58,22	48,39
1,337	5000	80,66	81,45	80,4	81,19	79,31	80,65
2,012	5000	80,2	81,29	80,16	81,15	82,21	63,51
-0,012	6000	82,38	81,59	81,08	81,19	54,46	76,01
0,662	6000	81,58	81,54	80,8	81,13	60,67	59,17
1,337	6000	80,78	81,44	80,56	81,11	79,51	80,52
2,012	6000	80,41	81,26	80,48	81,12	73,99	88,36
-0,012	8000	82,56	81,6	81,1	81,12	4,8	75,41
0,662	8000	81,72	81,54	80,89	81,04	63,44	78,71
1,337	8000	80,95	81,41	80,78	81	79,85	80,42
2,012	8000	80,68	81,22	80,87	80,93	91,27	75,93

Table 2.1: Coordinates (x,y) derived with MIDAS Center/MOMENT, CENTER/IQE and CENTER/GAUSS (see chapter 2.1.2)

2.2 Tests of the accuracy of the MIDAS methods

The main goal is to test the accuracy of the centroiding algorithms.

As a first method, it was assumed that there is a good "mean" value for the position of a star with a certain position on the CCD with different temperatures. The effect of chromatic aberration due to the changing temperature is ignored. The same is done again with "mean" values for each temperature and calculating this "mean" over all angular positions. Here the effect of the coma is ignored. This test delivers the values for the accuracy of the three measuring methods. The quantity that is showing the quality of a method is the variance of the mean value of the centroid positions derived with the different algorithms.

The second test is: how good can the physical behavior of the PSFs be modeled by a linear and a polynomial fit (Bestimmtheitsmass). The third quantity that is tested is the standard deviation of those physical behaviors with a linear regression.

2.2.1 Grouping and trends

If we try to group the positions found with the different methods it is obvious that the methods deliver different results, but the shifting caused by the two effects also has a different strength and the direction of the shift is not the same. From the physical point of view the shifts are caused by the same effect and therefore the resulting shift value should be the same, independently of the used algorithm to derive a position.

The difference in the shifts between the used methods allows us to judge the quality of the two methods. For the third positional algorithm, the CENTER/GAUSS algorithm, no trend can be found because the method is too inaccurate.

Center/Moment results

In the different plots we see the results for the Center/Midas algorithm grouped by temperature and by the position angle of the PSF. It is already visible that there is a trend, and also that the change in the temperature leads to a stronger deviation than a change in the position. It seems that an increase in the temperature leads to shift into the positive x and negative y direction. A change in the position from -0.012 to 2.012 in the FoV leads to a shift into the negative x and negative y direction.

When we compute a mean position and derive the error of this mean position we can use this to say something about the relative strength of the two effects causing the position to change. In the first plot 2.6(a) we have the results grouped by constant temperatures. In plot 2.6(b) we have the results for constant positions.

$\sigma(x)$ is ranging for a constant temperature from 0.6 to 0.49, while $\sigma(y)$ ranges from 0.06 to 0.1. For a constant position in the FoV the resulting $\sigma(x)$ is ranging from 0.09 to 0.2 and $\sigma(y)$ from 0.01 to 0.03. We see that the coma effect leads to a bigger error in the

Figure 2.5: Coordinates derived with Center/Moment and grouped by position angle and temperature

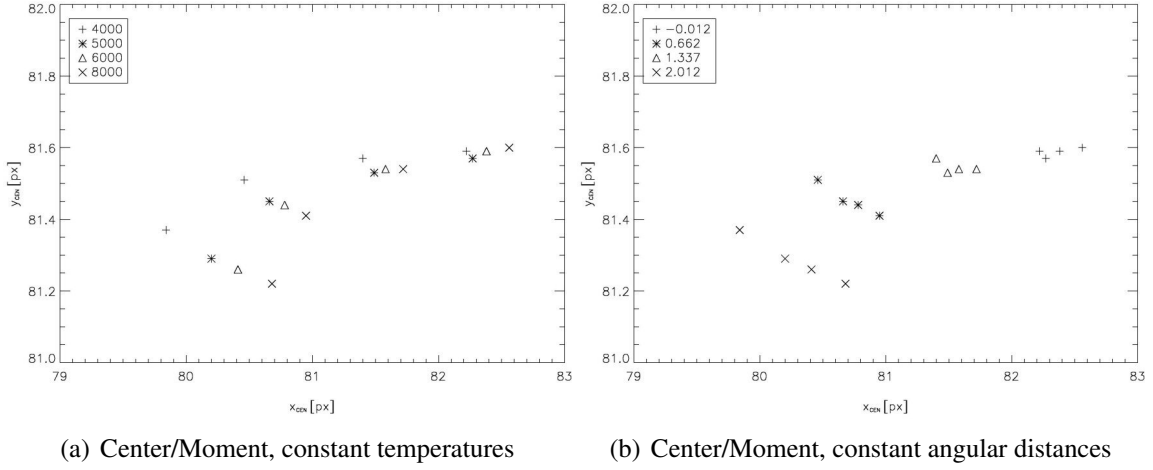


Table 2.2: Center/Moment, const. temperatures

Temp.	\bar{X}_{CEN}	\bar{Y}_{CEN}	$\sigma(x)$	$\sigma(y)$
4000	80,98	81,51	0,6	0,06
5000	81,16	81,46	0,53	0,07
6000	81,29	81,46	0,51	0,08
8000	81,48	81,44	0,49	0,1

Table 2.3: Center/Moment, const. angular distances

Pos.	\bar{X}_{CEN}	\bar{Y}_{CEN}	$\sigma(x)$	$\sigma(y)$
-0,012	82,36	81,59	0,09	0,01
0,662	81,55	81,55	0,08	0,01
1,337	80,71	81,45	0,12	0,02
2,012	80,28	81,29	0,2	0,03

mean position then the effect caused by a variable temperature. Also the error is larger in x direction than in y (see the scaling from 81.0 to 82.0 in x and 79.0 to 83.0 in y).

Also the mean variance of all four mean temperatures $\bar{\sigma}_{temp}(x, y) = (0.53, 0.08)$ and four mean positions $\bar{\sigma}_{pos}(x, y) = (0.12, 0.01)$ has been calculated and shows that the effect of coma is more important then the effect of chromatic aberration by a factor of 5.

Center/IQE results:

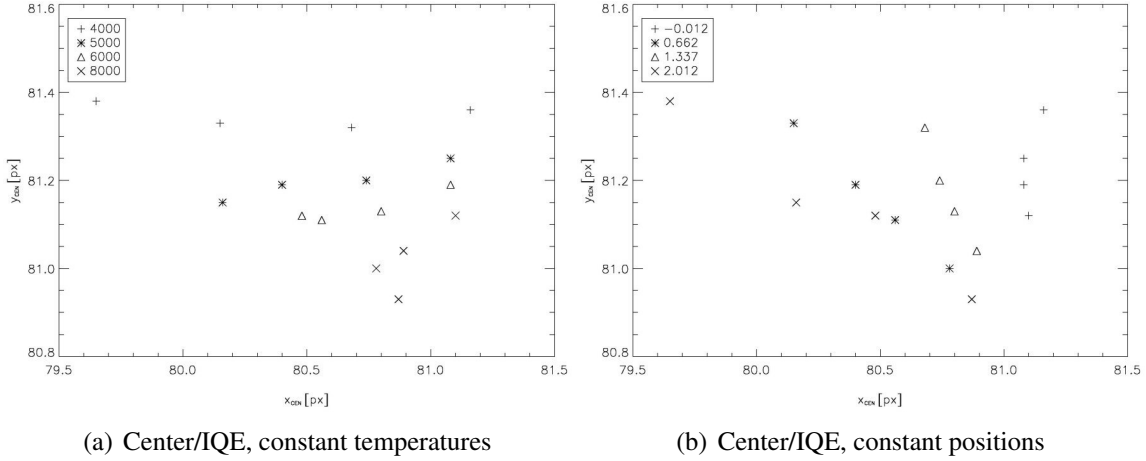
The same plots for the results from the Center/IQE algorithm are shown in figure 2.7(a) and figure 2.7(b). We see that this algorithm reacts differently to the change of the PSF and also the trend caused by a change in temperature or a change of the position in the optical field points to a different direction.

Again we have a trend with a different strength and direction for changing temperature and position. But the direction of the position change is not as clearly seen as in the Center/Moment plots. Here we see that an increasing temperature leads to a shift to the negative y direction. A change in the position in the FoV seems to lead to a shift of the Centroid to the positive x direction. These trends will be discussed in detail later.

The sigma values for mean positions for constant temperatures for the Center/IQE method lie between 0.38 and 0.08 in x and between 0.02 and 0.05 in y. For constant

2.2. TESTS OF THE ACCURACY OF THE MIDAS METHODS

Figure 2.6: Coordinates derived with Center/IQE and grouped by position angle and temperature



positions the variance in x lies between 0.02 and 0.3, in y between 0.04 and 0.08. Again we can at least say that in x direction we have a bigger shift then in y direction.

Table 2.4: Center/IQE, const. temperatures

Temp.	\overline{X}_{CEN}	\overline{Y}_{CEN}	$\sigma(x)$	$\sigma(y)$
4000	80,41	81,35	0,38	0,02
5000	80,6	81,2	0,23	0,02
6000	80,73	81,14	0,16	0,02
8000	80,91	81,02	0,08	0,05

Table 2.5: Center/IQE, const. angular distances

Pos.	\overline{X}_{CEN}	\overline{Y}_{CEN}	$\sigma(x)$	$\sigma(y)$
-0,012	81,11	81,23	0,02	0,04
0,662	80,47	81,16	0,15	0,06
1,337	80,78	81,17	0,05	0,05
2,012	80,29	81,15	0,3	0,08

If we observe the mean variance for the four constant temperatures $\overline{\sigma}_{temp}(x, y) = (0.21, 0.03)$ and compare it to the mean sigma for the constant positions $\overline{\sigma}_{pos}(x, y) = (0.13, 0.06)$ we see that no obvious difference between the two effects of chromatic and comatic aberration can be found.

Center/Gauss results:

If we plot the very widespread results of the Center/Gauss method we see that its very difficult to find a trend.

Also observing the sigmas of mean positions for constant temperatures we have variations ranging from 4.86 to 22.19 in x and from 1.36 to 10.49 for constant positions we have similar error values ranging from 0.58 to 18.44 in x and from 0.29 to 11.18 in y. So no trend can be found because the method is too inaccurate to find any significant physical trend. An overview of the mean sigmas for constant temperatures and constant positions ($\overline{\sigma}_{temp}(x, y) = (10.52, 6.82)$, $\overline{\sigma}_{pos}(x, y) = (6.23, 5.16)$) shows that the method is very inaccurate in comparison to the Center/Moment and the Center/IQE algorithm.

Figure 2.7: Coordinates derived with Center/Gauss grouped by position angle and temperature

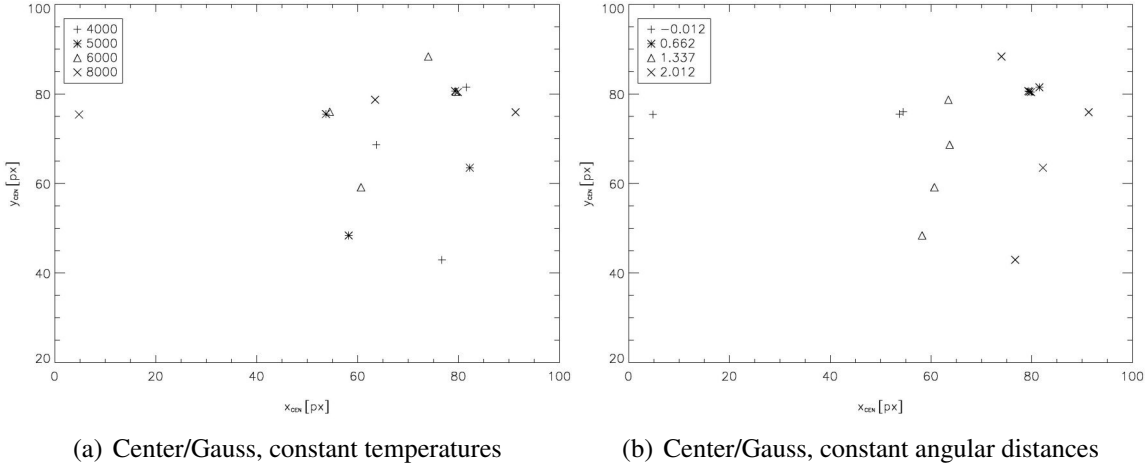


Table 2.6: Center/Gauss, const. temperatures

Temp.	\bar{X}_{CEN}	\bar{Y}_{CEN}	$\sigma(x)$	$\sigma(y)$
4000	75,85	68,65	4,86	10,49
5000	68,37	67,02	8,36	8,28
6000	67,16	76,02	6,69	7,12
8000	59,84	77,62	22,19	1,36

Table 2.7: Center/Gauss, const. angular distances

Pos.	\bar{X}_{CEN}	\bar{Y}_{CEN}	$\sigma(x)$	$\sigma(y)$
-0,012	48,62	77,11	18,44	1,7
0,662	61,51	63,73	1,49	7,49
1,337	80,04	80,77	0,58	0,29
2,012	81,04	67,69	4,4	11,18

Conclusion Both effects, chromatic aberration caused by changing temperature of the simulated star and comatic aberration caused by changing position of the star in the field of view, should not directly disturb the measurement of parallaxes for a certain star because neither the position of an object on the CCD nor the temperature of the Object should change significantly during the observation time in a CoRoT field. Because the effects are known and can be simulated they can show us how accurate our positional algorithm can be. If we know for instance that the PSF's with a temperature of 4000 K, 5000 K, 6000 K and 8000 K should be lying on a line and the increase in temperature should cause a shift in a certain direction by a certain amount, then we can test the accuracy of our measured position relative to this line. This accuracy allows us to judge the quality of a certain centroiding algorithm. If we try to find the effects of coma and chromatic aberration with the Center/Gauss Method no obvious trend can be found because the derived positions are much too inaccurate to show this effect clearly. We can see that the Center/Moment algorithm and the Center/IQE algorithm show very similar error ranges.

Mean variances If observe the mean variance for a physical parameter and a certain algorithm e.q. angular distance and Center/Moment (table 2.3) and compute the mean of the $\sigma(x)$ values (in this example 0.09, 0.08, 0.12 and 0.2) we get a mean sigma $\bar{\sigma}(x)$ (in

2.2. TESTS OF THE ACCURACY OF THE MIDAS METHODS

the example Center/Moment and constant angular distance we get $\bar{\sigma}(\bar{x}) = 0.12$) for this parameter and this algorithm.

Table 2.8: Mean variances for constant angular distances

	$\bar{\sigma}(\bar{x})$	$\bar{\sigma}(\bar{y})$	
Center/Moment	0,12	0,01	x
Center/IQE	0,13	0,06	
Center/Gauss	6,23	5,16	

Table 2.9: Mean variances for constant temperature constant

	$\bar{\sigma}(\bar{x})$	$\bar{\sigma}(\bar{y})$	
Center/Moment	0,53	0,08	
Center/IQE	0,21	0,03	x
Center/Gauss	10,52	6,82	

These mean variances can be computed for all three methods and for the two physical parameters temperature of the star (see table 2.9) and position in the field of view (see table 2.8). We find that two different parameters favor different methods. The Center/Moment algorithm is the better method (x) to fit the results of a changing angular distance and the method Center/IQE is better to model the results from a change in star temperature. It seems that both methods reach approximately the same accuracy. Only the results for the Center/Gauss algorithm are not as good as the Center/IQE and the Center/Moment algorithm. The main result is that there is no simple criterion to show whether the Center/Moment or Center/IQE algorithm is better to fit positions of stars in the CoRoT field, but both reach approximately the same mean variance of around 0.01 to 0.53 pixels. This means we can assume that the accuracy of the algorithms is of the magnitude of about 0.1 px in the real field (the simulation used in this test was four times oversampled), corresponding to a theoretical accuracy of one milliarcsecond.

2.2.2 Coefficient of determination

To find a numerical factor to judge the quality of the centroiding algorithms, we assume that there is a physical model for the deviation caused by chromatic distortion and by the coma effect, the data points for one temperature and four angular positions are plotted and it is tried to find a relation between the four data points. The regression function could be linear or polynomial (in reality the relation should be a $x = \tan(y)$ curve for a spherical lens system).

The value that shows the quality of a regression function and therefore the value to judge the quality of the centroiding algorithm was now the coefficient of determination of those regression functions.

Linear Models:

A linear model of the form $y = mx + b$ is used to fit the data. We see that the shift of the position of the PSF in the Field of View causes the centroid position to move relatively constant to the positive x direction.

Constant Temperatures Also we see that the increase in temperature leads to a shift in the positive x direction, but the method Center/IQE also shows a significant shift to the negative y direction.

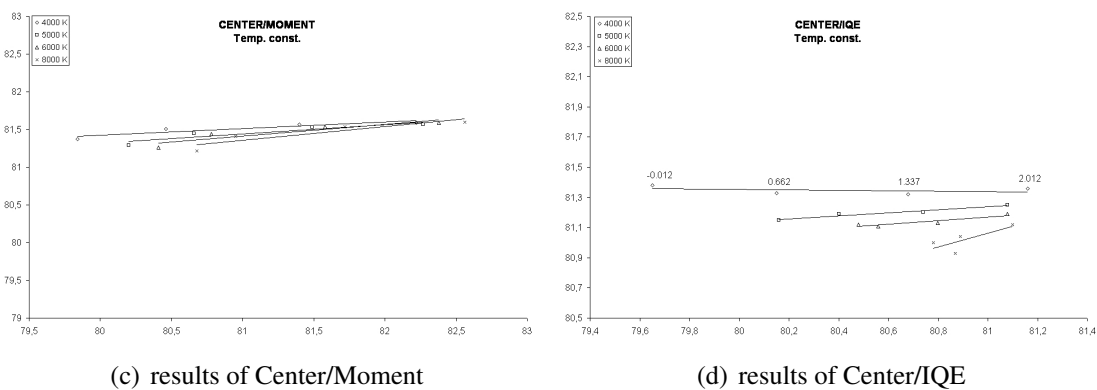


Figure 2.8: Linear models fitted for constant temperatures

The slopes of the fitted models lie between 0.09 and 0.18 with y-interception points between 66.7 and 74,5 for the Center/Moment results (table 2.10) and between -0.01 and 0,45 with y-interception points between 44.5 and 82.5 (table 2.11). This gives us already an idea about the quality of the method relative to changes in the position of the PSF in the FoV. The fact that the slope does vary less for the Center/Moment method then for the Center/IQE method could have a better accuracy then the Center/IQE method because less variation in the fitted physical model is observed.

2.2. TESTS OF THE ACCURACY OF THE MIDAS METHODS

Table 2.10: Center/Moment linear model parameters for constant temperatures

Temp.	slope m	y-intercept b
4000 K	0,0865	74,51
5000 K	0,1247	71,34
6000 K	0,1522	69,09
8000 K	0,1812	66,68

Table 2.11: Center/IQE linear model parameters for constant temperatures

Temp.	slope m	y-intercept b
4000 K	-0,0142	82,49
5000 K	0,0992	73,205
6000 K	0,1229	71,212
8000 K	0,4509	44,539

Constant angular positions When we observe the fitted models for constant angular positions we see again that the Center/Moment method shows less variations in the slope which varies only between 0.05 and -0.19, while for the Center/IQE method the slope varies between -0.35 and -1.2. Again its obvious that the methods react differently to the change in the physical properties of the PSF.

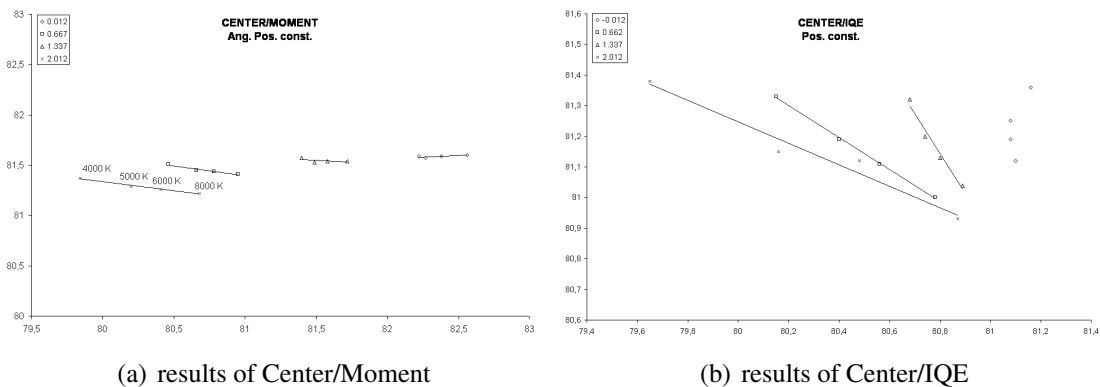


Figure 2.9: linear models fitted for constant angular positions

³Result canceled due to non-bijective data points

CHAPTER 2. TESTS WITH ALL PSFS

Table 2.12: Center/Moment linear model parameters for constant angular positions

angular distance	gradient	intersection
-0,012	0,0555	77,02
1,337	-0,1987	97,487
0,662	-0,0689	87,164
2,012	-0,1779	95,568

Table 2.13: Center/IQE linear model parameters for constant angular positions

angular distance	gradient	intersection
-0,012	1,907	-73,435 ³
1,337	-0,5229	123,23
0,662	-1,2991	186,11
2,012	-0,3511	109,33

Polynomial Models:

Here we use a simple polynomial model of second degree ($f(x) = ax^2 + bx + c$) to fit the data.

When we look at the plots, no big differences between the linear fits are obvious.

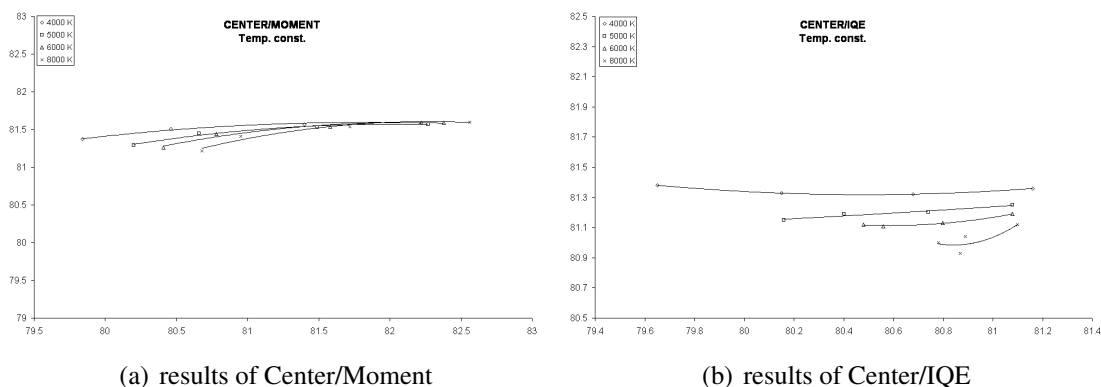


Figure 2.10: Polynomial models fitted for constant temperatures

Coefficient of determination

To evaluate the quality of the two methods the behavior of the so called coefficient of determination is considered. The coefficient of determination is the square of the Pearson(Karl Pearson, March 27, 1857 – April 27, 1936)'s correlation coefficient r .

2.2. TESTS OF THE ACCURACY OF THE MIDAS METHODS

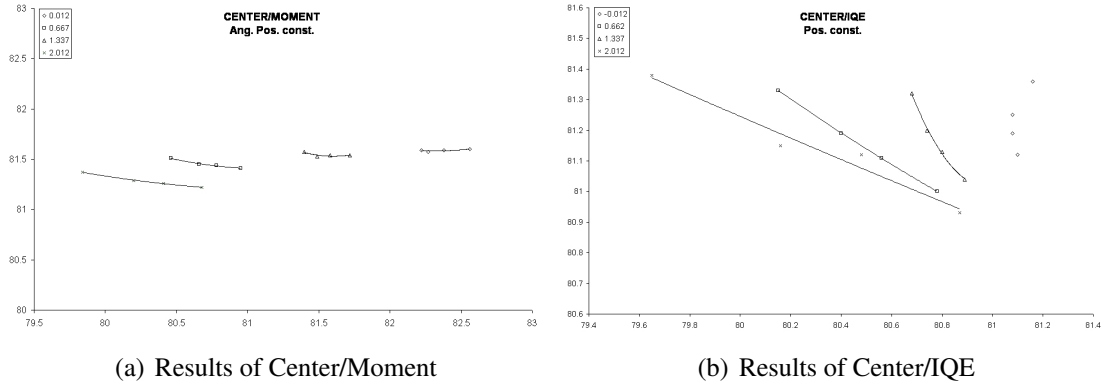


Figure 2.11: Polynomial models fitted for constant angular positions

The correlation coefficient r is defined the following way:

$$\frac{\sum (x_i - \bar{x}) * (y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x}^2) * \sum (y_i - \bar{y}^2)}}, \forall x_i, y_i \quad (2.3)$$

The coefficient of determination[48] is a measurement how good our data can be described by certain regression functions. In our case the regression functions are a polynomial fit of the first and the second degree. A coefficient of determination of 1 means that our data lies exactly on this regression function whereas a coefficient of determination of 0 would mean that our data can not be explained by that linear regression function.

Result

If we try to compare the results from the fitting tests trying to distinguish the effects of the two physical parameters temperature and distance to the optical axes for the two methods Center/Moment and Center/IQE we can get the following result:

E.g. for a constant temperature of 4000 K: if we compare the r^2 value of a fit with a linear model, we see that the Center/Moment algorithm delivers results which can be easier fitted by that model then the Center/IQE algorithm. On the other hand if we take a look at the behavior of a polynomial model we find that the Center/IQE results can be better explained by fitting to a curve then the Center/Moment results.

Evaluation To evaluate all results from the above fittings and to be able to analyze the quality of the two methods by the number of curves which can be fitted best, the number of cases when one of the two methods delivered the best results was counted (see table 2.14).

Comparing the r^2 values for the eight different data sets, we find that the Center/IQE method delivers in 10 of the 16 cases better results, the Center/Moment method delivers the best results in 6 cases.

CHAPTER 2. TESTS WITH ALL PSFS

Table 2.14: Comparison of the methods in terms of the coefficient of determination r^2

const.	regression function	Moment r^2	IQE r^2	better method
4000K	linear	0,8297	0,1137	Moment
	polynomial	0,9794	0,997	IQE
5000K	linear	0,8498	0,937	IQE
	polynomial	0,9711	0,938	Moment
6000K	linear	0,8392	0,8535	IQE
	polynomial	0,9621	0,9931	IQE
8000K	linear	0,83	0,5925	Moment
	polynomial	0,9539	0,6935	Moment
-0,012	linear	0,4407	0,5044	IQE
	polynomial	0,5215	0,9419	IQE
0,662	linear	0,2948	0,9702	IQE
	polynomial	0,7387	0,9971	IQE
1,337	linear	0,9538	0,999	IQE
	polynomial	0,9832	0,9999	IQE
2,012	linear	0,9859	0,9645	Moment
	polynomial	0,9986	0,964	Moment

A more detailed observation of the two fitting methods does not help us to decide which method is better: From the 8 linear models in 3 cases the Center/Moment algorithm delivers the best results. We get the same result for the polynomial models were again 3 times the Center/Moment shows the better results and is easier to fit, but in 5 cases it is the Center/IQE method which shows better results. So we have a final result of 10 to 6 for the Center/IQE method, which is no strong decision criterion.

Conclusion

It is not easy to distinguish the quality of two centroiding algorithms by only 16 simulated PSFs with two changing parameters. After some trials we can say that the Center/IQE method delivers better results from the mathematical point of view. But this does not mean that the Center/IQE method is the best to illustrate the physical relations causing deviation in the position of the PSF. If the optical system would have the same properties as a simple spherical lens the PSF should show by increasing distance to the optical axis an increasing distortion in its shape and therefore its centroid should show an increased shift value for an increased angular distance. If we take into account that the models are only made by four data points each we have a very wide error range in the judgment which method is delivering results which are then easier to be fitted by a physical model.

2.2. TESTS OF THE ACCURACY OF THE MIDAS METHODS

So after all it must be stated that its not possible to judge whether the Center/IQE or the Center/Moment algorithm delivers better results. The only algorithm that can be excluded from further astrometric considerations is the Center/Gauss algorithm due to its very wide variation of results.

Chapter 3

Tests with one shifted PSF

Our problem is now that we do not know the properties of the used system sufficiently to calculate a ray tracing. We only have a set of sixteen simulated PSF'S from the CoRoT Team in France. To judge the astrometric accuracy of CoRoT it is not sufficient to have a set of sixteen simulated PSF's for four different temperatures and four positions on the optical plane. CoRoT is observing the same field for a long time period, at least 25 days for the short runs. This means that the observed stars do not change significantly their absolute position on the CCD during the time period we are interested in. Also their temperature, therefore color stays during the observation time approximately the same (although variable stars can vary significantly in temperature and therefore in color). So its not necessary to work with all sixteen PSF's to judge the astrometric performance of CoRoT.

It is much more important to measure the accuracy of small shifts of one constant PSF. The idea behind the tests shown in this chapter is to shift the PSF via different techniques and to observe whether these shifts can be verified or not, and if yes, up to which accuracy the shift can be measured.

We have two way to shift our simulated PSF. One way is to rebin our four times over-sampled data so that the bins are shifted by one pixel in the simulation which corresponds to a shift by 0.25 pixels in the "real" data. The second way is to shift our PSF using a spline interpolation. This way allows us to shift the PSF in smaller steps then the 0.25 pixels, but it does not increase the amount of information stored in the ray-tracing simulation. A shift we make using a spline interpolation does not correspond to a physical shift of the PSF by a certain distance, it is only a numerical interpolation.

To be more flexible in the analysis of the methods than with the given MIDAS routines, all tests have been developed in IDL¹.

¹Interactive Data Language, ©by ITT Visual Information Solutions, Boulder, Colorado

3.1 Creation of shifted Originals

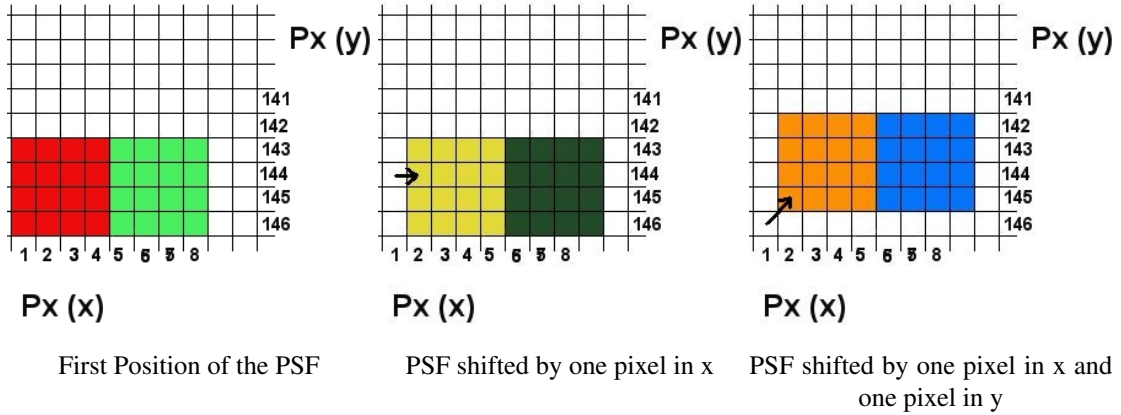
The first idea to estimate the accuracy of the CoRoT PSF for position measurement was proposed by Dr. D. Sinachopoulos and consisted of a simulation of a great number of PSF's with a distance of a milliarcsecond in the field of view. These simulations should have been computed using the same star only the position of the star should be varied.

The method would have consisted of the derivation of the positions of the different PSFs and then to observe whether the positional difference of a milliarcsecond between two simulations could be reproduced. To calculate these shifted simulations using ray tracing the optical properties of the whole system must be known. This was not the case for us, mainly because the optical design is the property of Alcatel France and was not freely available.

So we had to invent some workarounds. One workaround was to rebin the simulations from oversampled 160*160 pixels to the "real" 40*40.

The second idea is to shift the PSF using a spline interpolation and then trying to verify these shifts with different algorithms.

3.1.1 Rebinning of the simulations



The simulations have a resolution of 160 x 160 data points, this means they are four times oversampled.

The "real" area size of the simulated frame is 40 x 40 pixels on the CCD.

The intensity measured on each "real" pixel on the CCD corresponds to the mean value of 4 x 4 (sixteen) simulated data points. That means the average of the pixel values of 4x4 neighboring pixels is taken for the pixel value of one pixel in the realistic image.

To create an realistic image of a star on the CCD, the 160 x 160 pixels of the simulation ($PSF_{160 \times 160}$) have to be rebinned.

$$PSF_{40 \times 40}(i,j) = \left(\sum_{i,j}^{i+3,j+3} (PSF_{160 \times 160}(i,j)) \right) / 16 \tag{3.1}$$

$$i, j = 0, \dots, 39$$

Like this we received a rebinned image of 40*40 pixels. The idea is now to use this rebinning as a method to create shifted images without any change in the data, especially without any interpolation or extrapolation which could influence the accuracy of the centroiding algorithm.

To create a shifted image in x and y direction by 0.25, 0.5, 0.75 and 1 pixels in the "real" image, the value of shiftx and respectively shifty is set to 0, 1, 2, 3 or 4.

$$I_{40*40}(i+shiftx/4, j+shifty/4) = \left(\sum_{i,j}^{i+3, j+3} (I_{160*160}(i+shiftx, j+shifty)) \right) / 16 \quad (3.2)$$

$$shiftx = 0, 1, 2, 3, 4, 5$$

$$shifty = 0, 1, 2, 3, 4, 5$$

To calculate the outer pixels of I_{40*40} the co-domain was increased to 163*163 pixels and filled with zeros. So we received images with a shift of 0.25 real pixels in the x- and y-direction. The source code for the rebinning can be found in Appendix C.1.

3.1.2 Creation of shifted images using spline interpolation

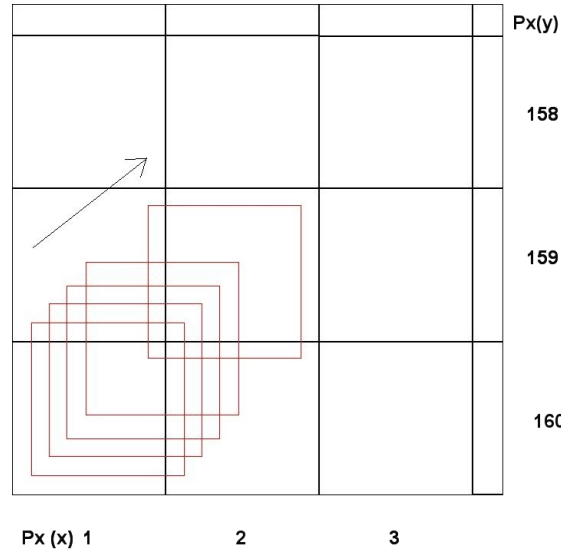


Figure 3.1: Created of shifted images using spline interpolation

We computed a spline (using the standard IDL spline algorithm, details see [49]) with a resolution of 10 supporting points per data point, that means that the 160x160 data points were enlarged to be a grid of 1600x1600 data points to create shifted images. From that grid a shifted PSF was created by taking each 10th data point from a row, respectively

3.2. SOME CENTROIDING ALGORITHMS AND THEIR QUALITY

column. Like this we receive a grid of 10 PSF's with a distance of 1/10 pixel in the oversampled PSF, that means 1/40 pixel distance between two PSF's in the "real" CoRoT field.

This shift was done in two steps, first an calculation of a spline interpolation curve in x direction, then the data points of a shift by $n \cdot 0.1$ px were stored. Then the same was done for the y direction, like this the PSF "moved" in the direction of the arrow in 3.1 with a step value of (0.1 px / 0.1 px). The source code for the spline shift can be found in Appendix C.2.

3.2 Some centroiding algorithms and their quality

To test the accuracy of certain algorithms we tested some of them with different methods. The algorithms we tested were :

- center of gravity algorithm
- symmetry
- triangle fit

The test for the center of gravity algorithm has been done completely, some tests have only been worked out to some certain level, where it was obvious that the needed accuracy could not be achieved by the tested algorithm, namely the triangle fit only exist in some plots and should be worked out further.

3.2.1 The center of gravity algorithm

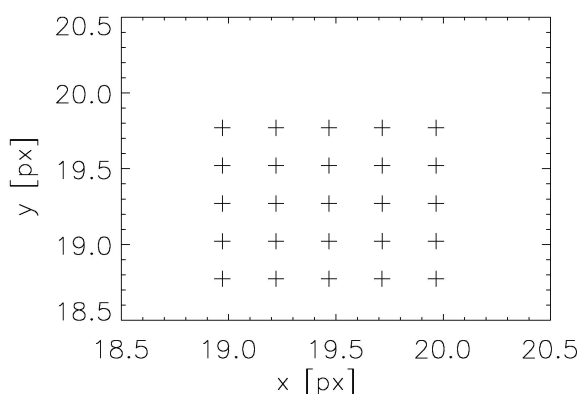


Figure 3.2: Centroid results for -0.012 and 6000 K

We calculated a centroid position for the positions created by the rebinning method. This was done using a center of gravity algorithm similar to the one that was used by the MIDAS Center/Moment algorithm. The first test was done with a PSF of a star with a Temperature of 6000 degree K and at the CCD position (0.420, -0.012). The results of these calculations can be seen in figure 3.2. The first position, the result for the original psf at pixels 20.0, 20.0 can be seen in the upper right corner.

The difference between the positions should be 0.25 pixels in x and in y direction. First we see that a shift of the simulated PSF results for all PSFs in a similar shift of the computed centroid position. We also see the influence of the asymmetry of the PSF on

the centroid position. A perfectly symmetrical PSF and a perfect algorithm would deliver centroid positions of (20.00,20.00), (19.75,20.0) and so on until (18.75,18.75). We see already that the results of the center of gravity algorithm are shifted to the direction with the pixels of higher intensity.

When we observe the results in table 3.1 and 3.2, we also see that the positions are not really lying in a distance of 0.25 pixels but have slightly different distances.

The results of the centroid algorithm for 25 shifted positions:

shiftx/shifty		0.0	0.25	0.5	0.75	1.0
	i/j	0	1	2	3	4
0.0	0	19,4751	19,2248	18,9755	18,7272	18,4794
0.25	1	19,4751	19,2248	18,9754	18,7272	18,4794
0.5	2	19,4751	19,2248	18,9754	18,7272	18,4794
0.75	3	19,4751	19,2248	18,9754	18,72720	18,4794
1.0	4	19,4751	19,2248	18,9754	18,7272	18,4794

Table 3.1: x positions derived with Centroid

shiftx/shifty		0.0	0.25	0.5	0.75	1.0
	i/j	0	1	2	3	4
0.0	0	19,6890	19,6890	19,6890	19,6890	19,6890
0.25	1	19,4388	19,4388	19,4388	19,4388	19,4388
0.5	2	19,1903	19,1903	19,1904	19,1904	19,1904
0.75	3	18,9422	18,9422	18,9422	18,9422	18,9423
1.0	4	18,6935	18,6935	18,6935	18,6935	18,6935

Table 3.2: y positions derived with Centroid

3.2. SOME CENTROIDING ALGORITHMS AND THEIR QUALITY

3.2.2 Data analysis of the centroid results for rebinned images

To observe the behavior of the results in detail we have plotted the difference of the position x_{ci} to x_{ci+1} (see figure 3.3).

The rebinned PSF's have been shifted by 0.25 pixels and also this value should be derived as a difference between two neighboring centroid positions.

Differences in the resulting x values:

i/j	0	1	2	3
0	0,2503	0,2493	0,2483	0,2478
1	0,2503	0,2494	0,2482	0,2478
2	0,2503	0,2494	0,2482	0,2478
3	0,2503	0,2494	0,2482	0,2478
4	0,2503	0,2494	0,2482	0,2478

Table 3.3: $x(i,j)-x(i+1,j)$

i/j	0	1	2	3
0	0,0000	0,0000	0,0000	0,0000
1	0,0000	0,0000	0,0000	0,0000
2	0,0000	-0,0001	0,0000	0,0000
3	0,0000	0,0000	0,0000	-0,0001
4	0,0000	0,0000	0,0000	0,0000

Table 3.4: $x(i,j)-x(i,j+1)$

Differences in the resulting y values:

i/j	0	1	2	3	4
0	0,0000	0,0000	0,0001	0,0000	0,0000
1	0,0000	0,0000	0,0000	0,0000	0,0000
2	0,0000	0,0000	0,0000	0,0000	0,0000
3	0,0000	0,0000	0,0000	0,0000	0,0000

Table 3.5: $y(i,j)-y(i+1,j)$

i/j	0	1	2	3	4
0	0,2502	0,2502	0,2502	0,2502	0,2502
1	0,2485	0,2485	0,2484	0,2484	0,2484
2	0,2481	0,2481	0,2482	0,2482	0,2481
3	0,2487	0,2487	0,2487	0,2487	0,2488

Table 3.6: $y(i,j)-y(i,j+1)$

We find a difference in x between 0.2478 and 0.2503 (see figure 3.3, the plot in the first row on the left side and table 3.3). In y we find a difference between 0.2484 and 0.2502 (see figure 3.3, the plot in the second row on the right and table 3.5).

These results allow us to judge the accuracy of the whole method, because the difference between the wanted 0.25 and the measured values can be seen as an error in the measurement. The problem is that the accuracy we can measure by that way only shows us how good the position a PSF of the given number of data points can be measured. We cannot judge how good a PSF with a higher number of data points would be measurable. If we had a higher number of pixels per PSF the error should be lower.

Also we see that the computed y-position for shifts in x-direction varies by only 0.0001 pixels. Because no change in the shape of the PSF in this direction was made there should also be no shift found. The value of 0.0001 is only caused by rounding errors. The same effect can be found for x-values for shifts in y-direction.

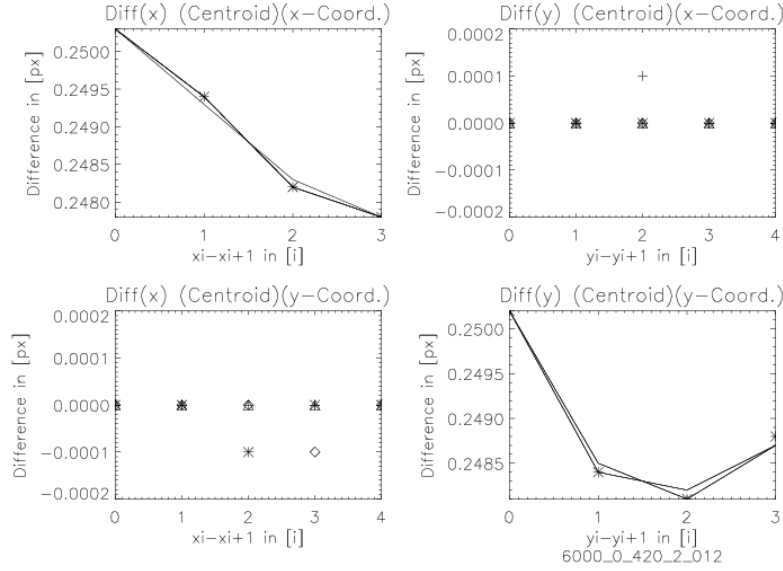


Figure 3.3: Differences between shifts for 6000 K, 2.012

Error estimations When we compare the results from the centroid algorithm and compute the distances between the derived positions, we find that the error lies between 0.003 and -0.022 pixels, that means that the centroid algorithm in this form can reach an accuracy of for a conservative accuracy value of 0.025 pixels a maximum theoretical accuracy of 250 mas (compare section 1.3.9). We must note that for this test, no noise or jitter simulation has been included, this value is only a numerical value due to the low number of data points.

To estimate whether the result for the PSF for 6000 K and the angular distance of -0.012 can be reached with all the PSF's the test has been made with all 16 PSF's. These plots can be found in the appendix and show similar error ranges. So we must say that without a better and more detailed knowledge of the PSF we cannot reach a better accuracy than 250 mas per observed star.

Conclusion In this test, we find that a shift of 0.25 pixels can be measured with an accuracy of 0.025 pixels corresponding to 250 mas in the CoRoT field of view. No jitter or noise simulations have been included nor did we work out the possibilities of obtaining a time series.

The source code for the centroid algorithm can be found in Appendix C.3.

3.2.3 Symmetricity

Based on a method to find position of a spectral line by finding its symmetry axis [41], we tested how accurate positions of our PSF can be derived with this method.

The Symmetricity Function The main idea behind this method is to describe mathematically the position of a symmetry axis of a curve. The Symmetricity Function is a Function describing the Symmetricity of each data point. We compute for each point this symmetricity value. For a really symmetrical curve we would get one data point with the value of 0. But if we do not have a truly symmetrical curve, we can say something about

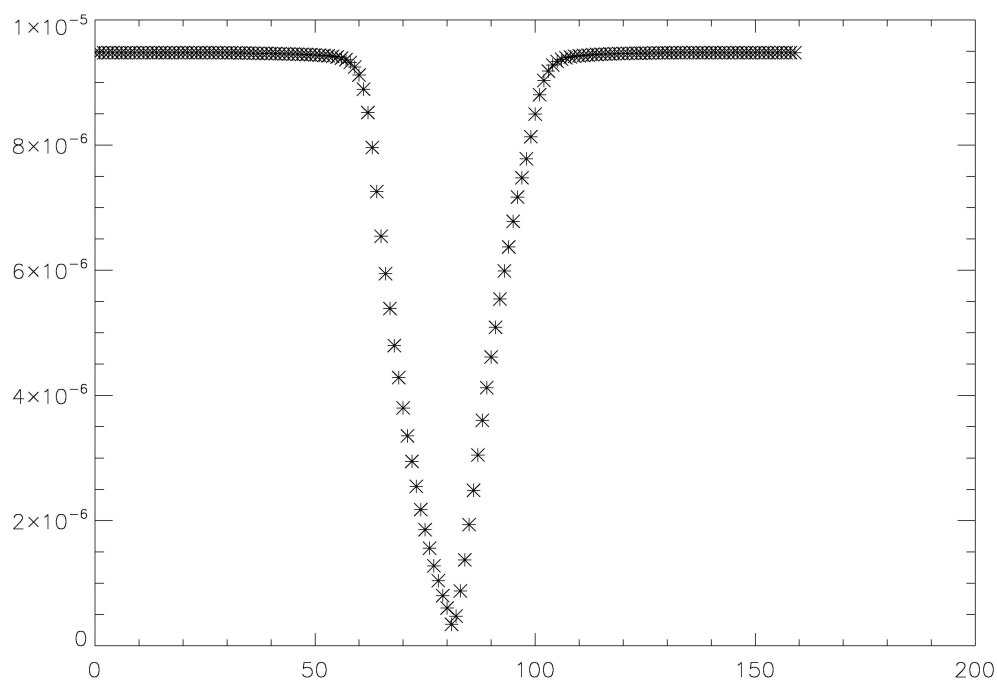


Figure 3.4: Symmetricity values for the original PSF

the position of the curve by comparing the curve, its minimum, or other properties with the minimum of a second curve, a shifted PSF.

The Symmetricity $F(x)$ at the position of pixel x is calculated in the following way: Be $I_{l1}(x)$ the intensity of pixel x in row $l1$. The dimension of the observed image be $k \times l$ (in our case $k=l$). We define the symmetricity F of the pixel x of the row $l1$ as

$$F_{l1}(x) = \sum_i (I_{l1}(x-i) - I_{l1}(x+i))^2, \text{ with } j < i < 2j. \quad (3.3)$$

The value of j depends on the value of x , j is the maximum distance from x to the first or last pixel of the row. That means for a PSF of 160×160 px, that the x value varies between 0 and 159, and therefore j varies for each data point. For $x = 0$ we have $j = 159$, also for $x = 159$, but for a value close to the middle of the PSF, e.g. $x = 75$, we have a j value of 85. To be able to work with more data points than the size of the PSF, data points outside the dimensions of the PSF are being set to 0. For $x = 85$ we would need the 85 data points on the left side of x and the 75 data points on the right side of x , plus 10 data points which we set to 0.

$$I(x-i) = 0 \quad \forall i \text{ with } (x-i) < 0, \text{ and} \quad (3.4)$$

$$I(x+i) = 0 \quad \forall i \text{ with } (x+i) > 2j \quad (3.5)$$

If the intensity function $I(x)$ is symmetrical, the symmetry axis is found at $F(x) = 0$. The point of the highest symmetricity is found at the minimum of $F(x)$. In our case, the PSF is not exactly symmetrical, so $F(x)$ is not equal to 0 for all x values. But we can define the position of the PSF by using the minimum of the symmetricity to describe to position of the PSF.

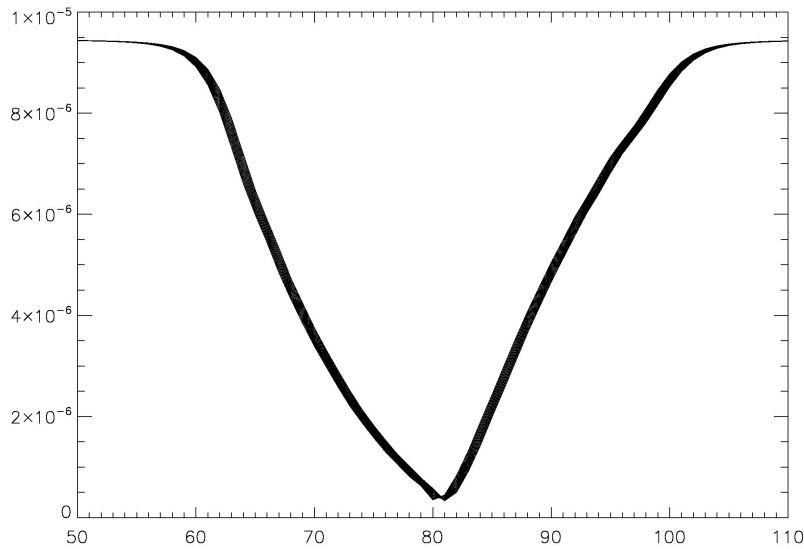


Figure 3.5: Symmetricity function for spline shifted PSF, large plot

One problem we see here is that the PSF is a two-dimensional function and we find a minimum for each row of the image, these minima's differ from row to row. The idea

3.2. SOME CENTROIDDING ALGORITHMS AND THEIR QUALITY

was to compute for some central rows of the PSF the symmetricity function and to find the minimum of each row. The position of the PSF is then the mean value of those central rows.

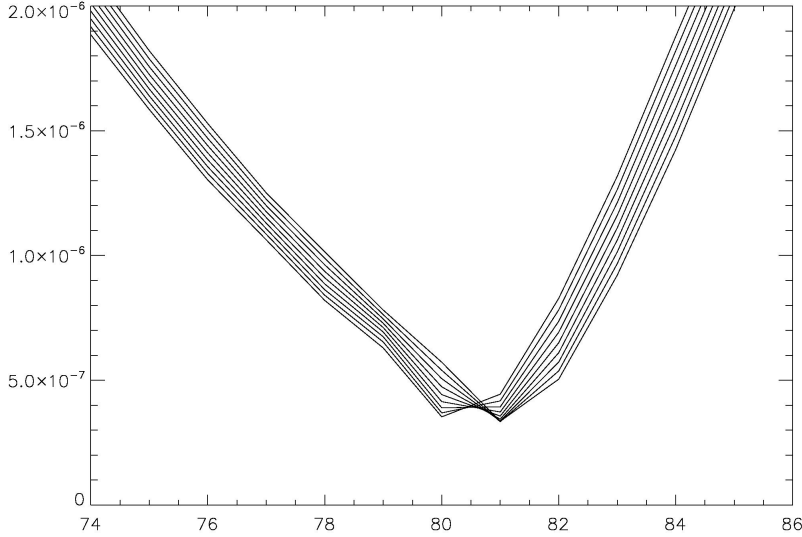


Figure 3.6: Symmetricity function for spline shifted PSF, detailed view

In figure 3.6 we see a more detailed view of the symmetricity curve. As we see it is possible to verify a shift from curve to curve.

Computation For the test we used rebinned images with a size of 40x40 px. The function $F(x)$ has been computed for all rows of of the PSF, during that process it was found, that due to the properties of the PSF only the rows 15 to 24 are reasonable to evaluate. The simulated PSF has been "spline"-shifted in steps of 1 pixel to the right and has then been rebinned, that means the tested PSFs were shifted to the right in steps of 1/4 pixel (0.0,0.25,0.5,0.75,1.0). As described above only the symmetricity functions for the rows 15 to 24 have been computed, also the number of data points per PSF is very low. To verify the shift of one curve to the next, first the curves had to interpolate because the values for $F(x)$ were all discretely defined for the integer values of x . The values in the flat area of the PSF were not usable for this, so the estimations were restricted on the slope. The shift $\Delta(x)$ between two curves F_1 and F_2 has been computed as:

$$\Delta(x) = (F_1(x) - F_2(x))/(F_2(x + 1) - F_2(x)) \quad (3.6)$$

Where $F_1(x) > F_2(x)$ and $F_1(x) < F_2(x + 1)$. Only the pixels 16,17,18,21,22 and 23 are in compliance with that condition. The question is now: can this shift be verified by the properties of $F(x)$.

Results: The values for all ten rows have been averaged and their variance has been computed. The mean value varies between 0.2653 and 0.284 and so lies systematically

by 0.03 pixels above the expected value of 0.25 pixels. With this method an accuracy of 0.06 pixels can be reached, though the values for a shift in y-direction are determined systematically too large. Analysis with shifts in opposite and in horizontal directions should be made to investigate the behavior of this method. Also no noise simulations have been included. The IDL code for the symmetricity algorithm can found in Appendix C.4

3.2.4 Triangle fit

The idea of this fitting algorithm was to fit our PSF in a simple geometrical form, beginning with a triangle in two dimensions and then going further to a pyramid in three dimensions.

It seems easy to fit a simple geometrical form like a pyramid, or in one dimension a triangle to the volcano shaped PSF, but its not as easy as it may seem, because the triangle has one prominent point, the top. It is the easiest way to fit a triangle by fitting two lines, but the problem remains, where to put the top of the triangle. The first idea of a constraint would be: if one data point lies below a neighboring point, then we know that we are on the decreasing side of the triangle.

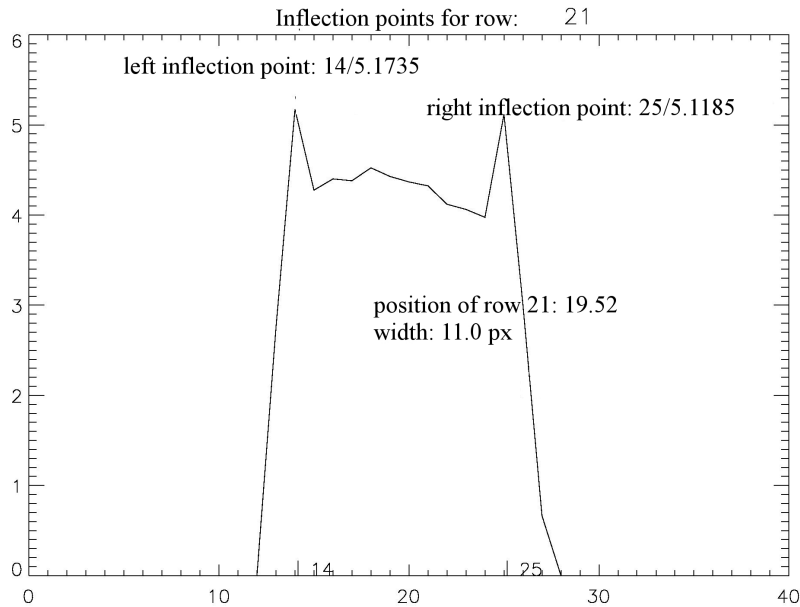


Figure 3.7: Triangle fit, first results

BUT, this leads our PSF analysis very fast to problems, because we have many small spikes, especially in the flat part of the "volcano". So, we have to modify our constraints and declare that if more than x data points are lower than a certain point (the maximum), then we have reached the decreasing side of the PSF. From that point where we can clearly say that we have reached the decreasing side we can begin to fit a line to the data points.

Mainly we find that we have the same problem as in the symmetry algorithm, we can only use a small number of the pixels, and this fact decreases the accuracy of the measurement of the position to a value that is far beyond the values we would need.

3.2.5 Jitter Simulations

With the help of Kristopher Kareff and Torben Arentoft from Aarhus, DK (private communication, 2005) we created some simulations of jitter and also some noise simulations.

In figure 3.8 we see a 1000-image simulation, the curve is the result of the centroid algorithm for all 1000 shifted images. The only noise simulated is jitter, no photon noise, sky background or readout noise is included.

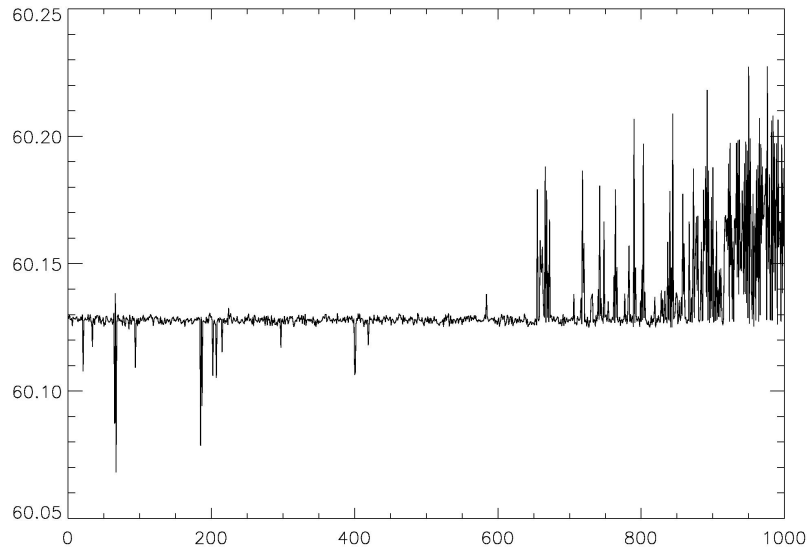


Figure 3.8: Centroid results for "jittered" PSF's

The main idea was to create a large number of shifted positions. The shifts have been done by spline interpolation.

The derived positions should show a certain inaccuracy caused by the quality of the centroiding algorithm used. What we found was that the centroiding algorithm delivers exactly the same positions that were put into the simulation, only some rounding errors were found. It would have been possible to use these simulations to simulate a set of simulated images and then trying to derive positions, but as the simulations only cover one PSF and not an entire field and because it was more important to really test different mathematical centroiding algorithms further experiments were performed with the jitter simulations.

3.2. SOME CENTROIDDING ALGORITHMS AND THEIR QUALITY

Chapter 4

Astrometry with "real" CoRoT data

4.1 The Observations

Finally, it is always best not to work with simulations but to take a look at some "real" CoRoT data. To judge the astrometric accuracy of CoRoT on a "real" full frame dataset Michel Auvergne contributed a dataset of 5 full frame observations:

from **May 9, 2007**, first long center run :

1 ANO_FULLIMAGE_20070509T195122_A1.fits

2 ANO_FULLIMAGE_20070509T202602_A1.fits

from **October 16, 2007**, first long anticenter run :

3 ANO_FULLIMAGE_20071016T184715_A1.fits

4 ANO_FULLIMAGE_20071016T173859_A1.fits

from **April 9, 2008**, second long center run

5 ANO_FULLIMAGE_20080409T190119_A1.fits

from the last run only received one file was received, probably only one full frame was downloaded.

4.1. THE OBSERVATIONS

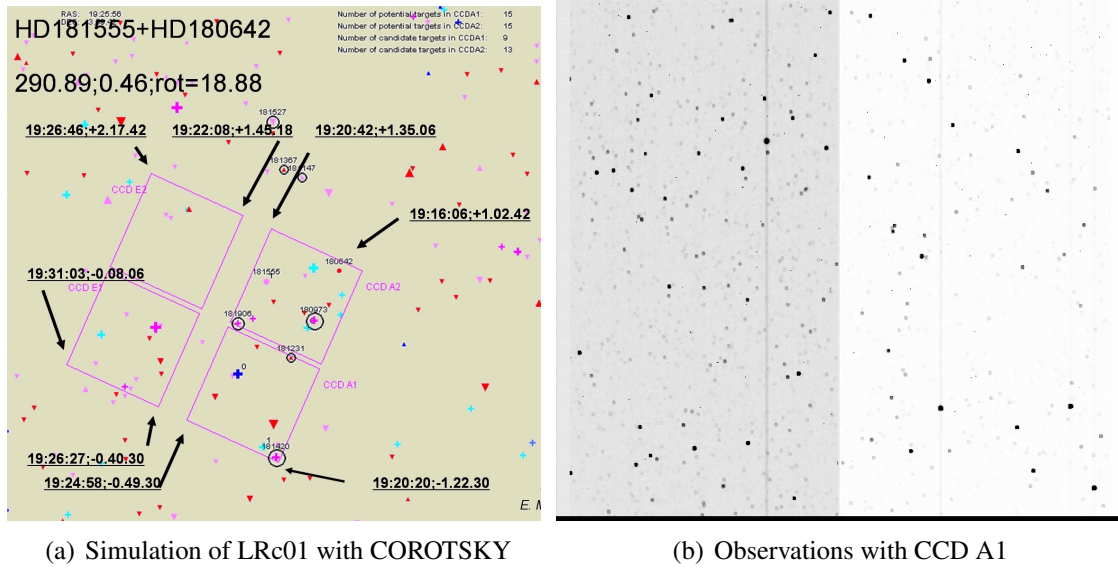


Figure 4.1: LRc01

First long center run (LRc01)

The first field LRc01 from May 9, 2007 is pointing towards the galactic center (see table 1.5) and is being observed for 152 days. Two observations of field LRc01 were observed, observation 1 is taken at 19:51:22 and observation 2 at 20:26:02, which gives a time difference between the two images of 35 min 20 sec.

In figure 4.1, on the left image (fig. 4.1(a)) we can see the coordinates of the corners of the CCD A1 and also the rotational angle of the field from a COROTSKY simulation [27]. On the right side (fig. 4.1(b)) we see an image of the CCD A1. We see directly the different noise levels in the two halves of the CCD, this is because for each half of the CCD there is a different A/D converter with different readout noise reading out the signal.

First long anticenter run (LRa01)

The second observed field was observed on October 16, 2007. It is pointing towards the anticenter of the Milky Way and is observed for 150 days. Consider that CCD's are turned around compared to the LRc fields due to the turn-over of the satellite when changing the field of view from the anticenter to the center and back. To put the CCD's in the original position the satellite would need to be turned around along two axes. It is probably easier just to let the CCD's change their relative position and to turn the satellite only around one axis.

From field LRa01 two full frame observations were received from 17:38:59 and 18:47:15 which leads to a time distance of 1 h 18 min 16 sec. In figure 4.3 we see the field around HD 49385 and HD 49330. On the left side we see the observation in the Palomar Digital Sky Survey (DSS) and on the right side we see the actual observation performed by

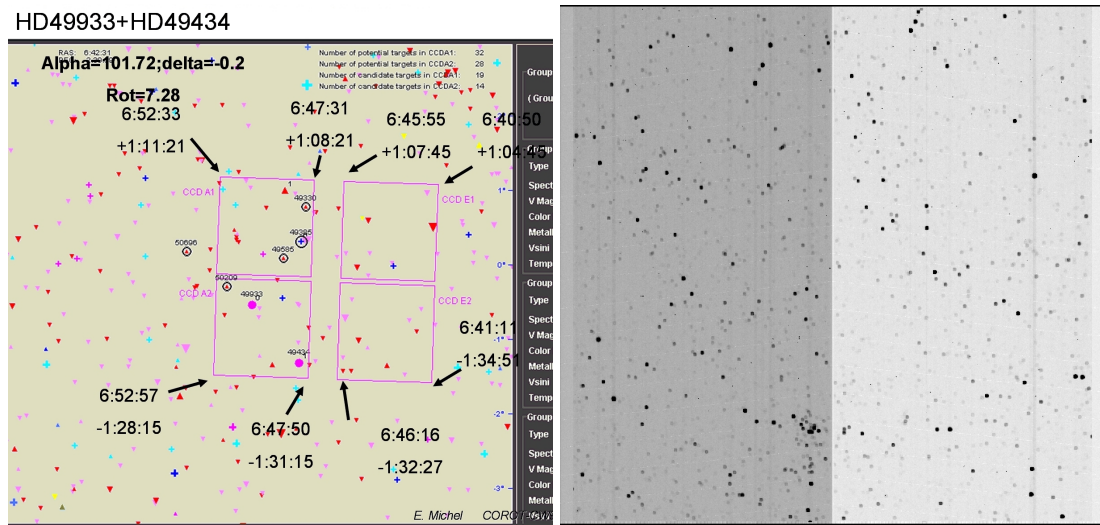


Figure 4.2: Coordinates of the corners of A1 in LRA01 and observation of the field

CoRoT. We see how defocused the optical system of CoRoT is. Its also nicely visible that the defocused optics leads to a problem often forgotten, the problem of overlapping PSFs and background stars.

Second long center run (LRc02)

No more full frame observations were downloaded. The only planned purpose of full frame pictures was the selection of the windows around selected stars which were constantly downloaded during the observation run. For the determination of parallaxes it would be helpful to have full frame observations also from the end of each run. But whether parallaxes are measurable at all was still a question when the downloading schedule was determined. Anyway we suggested to download the full frame observations from the end of each run too.

4.1.1 First data analysis, star count trials, identification of stars

Our goal is to measure the positions of the stars in the field automatically, we start first with a very simple algorithm.

Definition of a star If the ADU value of a pixel is below the cut-off value and a neighboring pixel (diagonally) has an ADU value higher than the cut-off value, the star is counted. The source code to count the stars in the field can be found in Appendix C.0.8.

Result A cut-off at 6000 ADU is a good starting assumption to test the star finding algorithms, later working with 2000 and even 1200 should be good. 1200 ADU is also

4.1. THE OBSERVATIONS

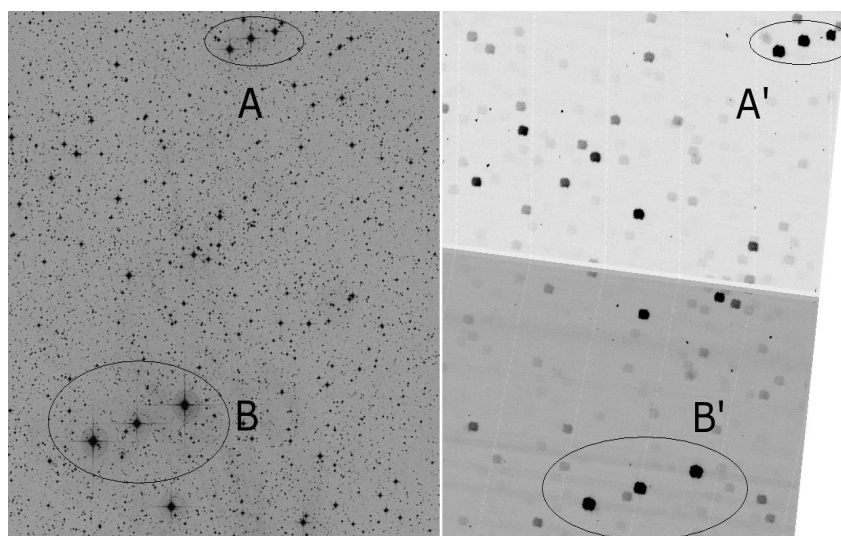


Figure 4.3: left: Palomar DSS data, right: Part of Field CCD A1 in LRa01

Table 4.1: Number of detections using the simple algorithm C.7 in file 2, first long center run

cut-off at ADU	8000	6000	2000	1200	500
detections	94	100	590	5684	none

close to the noise level. Obviously the results from the procedure C.7 do not deliver the correct number of stars in the field but only a raw value corresponding to the number of detected stars. It has to be improved, not just to find bright pixels but to find stars.

Improvement of the algorithm Now the idea is that for a pixel that "counts", and therefore lies above the noise level and is in contact with another bright pixel with a lower counting number than the pixel itself, the number of stars is not increased. The search started with left edge, then right edge. The source code to find the edge can be found in Appendix C.0.9.

This algorithm does still react to hot pixels, and each star still delivers around 5 results, but these "detections" can be easily reduced. After doing this it was found that from the list of stars only 4 stars remain from 6000 ADU. This corresponds to the observation with the naked eye.

Table 4.2: Coordinates of the "stars" for a cut-off at 6000 ADU in file 2

star #	1	2	3	4
x	791.00	1453.00	1948.00	2038.00
y	1515.00	490.00	497.00	185.00

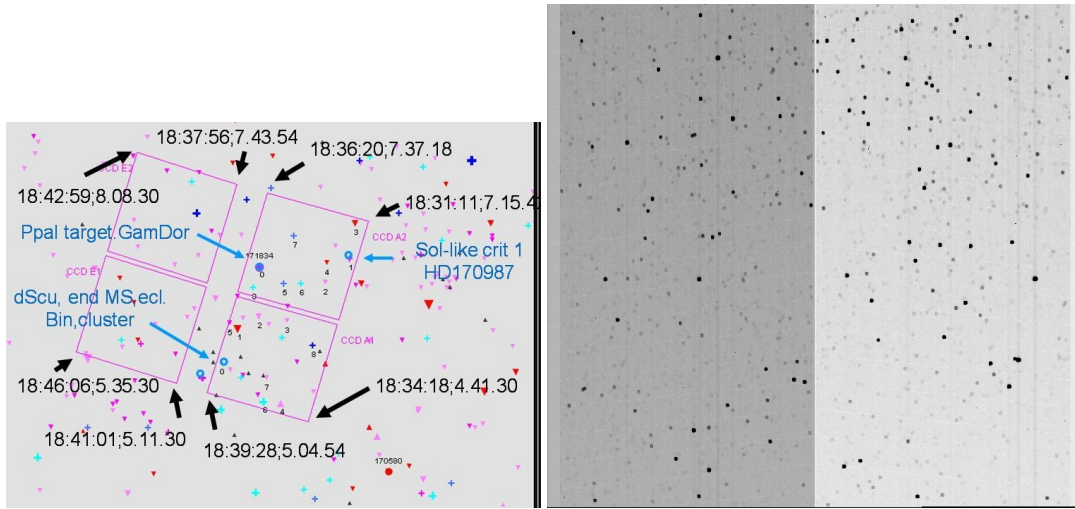


Figure 4.4: The positions of the CCD's of LRC02 and the observation



(a) Cut-off at 6000 ADU, 4 stars are visible

(b) Cut-off at 2000 ADU, 20 stars are visible

Figure 4.5: File 2 after a background reduction with different cut-off values, we see that in the field with a cut-off at 6000 ADU there are only 4 stars visible, in the field with a cut-off at 2000 ADU we see 20 stars

With the same method as described above and a cut-off of 2000 ADU we find 20 "stars".

Center of gravity results: Now we can use the star positions we found to measure the position of these stars with a more precise algorithm. As we saw already from the simulations, the center of gravity algorithm delivers the best results, and is also now our first choice. We tried the methods as described above on the two observations of the first long center run field with a time shift between the two observations of half an hour (exactly 35 minutes and 20 seconds) **for 5 stars**. We found that the measured shift between the two observations differs by 0.123 pixels in x and 0.030 pixels in y . This corresponds to an accuracy of the magnitude of **one arcsecond**.

4.2 Centroid positions for bright stars in field LRA01 for the two observations of 17:38 and 18:47 from October 16, 2007

As a test for the accuracy of a derived positional information from a COROT field, i observed the data from field LRA01, consisting of two full-frame observations with a time distance between the observations of 1 h 18 min 16 sec (4.1)

- ANO_FULLIMAGE_20071016T173859_A1.fits
- ANO_FULLIMAGE_20071016T184715_A1.fits

4.2.1 Selection of stars

The technique developed in section 4.1.1 was used as a start and it was tried to let the algorithms work automatically.

Starlist In these two fields the brightest stars were selected using IDL. The main idea is to ignore any information that is lower then a certain cut value, it was decided to set this cut at 2000 ADU, so an easy to handle number of stars was received. A bright pixel is detected in the moment a pixel $x + 1, y + 1$ is brighter then its neighboring pixel x, y AND if it is brighter then the cut value. In a first step the stars like this are counted to know the amount of memory needed for all stars, then the same routine stores all positions and the number of the detection in a list. This rudimentary first test counts all pixels brighter then the cut value, in this case 2000 ADU and returns the whole list of detections, this test is very rude, but it gives us a starting point. The source code for a first list of stars can be found in Appendix C.9.

4.2.2 Starlist filter

To filter this list another program is needed to distinguish the detected bright pixels from hot pixels and to assign multiple detections of steps to the same star. Also bright regions too small to be a star (hot pixels) are filtered by this second program. The program consists of three steps. We know the number of stars from *starlist.pro*, in this case thats 825, so the needed memory can already be assigned .

1. If the position in x of a detection is the same for two lines, these lines are stored as one and the broadness of the "star" is increased by 1.
2. The same routine is done for the position in y
3. If a star has a broadness of 6 list entries or more its position is kept in the list, if not deleted from the list.

Like this we get a list of stars with the coordinate of the first detected pixel (mostly the uppermost left pixel of the star's PSF), its intensity and its broadness. We see that some stars are not counted and some stars are counted multiple times . This test is also very open to errors and misinterpretations, but it eliminates lonely "hot" pixels, and we now have a list of "stars".

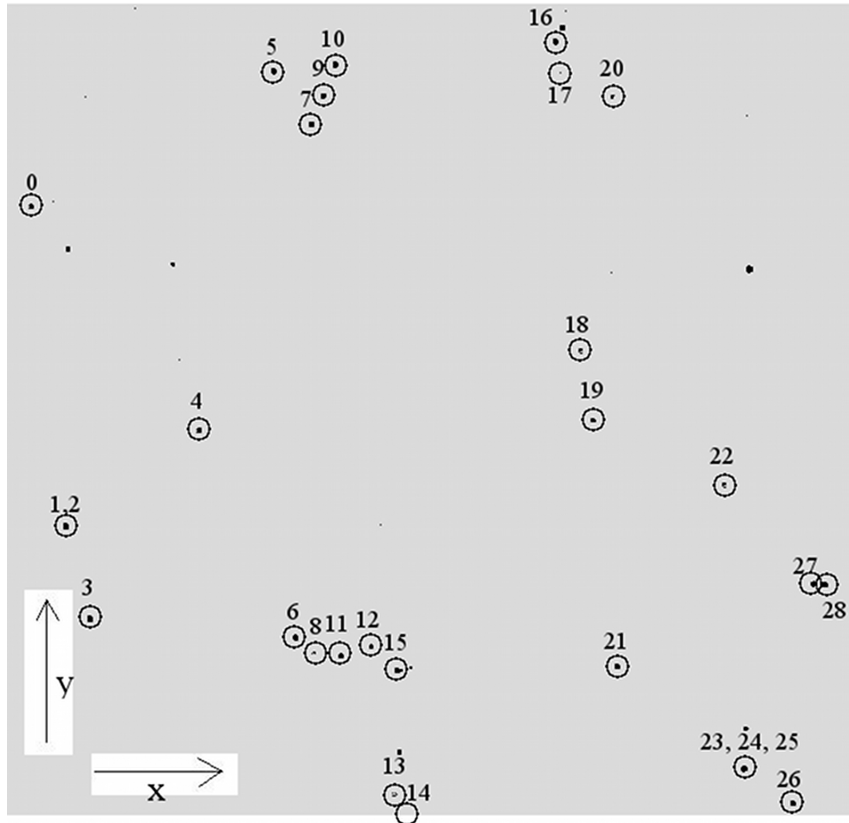


Figure 4.6: Stars in LRA01, 17:38 identified with automatic filter algorithm

With this filtered list, already the brightest stars in the field LRA01 can be identified, see figure 4.6.

The detailed IDL routine can be found in Appendix C.0.11

4.2.3 Determination of star position and shifts between the two fields

For this list of stars i calculated the position of the intensity centroid using the program "centroid_list", the program reads the filtered list of stars, and derives for each star within a certain limit an intensity centroid. The limits are 17 pixels in x and 18 pixels in y (the difference in the two directions i due to the shape of the PSF). Within these limits the centroid formula from equation 2.1 is applied. The result is stored in a .dat file.

The detailed IDL routine can be found in appendix C.0.12.

4.2. CENTROID POSITIONS FOR BRIGHT STARS IN FIELD LRA01 FOR THE TWO OBSERVATIONS OF 17:38 AND 18:47 FROM OCTOBER 16, 2007

The same process was done for the second field again. If we compare the resulting positions we see that each star shows position changes from field one to field two in a certain direction. This shift is very interesting, because it should be constant for all stars in the field. The fact that this shift is different for stars with a similar PSF gives us a good instrument to estimate the accuracy that can be reached within a CoRoT field and we can even evaluate the astrometric sensitivity of the instrument as a whole like this.

Table 4.3: Centroid results for full frame of field LRa01

	start value		centroid values for field				shift	
			17:38		18:47			
i	x	y	x	y	x	y	Δx	Δy
0	58	1533	63,711936	1539,9457	66,313126	1538,7825	2,60119	-1,1632
1	144	724	151,34943	730,60239	153,7807	729,65188	2,43127	-0,95051
3	204	490	211,95082	496,13833	214,37566	495,38513	2,42484	-0,7532
4	481	967	488,01711	973,23608	490,53203	972,27836	2,51492	-0,95772
5	671	1875	677,09082	1882,4504	679,67419	1881,221	2,58337	-1,2294
6	726	442	732,8192	448,42287	735,29678	447,40685	2,47758	-1,01602
7	766	1741	772,99337	1747,7121	775,56934	1746,7201	2,57597	-0,992
8	773	404	778,11472	408,08555	780,81173	406,84508	2,69701	-1,24047
9	796	1813	802,16026	1820,0364	804,6203	1818,9637	2,46004	-1,0727
10	826	1893	833,13511	1899,8571	835,65864	1898,8292	2,52353	-1,0279
11	840	396	846,45899	402,01279	848,84014	401,09411	2,38115	-0,91868
12	918	421	924,45014	426,3083	927,11284	425,3531	2,6627	-0,9552
13	982	47	985,74405	53,178699	987,82013	52,794341	2,07608	-0,384358
15	1019	367	1023,8619	371,21383	1026,3463	369,92194	2,4844	-1,29189
16	1384	1950	1390,4499	1957,1064	1393,2214	1956,1364	2,7715	-0,97
18	1448	1167	1453,7575	1173,8835	1456,0795	1172,6921	2,322	-1,1914
19	1479	994	1485,1943	999,14024	1487,6397	998,40021	2,4454	-0,74003
20	1529	1813	1533,3028	1818,2143	1536,0529	1817,0027	2,7501	-1,2116
21	1538	369	1545,6652	374,86	1548,1865	374,11088	2,5213	-0,74912
22	1810	828	1815,8487	834,82732	1818,1147	833,52783	2,266	-1,29949
23	1860	110	1867,8673	116,20904	1870,1716	115,19514	2,3043	-1,0139
26	1984	23	1991,5419	28,474045	1994,0999	27,672273	2,558	-0,801772
27	2036	579	2043,5297	584,60832	2046,1168	583,73012	2,5871	-0,8782
28	2064	576	2071,5071	581,86072	2074,1632	581,0043	2,6561	-0,85642

The mean values of the shift between the two fields are

$$\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} = \begin{pmatrix} 2,50 \\ -0,98 \end{pmatrix}$$

with a standard deviation of :

$$\begin{pmatrix} \sigma(\bar{x}) \\ \sigma(\bar{y}) \end{pmatrix} = \begin{pmatrix} 0,16 \\ 0,21 \end{pmatrix}$$

The shift between the two fields should be the same for the whole field, and the standard deviation is a very sensitive parameter to evaluate the quality of the astrometric measurement.

This means that we are able to reach an accuracy of around 0.2 pixels with a centroid algorithm and the given data for the determination of a position of a star in the field. Expressed in arcseconds we can reach an accuracy of half an arcsecond (see section 1.3.7). That means the two fields are shifted by 6.25 ± 0.5 arcseconds in x and -2.5 ± 0.5 arcseconds in y direction while no evidence for twisting of the field of view was found.

In this data are already some outliers were eliminated, also some data points ("stars") (marked with naN, which stands for "not a Number") were lying too close to the border of the CCD, so their position could not be measured with the same algorithm on the second field because they had shifted out of the detection area. The detected stars number 1 and 2 are the same star. Stars number 23, 24 and 25 are also the same star, the star number 14 is too close to the border of the image to be detected in the second field and for the star number 17 only parts of the PSF have a higher intensity then 2000 ADU.

Table 4.4: Outliers from centroids for full frame of LRA01

	start value		centroid values for field				shift	
			17:38		18:47			
i	x	y	x	y	x	y	Δx	Δy
2	151	722	153,97262	730,27158	155,28576	728,81955	1,31314	-1,45203
14	988	8	991,79187	9,4135823	naN	naN	-	-
17	1402	1877	1402,3603	1878,6679	naN	naN	-	-
24	1865	108	1869,552	116,12097	1870,8582	114,46537	1,3062	-1,6556
25	1873	110	1873,93	116,02761	1875,0629	114,89972	1,1329	-1,12789

4.3 Overlay of shifted Originals

In section 4.2 it was tried to find the shift between the two observations of LRa01. Field **A** stands for the observation from 17:38 and **B** for the observation from 18:47. Because the CCD has two halves with different noise level, it was decided to use only one half of the field, in this case the left half from pixel 0 to pixel 1079 in x. The method was tried on a field without reduction and the same field with background reduction. The background reduction consisted of setting all pixels with a pixel value of less then 2000 ADU to 0, so the analysis was only performed with the brightest stars.

The difference to the method before is that now that we do more then just compute the center of gravity for each star. We try to find the shift between the two fields using all available data.

Computation for a step size of 1 pixel

The method used here is trying to shift the second image so that the difference between the shifted image \tilde{B} and the first image **A** becomes minimal:

The first step is to identify the shift in pixels for which the value

$$sum_{l,m} = \sum_{i,j} \left| \tilde{b}_{ij,lm} - a_{ij} \right| \quad (4.1)$$

becomes minimal.

\tilde{B} is the shifted image of the second observation with the pixel values $\tilde{b}_{ij,lm}$ for the pixels i and j, shifted by the amount of l and m, $l, m = -5, \dots, 5$, **A** is the original observation. The the value $d = sum_{l,m}$ becomes minimal when the shifted image has the minimal difference to the original image(see Listing C.12).

First we need to shift the image: The computation of a shift with step size 1 pixel is easy because it is just a move of values from one pixel to the next, so no algebraic operation is needed. Then we have to compute $d_{i,j}$. The values i and j for the minimum of $d_{i,j} = d_{min}$ are the optimal shift in x and y.

In this step we use the whole half of the field to compute d . The value of d_{min} for the minimum is shown in table 4.5 for the image with the cut at 2000 ADU and table 4.6 for the image without noise reduction. The source code to determine the shift between two images can be found in Appendix C.12

Computation for a step size of 0.1 pixel and less:

The second step is a little bit more complicated. Now we want to find the best shift in subpixel steps within the pixel (x,y). The shift is done like in section 3.1.2 for a simulated PSF, we shift the image using a spline interpolation.

We create an array of supporting points with a step size of 0.1, the intensity at these step points is now the interpolated value from the spline interpolation. Like this we can move our star again, but now in subpixel steps. To be sure that the coordinate measured in step 1 is correct, not only an interpolation is performed, but the PSF is shifted even further, at most by 2.9 pixels ($\Delta x, \Delta y = 0.1, \dots, 2.9$).

As a result the optimal shift position with an accuracy of 0.1 pixels and again a value for d_{min} is obtained.

The interesting thing is that the value for d_{min} does not decrease significantly if we go into the subpixel region. We can see that a smaller step size for the interpolation does not really increase significantly our knowledge of the best position, because the value of difference d_{min} between a shifted image and an original image does not change significantly by increasing the number of interpolation points. It seems that the background noise level is the limiting factor for the accuracy of this method.

The two parts of code for smaller step sizes can be found in Appendix C.0.14

Table 4.5: minimal total sum of the absolute value of original minus shifted image for images with a cut at 2000 ADU

accuracy	shift (x,y)	d_{min}	computational time
1 px	(-3 ; 2)	3.388.058	ca. 5 sec.
0.1 px	(19/30 ; 15/30)	189.726	ca. 8 sec.
0.05 px	(40/60 ; 31/60)	189.723	ca. 45 sec.
0.025 px	(80/120 ; 64/120)	189.486	ca. 3 min
0.01 px	(201/300 ; 163/300)	189.471	ca. 20 min

Table 4.6: minimal total sum of the absolute value of original minus shifted image without cut

accuracy	shift (x,y)	d_{min}	computational time
1 px	(-2 ; 2)	18.068.872	ca. 5 sec.
0.1 px	(22/30 ; 24/30)	102.636	ca. 8 sec.
0.05 px	(46/60 ; 49/60)	102.394	ca. 45 sec.
0.025 px	(92/120 ; 99/120)	102.310	ca. 3 min
0.01 px	(232/300 ; 249/300)	102.285	ca. 20 min

Result of the method The results from the tables 4.6 correspond to a shift of $-2 + 232/300 = -1.23$ pixels in x and $2 + 249/300 = 2.83$ pixels in y. These values are not distances but have to be seen inverted, they are standing for the amount of shift needed to transform the second observation into the first. That means that the values in distance would be (1.23/-2.83). The result in table 4.5 corresponds to a shift of (-2.33/2.54) pixels in x and y direction, that means we find a distance of (2.33/-2.54) for images with a cut at 2000 ADU.

4.3. OVERLAY OF SHIFTED ORIGINALS

Dermination of the noise level

It was important to estimate the noise level of the background, this was done by computing the mean background intensity value and its sigma of a background area in a chosen empty field. Just to be on the safe side these values for 7 fields were computed, 5 with a size of 30x30 pixels and 2 with a value of 15x15 pixels. This was done for the two observations of LRa01 from 17:38 and from 18:47. The noise levels, the resulting mean values of the areas don't show significant difference between the two observations.

Just as a control we computed also the mean values and variances sigma for smaller areas of 15x15 pixels, the results for these areas show us that we can use the result of smaller areas to say something for the whole observation. The mean values and the sigma values are comparable.

Table 4.7: Noise level for fields in LRa01 at **17:38**

array #	x	y	mean	sigma	total	$\sum_{i,j} a_{ij} - b_{ij} $
1	450:479	600:629	1116,69	7,041000	1005020	6250
2	580:609	480:509	1115,40	6,389120	1003860	7199
3	550:579	1760:1789	1120,60	5,851080	1008540	8204
4	600:629	440:469	1113,03	5,501550	1001730	9289
5	955:984	1705:1734	1121,60	7,777610	1009440	
6	580:595	480:495	1117,52	6,108990	286086	1709
7	450:465	600:615	1116,36	6,319980	285789	

Table 4.8: Noise level for fields in LRa01 at **18:47**

array #	x	y	mean	sigma	total	$\sum_{i,j} a_{ij} - b_{ij} $
1	450:479	600:629	1116,92	6,926800	1005230	6621
2	580:609	480:509	1115,36	6,968220	1003820	9625
3	550:579	1760:1789	1123,16	43,880600	1010840	10246
4	600:629	440:469	1112,86	5,821310	1001580	9456
5	955:984	1705:1734	1121,38	7,997230	1009240	
6	580:595	480:495	1116,49	6,946770	285821	1861
7	450:465	600:615	1115,92	5,425780	285676	

The main reason behind the noise determination is to get a value to judge the quality of the positional algorithm of the "overlay" method. There the sum of the difference between two fields was computed and it was observed where the sum for this shift would be minimal.

If now empty background fields are compared and this sum is computed for multiple areas it becomes possible to analyse the stability of this method. The computation of

the mean value from 4 fields with a size of 30x30 pixels (the same size as used in the overlay method) resulted in a mean value of 7736 ADU for the observation from 17:38 respectively 8987 ADU for the observation from 18:47. The most important value is the variance sigma of these mean values which lies for the observation from 17:38 at **1307** and for the observation from 18:47 at **1613**. This means any minimum found with the "overlay" method was not significant enough to say that the PSF was at this position. The difference between the original and the shifted field from 189.726 to 189.471 of 255 ADU for a accuracy of 0.1 pixels to 0.01 pixels is much smaller then the sigma value of 1613 (see table 4.5). We can state only that we can reach an accuracy of around 0.2 pixels with the "overlay"- method.

Final Result This magnitude in accuracy of 0.2 pixels corresponds to an accuracy of a half arcsecond.

The shift we find between the two observations has for the observation reduced using a cut-off at 2000 ADU a value of $\begin{pmatrix} 2.3 \pm 0.2 \\ -2.5 \pm 0.2 \end{pmatrix}$ and for the unreduced observation a value of $\begin{pmatrix} 1.2 \pm 0.2 \\ -2.8 \pm 0.2 \end{pmatrix}$.

Side note When the sum d of the difference of between two star-less fields A and B is computed,

$$d = \sum_{i,j} |a_{ij} - b_{ij}|. \quad (4.2)$$

A here corresponds to field i and B to field i+1. For both fields only background noise is measured. We get the interesting result, that for a 16x16 pixels field we get a value of $d = 1709$ respectively 1861 which lies close to the value of $16 * 16 * 7$, where **7** is approximately the variance of the pixel values of the background. For a field of 30x30 pixels the difference between the two fields is between 6250 and 9289, which lies close to the theoretical value of $30 * 30 * 7 = 6300$. Perhaps this result can be useful for further methods, but it has not yet been worked out any further.

4.3.1 Does increasing the number of observations of the same field also increase the astrometrical accuracy?

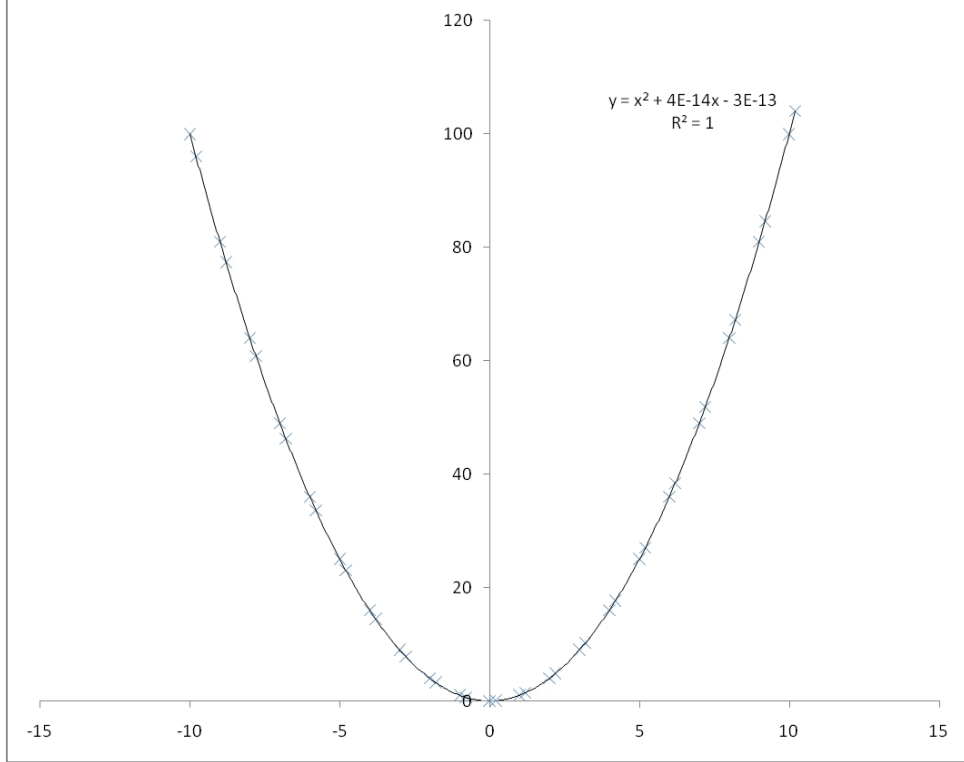


Figure 4.7: Two observations with an perfectly known shift between the observations

One of the questions that should also be answered with this work was whether the accuracy of a determined position is higher if the field is observed multiple times, perhaps even with an interval of 30 seconds over a period of a 120 days.

At first it seems obvious that if we want to fit a model to some data points, and we increase the number of data points, we get a better accuracy for the parameters of our model. In our case we have a set of equidistant data points $x_I(i, j)$ (pixel values) for observation I and a second set of equidistant data points with a different position for observation II. $\vec{x}_{II}(i, j) + \Delta \vec{x}$.

The problem is to derive the value of $\Delta \vec{x}$ exactly. If we cannot determine the amount of shift with enough accuracy we also cannot increase our knowledge of the nature of our model, this fact is illustrated in figure 4.7 and 4.8.

We see that even if we get more data points, we would not get a better fit for our model, in this case a parabola. The problem of position determination is the same from frame to frame, an increased number of observations would not automatically increase our knowledge in the positions of the observed stars. The accuracy of the measured positions is dependent on the accuracy of every frame. Only if we assume that the measured

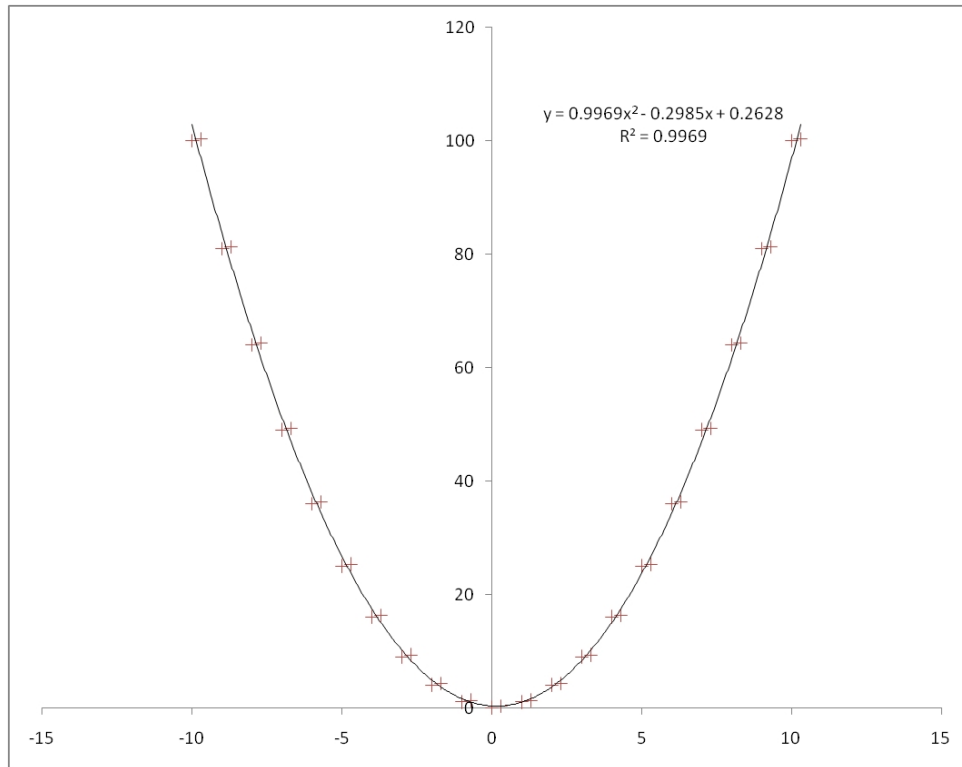


Figure 4.8: Two observations with an inaccurate shift between the observations, we see that data points from a second observation do not increase the accuracy of the curve, the measured curve from only one observation shows even a higher accuracy

position of each star has a certain error distribution, then we could increase our positional knowledge by computing mean positions and mean shifts. It can be assumed that the positions are distributed in a cloud with Gaussian distribution and a sigma which depends on the noise level of the observations. So this noise would be found in all positional data. Perhaps some frequency filtering (Wiener filter) could help to clean that noise from the positional data, but this idea has not been tried further.

We see that even if we get more data points, we would not get a better fit for our model, in this case a parabola. The problem of position determination is the same from frame to frame, an increased number of observations would not automatically increase our knowledge in the positions of the observed stars. The accuracy of the measured positions is dependent on the accuracy of every frame. Only if we assume that the measured position of each star has a certain error distribution, then we could increase our positional knowledge by computing mean positions and mean shifts. It can be assumed that the positions are distributed in a cloud with Gaussian distribution and a sigma which depends on the noise level of the observations. So this noise would be found in all positional data. Perhaps some frequency filtering (Wiener filter) could help to clean that noise from the positional data, but this idea has not been tried further.

4.3. OVERLAY OF SHIFTED ORIGINALS

Chapter 5

Parallax measurement, data reduction - a theoretical view

How to derive real parallaxes from a data set? Some authors already described the problems involved, first some projection effects have to be taken into account if we take pictures of the sky by a planar detector (CCD or photo plate). Second we must reduce the measured angles to parallaxes. Here we have the problem that our field moves already from observation to observation. If we observe different fields and if we want to reduce the data to global coordinates e.g. if we want make a catalog of the positions of these stars, we must take into account some side effects that change the apparent position of the observed stars. One effect is the relativistic light deflection (up to 1.7 arcseconds close to the sun [15]), another effect is the aberration caused by the movement of the earth and the sun relative to the observed field. Also effects like earths nutation cause a change in the position of the star field relative to our classical coordinate systems. But because we have only observations with the same instrument of the same field and in the same direction we can exclude all of these effects from our work.

The effect of aberration can be seen as constant over the whole field, the effect of relativistic light deflection is also not important because no observations are taken towards the direction of the sun. Only the effect of light deflection caused by the mass of the earth should perhaps be considered, because it the relative deflection caused by earth is already $\cong 40 \mu\text{asec}$.

Effects like the nutation can be neglected, because they only change the position of the field relative to the coordinate system earth.

5.1 Gnomonic Projection

First the formulas needed for a gnomonic projection are described (from: Jean Kovalevsky, Modern Astrometry, pp 73):

The transformation from local planar coordinates (like a CCD, coordinates A, B') to celestial coordinates α , δ is what we are interested in.

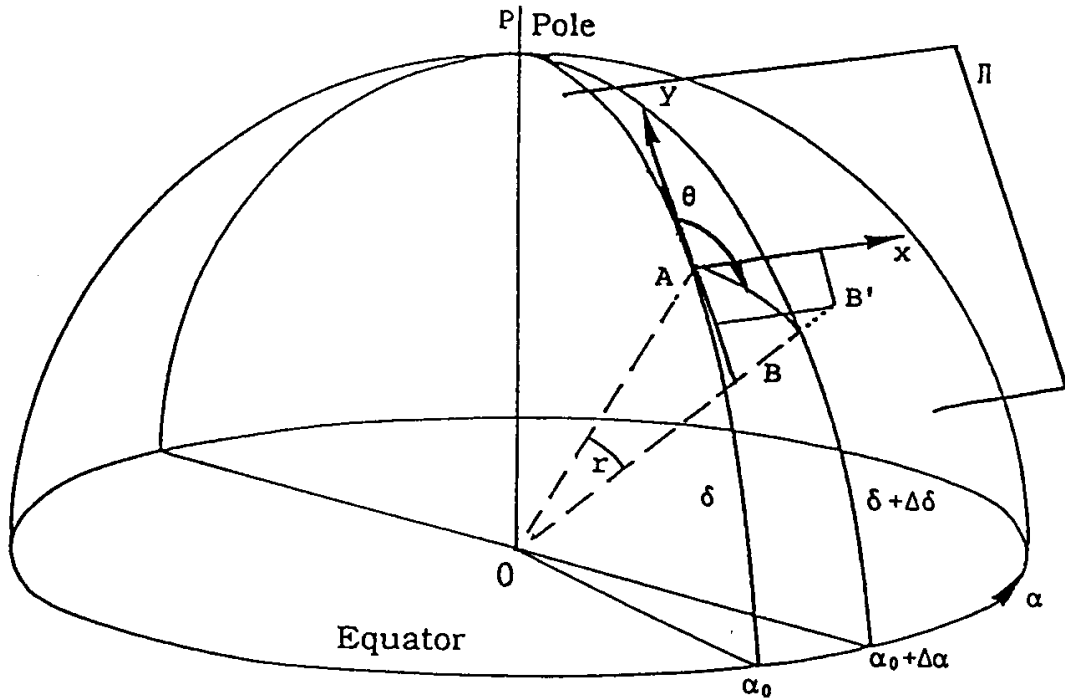


Figure 5.1: Gnomonic projection

The differential coordinates $\Delta\alpha$, $\Delta\delta$ from a measured plate must be transformed into linear coordinates.

We have :

$A\vec{x}$ is tangent to the declination small circle and $A\vec{y}$ is tangent to the celestial meridian, the positive direction is pointing toward the north. This plane is forming a system of coordinates, the so called "standard coordinates".

The transformation from differential coordinates ($\Delta\alpha$, $\Delta\delta$) to standard coordinates (\vec{x}, \vec{y}) is called **gnomonic** or **central projection**. We have two observational points A and B with the point of intersection A and the measured position B on the sphere,

$$B = \begin{pmatrix} \alpha + \Delta\alpha \\ \delta + \Delta\delta \end{pmatrix} \quad (5.1)$$

B' is the projection of B in the tangential plane set up by $A\vec{x}$ and $A\vec{y}$.

$$B' = A + \begin{pmatrix} x \\ y \end{pmatrix} \quad (5.2)$$

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \tan r \sin \phi \\ \tan r \cos \phi = x \cot \phi \end{pmatrix} = \begin{pmatrix} \frac{\sin \Delta\alpha}{\cos \delta (\tan \delta \tan(\delta + \Delta\delta) + \cos \Delta\alpha)} \\ \frac{\tan(\delta + \Delta\delta) - \tan \delta \cos \Delta\alpha}{\tan \delta \tan(\delta + \Delta\delta) + \cos \Delta\alpha} \end{pmatrix}. \quad (5.3)$$

These terms for x and y can be developed in powers of $\Delta\delta$ and $\Delta\alpha$:

$$x = \Delta\alpha \cos \delta - \Delta\alpha \Delta\delta \sin \delta + \frac{\Delta\alpha^3}{6} (3 \cos^3 \delta - \cos \delta) + \dots \quad (5.4)$$

$$y = \Delta\delta + \frac{\Delta\alpha^2}{2} \sin \delta \cos \delta + \frac{\Delta\delta^3}{3} + \frac{\Delta\alpha^2 \Delta\delta}{2} (\cos^2 \delta - \sin^2 \delta) + \dots \quad (5.5)$$

5.2 Displacement due to annual parallax

The next formalism shown is the displacement due to annual parallax, also i use the formalism from Jean Kovalevsky (Modern Astrometry, pp 74:

Let us state that:

$$\vec{B}\vec{E} = R\vec{u} \quad (5.6)$$

with the definitions:

\vec{E} : Center of Earth

\vec{B} : barycenter of the solar system

\vec{S} : Star

$r = |\vec{S}|$: distance to star

$$\bar{\omega} = a/r. \quad (5.7)$$

Respectively for $a=1$ $\omega = 1/r$: the parallax of the star

\vec{u} : unit vector of $\vec{B}\vec{E}$

For the movement of the earth around the sun we can say:

$$R\vec{u} = a\vec{u}(1 + f(t)). \quad (5.8)$$

The parallactic displacement on the unit sphere is:

$$\Delta\vec{s} = \vec{s}' - \vec{s} = \vec{B}\vec{E}, \quad (5.9)$$

which can also be written with the outer vectorial product:

$$\Delta\vec{s} = s \wedge \left(s \wedge \frac{R\vec{u}}{r} \right) \quad (5.10)$$

or, when we think of the annual movement of the Earth

$$\Delta \vec{s} = \bar{\omega} \vec{s} \wedge (\vec{s} \wedge \vec{u}) (1 + f(t)). \quad (5.11)$$

In equatorial coordinates (L:longitude of the sun, ϵ : obliquity of the ecliptic) we get

$$\cos \delta \Delta \alpha = \bar{\omega} (1 + f(t)) (\cos \epsilon \cos \alpha \sin L - \sin \alpha \cos L) \quad (5.12)$$

and

$$\Delta \delta = \bar{\omega} (1 + f(t)) (\sin \epsilon \cos \delta \sin L - \sin \delta \cos \alpha \cos L - \cos \epsilon \sin \delta \sin \alpha \sin L). \quad (5.13)$$

The longitude of the sun can be computed following the formula on page C24 of the *Astronomical Almanac*. The obliquity can be computed using Jean Meeus: *Astronomical Algorithms*, p. 135.

This formalism allows us to understand the parallactic displacement correctly.

5.3 Methods to reduce star positions of photographic plates

Also it shall be described how parallaxes could be measured using data from photographic plates. Here especially scale factors and also projections, but also the choice of reference stars is important.

We also have a problem with our CoRoT observations, because we need to have a sufficient number of reference stars, bright enough to be detected and still so far away that they show no parallactic ellipse themselves or with very well know parallaxes, proper motions, etc.. The problems are similar to the ones found by reducing positions on photographic plates.

A short overview of persons and methods: **F. Schlesinger**[34], proposes already in 1899 to use photographic plates to measure parallaxes. Until that time parallaxes have only been measured using transit instruments and the measurement has been made by human eye and a clock as a reference system. In [10],1932 A.Deutsch shows that the relations between the rectangular coordinates of two plates can be expressed by linear functions depending on the Turner-Method [40] as described by F.W.Dyson, 1894. 1973, Conard Dahn describes his use of measuring machines to derive parallaxes for the central star of the planetary nebula NGC 7293, [8]: he speaks of the methods proposed by Frank Schlesinger [34], 1899, to measure parallaxes. He used reduction techniques, described in [31], 1970. Norris Russell [33], 1910 also describes the use of the Turner linear formula to derive parallaxes at the Cambridge observatory and the use of comparison-stars and their distribution about an observed star. It is also very interesting to see how **J.C. Kapteyn** solved the problem of the unknown parallaxes of reference stars with a formula for the mean parallax for stars of a given magnitude [21].

The Dependences Method

But lets begin with the description of the dependences method of **A. König**: "The method of dependences, given by Schlesinger (1926), is without doubt the most elegant and the one most often applied ...". Its main advantage is though it can be applied to only two reference stars, it is possible to use three or more stars. The coordinates x, y of an object are linear functions of the coordinates of the reference stars x_i and y_i . So for three stars we have

$$x = D_1x_1 + D_2x_2 + D_3x_3, y = D_1y_1 + D_2y_2 + D_3y_3 \quad (5.14)$$

with D_i the so called dependences, they obey the condition:

$$\sum D_i = 1 \quad (5.15)$$

They represent geometrically the corners of a triangle formed by the reference stars with the object inside the triangle. They are independent of the coordinate system. The relations 5.14 are valid for measured and tangential coordinate systems, but not for spherical coordinates or their differences. In the transformation to the sphere (see 5.1) it is very easy to apply them.

The other method to derive positions of stars from reference stars is the simple interpolation method, here we only work with spherical differences $\Delta\alpha$ and $\Delta\delta$ on one hand, and the differences of the measured coordinates Δx and Δy on the other hand. One simply establishes a relation that $\Delta\delta$ and $\Delta\alpha$ are expressed in the same units. That is what is usually done today when coordinates of stars on CCD's are reduced.

A.Deutsch [10] describes the dependences method more detailed:

The relation between the orthogonal coordinates of two plates with close central points (the observations are of the same region and directing along the same line of sight). can be expressed by linear functions: x and y are the orthogonal coordinates on a plate, X and Y the coordinates on a second plate. We have the equations

$$ax_i + by_i + c = X, dx_i + ey_i + f = Y \quad (5.16)$$

with $i=1,2,3$ for the 3 reference stars and

$$X_0 = ax_0 + by_0 + c, Y_0 = dx_0 + ey_0 + f \quad (5.17)$$

for the unknown star.

Composing 5.16 and 5.17 we get:

$$X_0 = \frac{\begin{vmatrix} X_1 & y_1 & x_0 \\ X_2 & y_2 & x_0 \\ X_3 & y_3 & x_0 \end{vmatrix} + \begin{vmatrix} x_1 & X_1 & y_0 \\ x_2 & X_2 & y_0 \\ x_3 & X_3 & y_0 \end{vmatrix} + \begin{vmatrix} x_1 & y_1 & X_1 \\ x_2 & y_2 & X_2 \\ x_3 & y_3 & X_3 \end{vmatrix}}{\begin{vmatrix} X_1 & y_1 & 1 \\ X_2 & y_2 & 1 \\ X_3 & y_3 & 1 \end{vmatrix}} \quad (5.18)$$

5.3. METHODS TO REDUCE STAR POSITIONS OF PHOTOGRAPHIC PLATES

and an analog formula for Y_0 .

The right side of these equations is known and after some transformations we can write

$$X_0 = \frac{N_1 X_1 + N_2 X_2 + N_3 X_3}{N} \quad (5.19)$$

with

$$N = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} = N_1 + N_2 + N_3$$

if we define now $N_i/N = D_i$ we get: $D_1 + D_2 + D_3 = 1$,

$$X_0 = D_1 X_1 + D_2 X_2 + D_3 X_3$$

and analog

$$Y_0 = D_1 Y_1 + D_2 Y_2 + D_3 Y_3$$

Peter van de Kamp [20] points to another interesting fact: any set of faint background stars represents an acceptable close approximation to a fixed background. Until now this point has not been taken into consideration but is very important, the background stars that CoRoT would deliver are very often smeared with other stars. The fainter the star becomes the higher the probability that another stars PSF is interfering. When we take a look at the observations we also see that we do not have many background stars in the field which are easy enough to identify. Van de Kamp also points out that its not necessary to use more then four reference stars per observed "parallax" star, because that would not increase the reached accuracy generally. Also he points out that the "classical" parallax determination is based on twenty to thirty plates spread over five to seven successive observing seasons. with errors of ± 0.01 arcseconds. A value we could not reach with CoRoT at the momentary state of data reduction.

The Turner Method As described in [47], the reduction of positions using Turners method [39] is comparable to the dependencies method of Schlesinger, it is necessary for the Turner method to choose three reference stars around the object. The formula looks as follows:

$$\xi_i = \tan(a_i - A) \sin q_i \quad (5.20)$$

$$\eta_i = \tan(P - q_i) \quad (5.21)$$

$$\tan(q_i) = \tan p_i \cos(a_i - A) \quad (5.22)$$

ξ_i, η_i are the orthogonal coordinates in rectascension and polar distance (respectively declination). a_i and P_i are the rectascension and polar distance of the three reference stars. A and P are the assumed positions of the object in in rectascension and polar distance. All coordinates are relatively to the position of the observed object as central point. The values

for the position of the object are only preliminary and often taken from older catalogs or former measurements.

From the equations

$$\xi_i = ax_i + by_i + c \quad (5.23)$$

and

$$\eta_i = dx_i + ey_i + f \quad (5.24)$$

where x_i, y_i are the measured orthogonal coordinates of the three stars, related to the object, we can derive the constants a,b,c,d,e and f, when we say $\xi_1 - \xi_2 = \xi_{12}$, etc. :

$$a = \frac{\xi_{12}y_{23} - \xi_{23}y_{12}}{x_{12}y_{12} - x_{23}y_{12}}$$

$$b = \frac{\xi_{12} - ax_{12}}{y_{12}}$$

$$d = \frac{\eta_{12}y_{23} - \eta_{23}y_{12}}{x_{12}y_{23} - x_{23}y_{12}}$$

$$e = \frac{\eta_{12} - dx_{12}}{y_{12}}$$

$$c = \xi_1 - ax_1 - by_1 = \xi_2 - ax_2 - by_2 = \xi_3 - ax_3 - by_3$$

$$f = \eta_1 - dx_1 - ey_1 = \eta_2 - dx_2 - ey_2 = \eta_3 - dx_3 - ey_3$$

By that way we get the wanted position of the object:

$$\alpha_{Object} = A + A_1$$

$$\delta_{Object} = D + D_1$$

where $A_I = \frac{1}{15}c \sec \delta$ and $D_1 = f$

About the problem to solve optical problems analytically and empirically

Heinrich K. Eichhorn writes something very interesting about the two schools of astrometric reduction from in [12] (pp 73), 1970: "There are in principle two schools of thought how to take into account the deviations from gnomonic projection in astrometric work. One could for instance, carefully investigate the functional form of all possible deviations from gnomonic projection and accordingly modify the equation. The parameters describing these effects would then have to be determined either in the course of the adjustment or by extraneous methods. This is the approach of the *German School* of photographic astrometry, founded essentially by F. Küstner. In modern times, much of what has been known to astrometrists for decades, including matter described in König's (1933) article, was independently developed by analytical photogrammetrists (compare

5.3. METHODS TO REDUCE STAR POSITIONS OF PHOTOGRAPHIC PLATES

Brown[1965, 1968]). The other possibility is not to analyze the individual sources of that contribute deviations from the ideal gnomonic projection, but to generalize the formula of gnomonic projection by correction parameters. ... It must then be decided empirically which of the parameters are to be included in the model. This approach ... (the so-called six-constant model, the six constants are also known as plate constants) was mainly advocated and applied in large scale by F.Schlesinger, the founder of the *American School* of photographic astrometry.“

5.3.1 Deriving of celestial coordinates of stars in the CoRoT fields using reference stars

From the field LRa01 we have the x-, and y-coordinates of some stars and want to try to derive celestial positions using the dependences algorithm and the turner method. We identified the stars using DS9 [50], version 5.6.

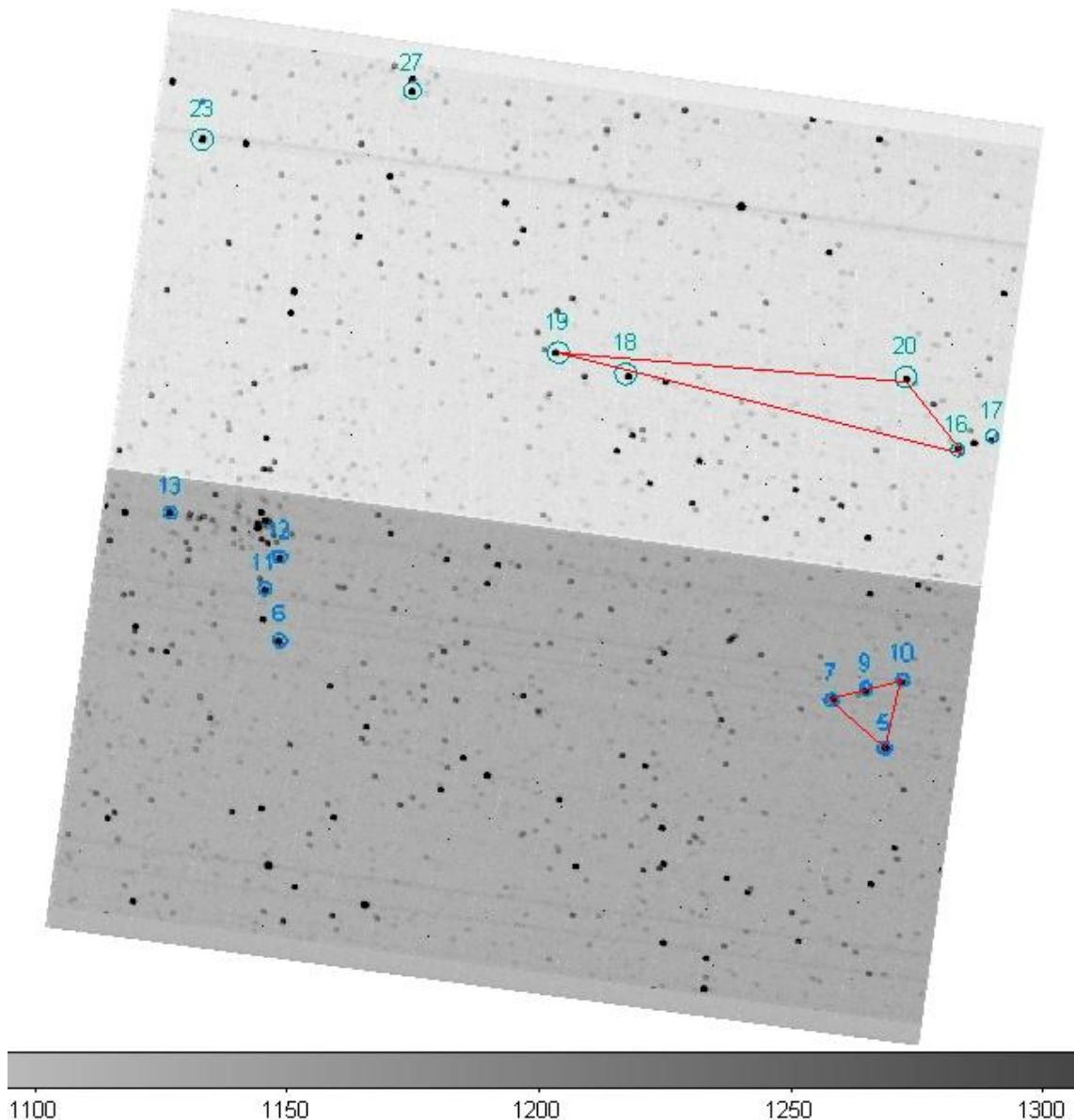


Figure 5.2: Field LRa01 with some identified stars

The identified stars in the CoRoT field can be seen in figure 4.6. These stars have been identified and compared with catalogue data. These literature

5.3. METHODS TO REDUCE STAR POSITIONS OF PHOTOGRAPHIC PLATES

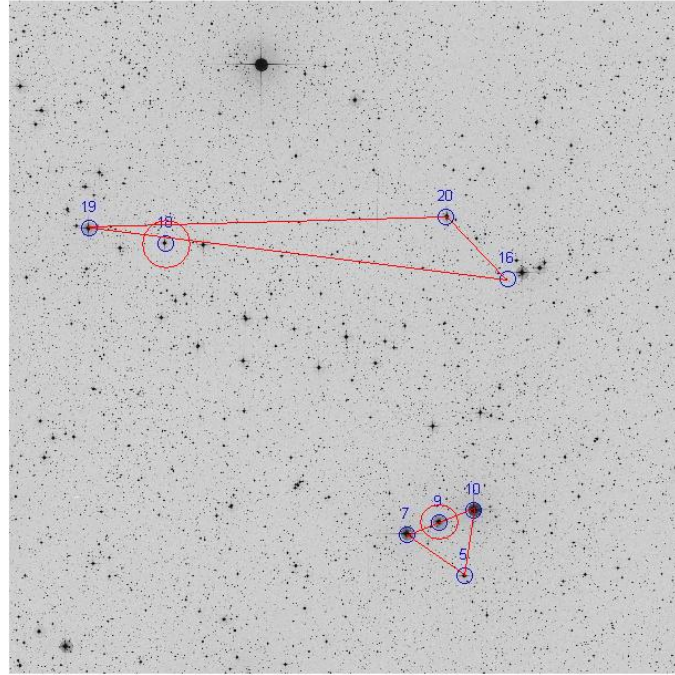


Figure 5.3: The same stars identified in DSS data using SIMBAD

positions are taken from the SIMBAD database ([45] and are based on the measurements of the Hipparcos Satellite [28] (see table 5.1). The same centroid method as described in chapters 2.1.2 , 3.2.1 and 4.2 to derive the position of the stars is used.

Table 5.1: Some stars identified in LRa01, catalog data, and centroid positions x and y

No.	Name	RA	Dec	x	y
5	HD 292297	101.962457	0.2420525	679.16707	1883.834
7	HD 49385	102.04793	0.3049731	774.18959	1749.0629
9	HD 292296	102.000281	0.3220531	804.10185	1821.6459
10	HD 49330	101.948147	0.3401889	834.08549	1900.9467
16	HD 49224	101.876811	0.6929617	1392.7075	1959.0195
18	HD 289230	102.407913	0.7360883	1456.8434	1176.0232
19	HD 49765	102.521107	0.7584714	1487.9619	1002.6924
20	HD 49330	101.988616	0.7761169	1538.0846	1821.8891

For the stars No. 9 and No. 18 the Turner method has been applied to using the x - and y -positions derived with the centroid algorithm. The reference stars for the two tests are for star No. 9: 5, 7 and 10 and for star No.18 the stars 16, 19 and 20 were used as reference stars.

As a result we get for both stars new improved positions. Here we used the image without cut-offs to compute the centroid positions.

Star number 9 (HD 292296) The position in the SIMBAD database and the measured position show a difference of 0.0019 arc seconds in rectascension and a difference of 2.04 arc seconds in declination. We see that the accuracy is better then in the case for more distant reference stars.

Table 5.2: Turner method applied to star No.9 (HD 292296)

	RA	Dec
measured	102.0002807	0.32148487
literature	102.0002812	0.3220531
difference in degrees	-5.41873E-07	-0.00056823
difference in arc sec	-0.001950742	-2.04562929

Star number 18 (HD 289230) The position in the SIMBAD database and the measured position show a difference of 0.01 arc seconds in rectascension and a difference of 9.89 arc seconds in declination. For star number 18 the distance between the reference stars is

Table 5.3: Turner method applied to star No.18 (HD 289230)

	RA	Dec
measured	102.4079168	0.733338539
literature	102.4079133	0.7360883
difference in degrees	3.46321E-06	-0.002749761
difference in arc sec	0.012467571	-9.899140437

much larger then for star No.9, this is probably the reason why the accuracy is also much worse.

Conclusion We can say that the Turner algorithm worked, mainly because the description in [47] is very detailed and easy to follow. As a result of this test we can say that an accuracy of **0.0019 arc seconds** could be reached in the declination, but we also have to say that in rectascension there is a much bigger difference between the measured position and the literature value. This test should be done with more stars. It seems important to choose reference stars close to the target star to reduce inaccuracies. Also the effects influencing the Turner method should be understood much better, especially the difference in accuracy between the declination and the rectascension is not properly understood right now.

The Dependences method applied to the field LRa01 When the dependences are computed as described in [10] the inevitable condition that the sum of all dependencies must be 1 is not fulfilled, probably the coordinates must be computed relatively to the object. The point of origin must probably be the object's position.

5.3. METHODS TO REDUCE STAR POSITIONS OF PHOTOGRAPHIC PLATES

The Turner algorithm can be found in Appendix C.15.

Chapter 6

Conclusion

Finally we must say that its very challenging to derive parallaxes from such a complicated optical system like CoRoT without better simulations. We have tried to solve which algorithm from the ESO/MIDAS package is best to derive positions, but found only that the CENTER/GAUSS algorithm is really inaccurate to derive positions of the CoRoT PSF. We examined simulation of shifted positions without noise to show the numerically possible accuracy in the subpixel range, we find that due to the limited number of data points (pixels) and the lack of analytical knowledge of a detailed shape of the PSF, we can only reach an accuracy of 0.025 pixels or 62,5 mas.

After working with the simulations, we received observations from CoRoT, these observations have been analyzed. As a result of these analysis we could show that the positions on one frame can be derived with an accuracy of up to 0.2 pixels corresponding to an angular accuracy of 0.5 arcseconds. We showed that an increase in the number of observations does not of necessity increase the astrometrical accuracy. To test the reachable accuracy in celestial coordinates the Turner method [39] has been applied using two groups of stars in CoRoT field LRa01 as reference stars. The position of two stars could be derived with an accuracy of up to 0.0019 arcseconds. But this test should be made with a higher number of stars to be able to estimate the achievable accuracy correctly. Due to the already shown limitations in the accuracy of position measuring we strongly believe that the effect of trigonometric parallaxes could only be measurable for the nearest stars, depending on our understanding of the background noise level.

If we could reduce the noise completely and if we had stable images, and we only had jitter changing the position of the PSF within a pixel, we would have the accuracy of 62.5 mas per observation, assuming we can take 1000 observations of the same star field (CoRoT would do this in two days) we can reach an accuracy of $62.5/\sqrt{1000} = 1.97\text{mas}$.

Outlook

We always should have in mind the scientific goal we are looking for, but at the same time it is extremely worthy to look at the things already done some generations ago. The

problems solved for the astrometric reduction of photo plates are also a good starting point to solve the problems of modern astrometry with CCD's. We will see a further increase in the area size of CCD's and the larger these areas become, the more the problems come back we had with photographic plates. Astrometry is one of the oldest branches in Astronomy, but still one of the most exciting ones, and we hope to see many very fascinating improvements in our view of the world by the next astrometric missions like GAIA and others. We want to give the last word to S.A. Mitchell, the Second Director of McCormick Observatory. He said: "Parallax work costs a great many hours loss of sleep, and it is therefore a difficult research for one to carry on who in addition to this work must engage in university teaching."

Chapter 7

Danksagung

Ich möchte danken: Meinen Eltern, besonders meiner Mutter, die immer an mich geglaubt hat, ohne deren Unterstützung und Vertrauen diese Arbeit, das Studium, alles was ich bin nie möglich gewesen wäre. Danke auch an meine Patentante Monika und ihrem Mann Manfred. Ihr habt mir mehr geholfen, als ihr es vielleicht ahnt, Ihr wart da, als ich nicht mehr weiter wusste. Caroline möchte ich danken, für Unterstützung und Hilfe, für Ratschläge und wichtige Erfahrungen, und dafür dass Du meiner Mutter eine wichtige Freundin bist. Ich möchte meinen Freunden danken, Lothar, Marc, Damien, ohne Euch wäre ich nie dahin gekommen wo ich jetzt bin. Danken möchte ich auch Dr. Hannes Schneiderbauer, unsere Freundschaft hat mich sehr bereichert, danke dafür.

Dr. Dimitris Sinachopoulos möchte ich danken für erste Vorarbeiten und Einführung in die Tiefen der MIDAS-Programmierung.

Thanks to Rachel Drummond for help in understanding the "real" positions of the simulations. Prof DDr. W.Weiss for financial support and for providing the simulated data, and ideas like the symmetry algorithm. Thanks to the danish CoRoT-Team around Torben Arenthoft, you are one of the people i am very happy to have met during my diploma work. Je veux remercier Michel Auvergne de la CNES, le CoRoT Instrument Scientist, pour les données très importants et véritablement précieux pour moi, les simulations au debut et quelques ans plus tard les observations "réels" de CoRoT, vous m'avez permet de finaliser ce travail, merci beaucoup.

I want to thank Professor Maria G. Firneis for more then only scientific support, I learned a lot from you and I hope will learn even more, you are an incredible "great attractor", on the one hand like a library for age-old knowledge and on the other hand working on brand-new ideas, some of them too new for other scientists to even express them openly.

Thomas Kallinger and Alexander Kaiser, for beeing colleagues and friends, i never will forget our barbecue in the Starkfriedheim. I want to thank my friends from the Astronomical Institute in Vienna: Harald Leibinger, Fabian Schierscher, Paul Beck and Bernhard Brinninger "Brinni", for sharing the same ideas and for spending the nights together. Alexander Bindeus, and Andreas Chwatal, for being friends and sharing more

passions then just the passion for astronomy.

I also want to thank my company Silver Server, especially Bert, they made it possible for me to earn some money and at the same time study, the same thanks go to the Buschenschank Pötzsleinsdorf and to Hushmand at the Shiraz.

I also want to thank DDr. Thomas Posch, thanks for open job opportunities for students, from science to scientific outreach, you are open for anyone who is interested in astronomy and that makes it a pleasure to work with you and to learn from you.

Last but not least I deeply want to thank Bitu, my love, you gave me back the belief in the future, the belief in myself. You gave me the support I needed. You were behind me all the time. And at the same time you are in front of me, you are my future, along with all the dreams we share, you reminded me how curiosity can make life so interesting and free and how important it is to be loved and to love somebody. You showed me how important it is to have a goal in life and how important it is to work for that goal.

Bibliography

Papers, Books

- [1] W. F. van **Altena**, J. T. Lee, E. D. Hoffleit, 1995: Yale Trigonometric Parallaxes, Fourth Edition (van Altena+ 1995)
- [2] Friedrich **Becker**, 1966: Einführung in die Astronomie, Bibliograph. Inst., BI-Wissenschaftsverl., p. 54 ff
- [3] Ludwig **Bergmann**, Clemens **Schaefer**, Lehrbuch der Experimentalphysik, Band 3, Optik, 10.Auflage
- [4] Max **Born**, Emil **Wolf**, 2003, Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light (7th Edition), ISBN 0-521-642221, Cambridge University Press
- [5] Jos de **Bruijne**, 2006: Gaia - Taking the Galactic Census, 2006-02-13 (Rev. 1) http://www.rssd.esa.int/SA/GAIA/docs/info_sheets/IN_accuracy_section.pdf
- [6] Roger **Ceragioli**, 2005: A survey of Refractive systems for astronomical telescopes by Roger Ceragioli, Website from June 14, 2005, now removed by the author
- [7] **CNES** Press Release 48, 2003
- [8] Conard **Dahn**, 1993: Trigonometric parallax determination for the central star of the planetary nebula NGC 7293, Publ. Astron. Soc. Pacific 85, 224
- [9] **M.Deleuil**: News from the Corot Space Mission and first results, in preparation.
- [10] **A.Deutsch**, 1932: Bemerkungen zu F.Schlesingers Methode der "dependencies", 1932, Astronomische Nachrichten, Band 246, 5881, p.12
- [11] **Digitized Sky Survey (DSS)**: The Digitized Sky Survey was produced at the Space Telescope Science Institute under U.S. Government grant NAG

BIBLIOGRAPHY

- W-2166. The images of these surveys are based on photographic data obtained using the Oschin Schmidt Telescope on Palomar Mountain and the UK Schmidt Telescope. The plates were processed into the present compressed digital form with the permission of these institutions.
- [12] Heinrich K. **Eichhorn**, 1974, Astronomy of star positions, ISBN: 0804441871
- [13] M.W.**Feast**, R.M.**Catchpole**, 1997, The Cepheid period-luminosity zero-point from HIPPARCOS trigonometrical parallaxes, Monthly Notices of the Royal Astronomical Society, Volume 286, Issue 1, pp. L1-L5.
- [14] M.**Fridlund**, A.**Baglin**, J.**Lochard** and L.**Conroy**: Proceedings of "The CoRoT Mission Pre-Launch Status - Stellar Seismology and Planet Finding" (ESA SP-1306). ISBN 92-9092-465-9., p.353
- [15] M. G. **Firneis**, Optimale Dimensionierung optischer Apparaturen zur Untersuchung von Ausgleichsansätzen zur Bestimmung der relativistischen Lichtablenkung, Ann. Univ.-Sternw. Wien, Band 31, Nr. 4, p. 65 - 234
- [16] The **Gaia** mission: science, organization and present status, Proceedings of the International Astronomical Union, 2007 , 3 : 217-223 Cambridge University Press, <http://sci.esa.int/science-e/www/object/index.cfm?fobjectid=33840>
- [17] Stefan **Hippler** ,Andrei Tokovinin, 2009: Adaptive Optik Online am Max-Planck-Institut für Astronomie in Heidelberg http://www.mpia.de/homes/hippler/AOnline/ao_online.html, May 2009
- [18] Michael **Hoskin**, 1966: Stellar Distances: Galileo's Method and its subsequent History, IJHS, Indian Journal of History of Sciences, 1.1 (1966),pp. 22-29, http://www.new.dli.ernet.in/rawdataupload/upload/insa/INSA_1/20005aef_22.pdf
- [19] Louise F. **Jenkins**, 1952: General catalogue of trigonometric stellar parallaxes
- [20] P. van de **Kamp**, 1962: Astrometry with long-focus telescopes, in "Astronomical Techniques" (ed. by W. A. Hiltner) Chicago-London (1962),pp. 487 - 536
- [21] J. C. **Kapteyn**,1910: List of parallax determinations compiled by Prof. J. C. Kapteyn and Dr. H. A. Weersma, Publ. of the Astronomical Laboratory of Groningen, No.8, p.24
- [22] A. **König**, Astrometry with Astrographs, 1964: Astronomical techniques. Edited by William A. Hiltner. Stars and Stellar Systems. Published by the University of Chicago Press, Chicago, IL USA, 1964, p.461

BIBLIOGRAPHY

- [23] Jean **Kovalevsky**, 2001: Modern Astrometry, Second Edition, ISBN 3-540-42380-X, Springer-Verlag
- [24] R. L. **Kurucz**, 1970: Atlas: a Computer Program for Calculating Model Stellar Atmospheres, SAO Special Report #309 (1970) and Model Atmospheres, VizieR On-line Data Catalog: VI/39. Originally published in:
- [25] H. **Lohninger**: Angewandte Mikroelektronik, 1990, Band 1, IWT-Verlag, als elektronische Kopie: http://www.pixlibris.com/d/download_angewmikrel.html
- [26] Jaymie **Matthews**, 1998: "Stellar Seismology from Space", Cassiopeia, newsletter of CASCA (the Canadian Astronomy Society / Societe Canadienne D'Astronomie), March 1998, <http://ams.astro.univie.ac.at/?s=most>
- [27] E. **Michel**, 2005: COROT - CW9, 12/05 ESTEC
- [28] M.A.C **Perryman** et al, 1997: The Hipparcos Catalogue, Astron. Astrophys., 323, L49-L52 (1997)
- [29] V.V. **Podobed**, 1962: Fundamental Astrometry, English Edition by A.N. Vysotsky, p. 172, Library of Congress Catalog Card Number 64-15810, The University of Chicago Press
- [30] Martin **Rickenbacher**, 2006: Die Basismessungen im Grossen Moos zwischen Walperswil und Sugiez. Cartographica Helvetica, Heft 34, Juli 2006, SGK- Schweizerische Gesellschaft für Kartographie
- [31] R.K. **Riddle**, 1971: First catalog of trigonometric parallaxes of faint stars, Riddle, R. K.; Priser, J. B. Publications of the United States Naval Observatory, 2d ser., v. 20, pt. 3, Washington : U.S. G.P.O., 1971, 39 p. : ill. ; 29 cm.
- [32] David H. **Roberts**, 1987: Time Series Analysis with CLEAN. I. Derivation of a Spectrum, Astronomical Journal, Vol.93, Number 4
- [33] Henry Norris **Russell**, 1910: Determination of Stellar Parallax, Astronomical Journal, Vol. 26, Number 8
- [34] Frank **Schlesinger**, 1899: Suggestions for the Determination of Stellar Parallax by Means of Photography, Astrophysical Journal, vol. 10, p.242
- [35] D. **Sinachopoulos**, 2002: Astrometric and photometric accuracy from simulated DIVA-Skymapper data

BIBLIOGRAPHY

- [36] How Precise is **SIM**? : <http://eis.jpl.nasa.gov/planetquest/Navigator/material/SIMSciOperations.ppt>, Ressources: NASA
- [37] Microarcsecond precision opens a new window to a multitude of phenomena observable with **SIM**: <http://planetquest.jpl.nasa.gov/images/precision.pdf>, June 2005, Ressources: NASA
- [38] **SIM** current mission status: <http://planetquest.jpl.nasa.gov/SIM/SIMCurrentMissionStatus.cfm>, 05/2005, Ressources: NASA
- [39] H.H. **Turner**, 1893: Preliminary notes on the reduction of measures of photographic plates, MNRAS Vol. 54, pp. 11
- [40] F.W.**Dyson**, 1894: Note on Professor Turner's paper on the reduction of measures of photographic plates, Monthly Notices of the Royal Astronomical Society, Vol. 54, p.573
- [41] W.W.**Weiss**, H. Jenkner and H.J. Wood, 1978: A statistical approach for the determination of relative Zeeman and Doppler shifts in spectrograms, A&A 69, pp. 247-257 (1978)
- [42] A. **Unsöld**, B. Baschek, 2002: Der neue Kosmos, 7.Auflage, ISBN 3-540-42177-7
- [43] Thierry **Viard**, 2000: The Corot Telescope
- [44] **VISAT** - Vienna Selection of Astronomical Targets, Communications in Asteroseismology, vol. 143, p. 43-51, <http://ams.astro.univie.ac.at/visat/>
- [45] M. **Wenger** et al, 2000: The SIMBAD astronomical database. The CDS reference database for astronomical objects, Astronomy and Astrophysics Supplement, v.143, p.9-22
- [46] **E. Wolf**, 1951: The diffraction theory of aberrations, Rep. Prog. Phys. 14 95
- [47] **Max Wolf**, 1913-1919 : Photographische Positionsbestimmung von 356 Schultzschen Nebelflecken, Veröffentlichungen der Grossherzoglichen Sternwarte zu Heidelberg, Band 7, No. 6., 1913-1919, pp. 146

Software

- [48] Microsoft **Excel** 2003, Microsoft Excel Online Help

BIBLIOGRAPHY

- [49] Walter W. **Jones**, 1976: spline.pro, Naval Research Laboratory, Sept 26, 1976. Reviewer: Sidney Prahl, Texas Instruments. Adapted for IDL: DMS, Research Systems, March, 1983.
- [50] **SAOIMAGE**, Smithsonian Astrophysical Observatory, Bulletin of the American Astronomical Society, Vol. 22, p.935, <http://hea-www.harvard.edu/RD/ds9/>
- [51] Everything you always wanted to know about **MIDAS**, 1996, Software Report: ESO-MIDAS, European Southern Observatory, BAAS 1996, Vol. 28, pp. 981+, 1996BAAS...28..981 , <http://www.sc.eso.org/~ohainaut/midas.html>
- [52] **ZEMAX**: Software For Optical System Design , <http://www.zemax.com/>
- [53] Chromatic Aberration, <http://en.wikipedia.org/w/index.php?title=File:Lens6a.svg&oldid=129245100>
- [54] Coma, [http://en.wikipedia.org/w/index.php?title=Coma_\(optics\)&oldid=260533913](http://en.wikipedia.org/w/index.php?title=Coma_(optics)&oldid=260533913)

Technical Datasheets

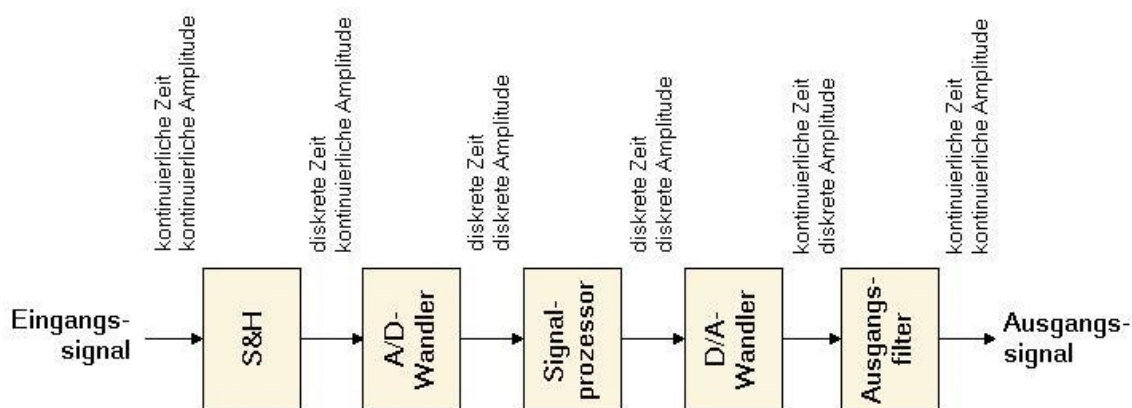
- [55] e2v technologies Data sheet for CCD42-80, Data Sheet Nr. 100025, page 1

BIBLIOGRAPHY

Appendix A

Glossary

ADU : Analog-Digital Unit [25]



For conversion of analogical information into digital information a so called Analog-Digital-Converter or Unit is needed. A CCD chip measures the amount of light by converting photons into electrons. The electrons are collected and counted by a so called ADC, the important property of the converter is the amount of data the unit can handle. As any digital information is stored in bits, the ADC is characterized by its capacity in bits. E.G. an 8 bit ADC can store up to 8 bit, that means $2^8 = 256$ different states, characterizing the amount of light that was collected during a certain time period. Our CoRoT data has pixel values between 0 and 12000 ADU, that means the used detector has a 16 bit ADC with a numerical maximum of 65536 so called ADU's. These numbers hold no direct information about the measured amount of light, they only describe the amount of light a pixel absorbed relatively to the other pixels in the image.

PSF: Point spread function

The **Point Spread Function** is a relatively new concept in optics, due to the availability of computers and ray tracing software. The name Point spread function was long time not used like this, in [4] the word "Point spread function" does not even appear. But nowadays it is relatively easy to compute the image of a point source on an optical plane, the so called Point spread function. It is only valid for one position and is also often only numerically solved. From a classical point of view, the light from a point source is diffracted into a so called Airy function, but an Airy function only describes the image of a point source along the optical axis, so side effects like chromatical or spherical aberrations have to be computed using Zernike polynomials, optical transfer functions and modulation transfer functions, etc., a lot of mathematics and very difficult to solve analytically. Today its not necessary to know all this theory in detail, it is enough to know the properties of the lenses and a good ray tracing program (like ZEMAX) computes the so called point spread function numerically for a given position in the field of view, for given color, etc. So today the point spread function became a very common tool to simulate optical systems and their properties.

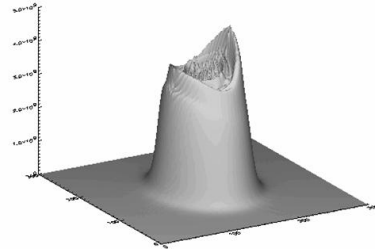
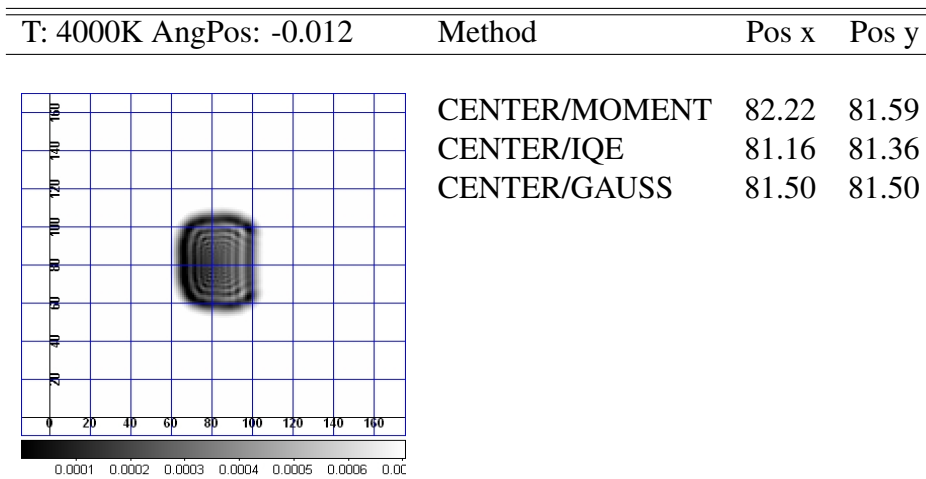


Figure A.1: CoRoT PSF [43]

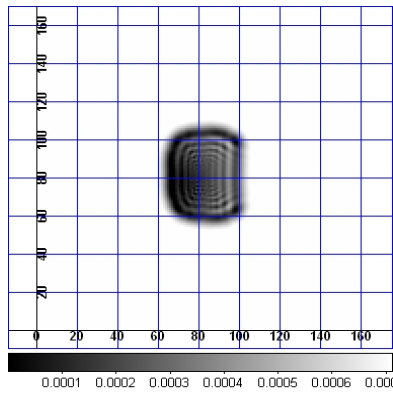
Appendix B

The PSFs and their derived positions

Finally we see here the 16 PSFs and their difference in shape for different temperatures and angular distances of simulated stars. Some variations in size and intensity can be seen directly in these images.

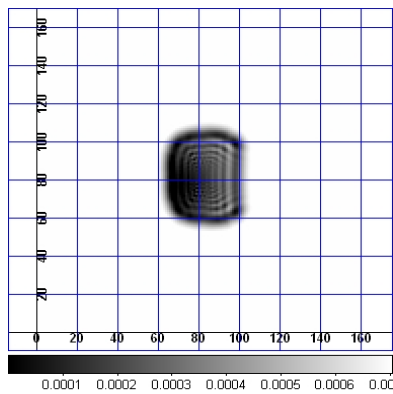


T: 5000K AngPos: -0.012



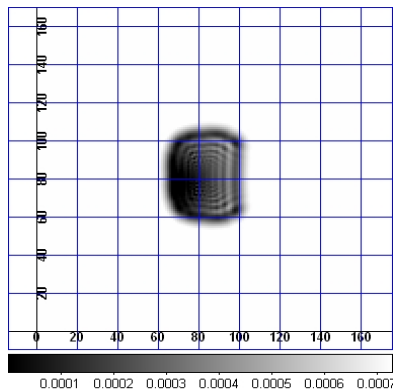
Method	Pos x	Pos y
CENTER/MOMENT	82.27	81.57
CENTER/IQE	81.08	81.25
CENTER/GAUSS	53.72	75.52

T: 6000K AngPos: -0.012



Method	Pos x	Pos y
CENTER/MOMENT	82.38	81.59
CENTER/IQE	81.08	81.19
CENTER/GAUSS	54.46	76.01

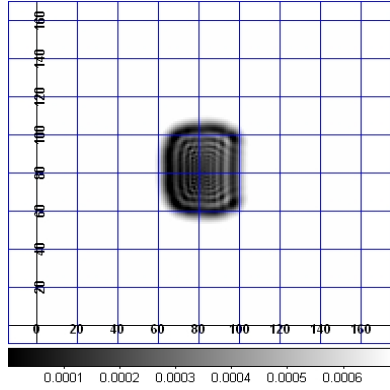
T: 8000K AngPos: -0.012



Method	Pos x	Pos y
CENTER/MOMENT	82.56	81.60
CENTER/IQE	81.10	81.12
CENTER/GAUSS	4.80	75.41

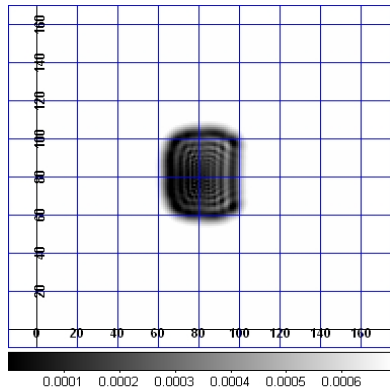
APPENDIX B. THE PSFS AND THEIR DERIVED POSITIONS

T: 4000K AngPos: 0.662



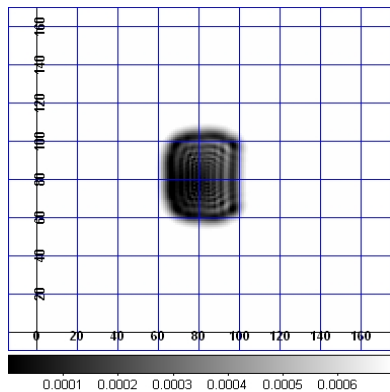
Method	Pos x	Pos y
CENTER/MOMENT	81.40	81.57
CENTER/IQE	80.68	81.32
CENTER/GAUSS	63.69	68.64

T: 5000K AngPos: 0.662



Method	Pos x	Pos y
CENTER/MOMENT	81.49	81.53
CENTER/IQE	80.74	81.20
CENTER/GAUSS	58.22	48.39

T: 6000K AngPos: 0.662



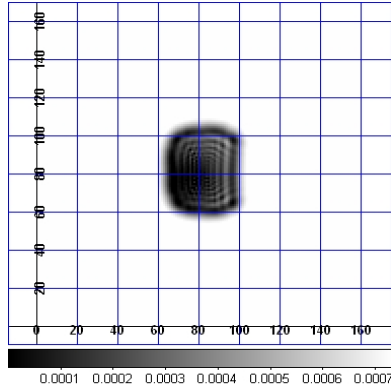
Method	Pos x	Pos y
CENTER/MOMENT	81.58	81.54
CENTER/IQE	80.80	81.13
CENTER/GAUSS	60.67	59.17

T: 8000K AngPos: 0.662

Method

Pos x

Pos y



CENTER/MOMENT

81.72

81.54

CENTER/IQE

80.89

81.04

CENTER/GAUSS

63.44

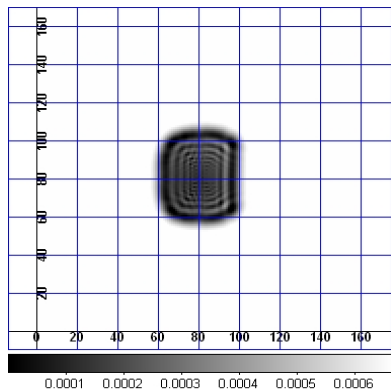
78.71

T: 4000K AngPos: 1.337

Method

Pos x

Pos y



CENTER/MOMENT

80.46

81.51

CENTER/IQE

80.15

81.33

CENTER/GAUSS

81.50

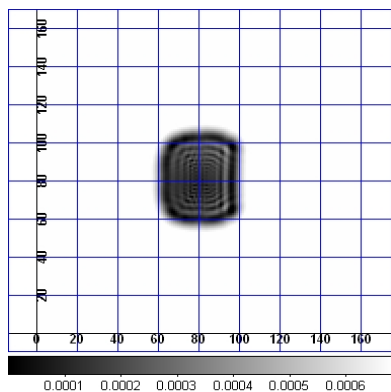
81.50

T: 5000K AngPos: 1.337

Method

Pos x

Pos y



CENTER/MOMENT

80.66

81.45

CENTER/IQE

80.40

81.19

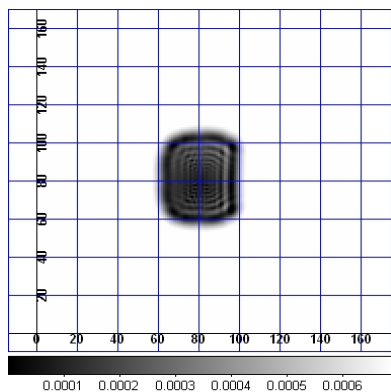
CENTER/GAUSS

79.31

80.65

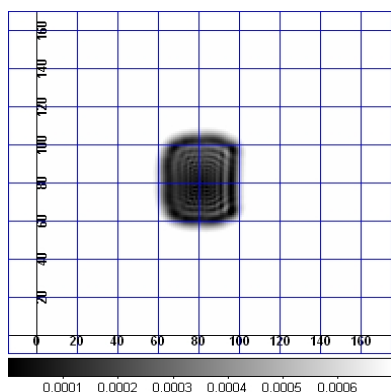
APPENDIX B. THE PSFS AND THEIR DERIVED POSITIONS

T: 6000K AngPos: 1.337



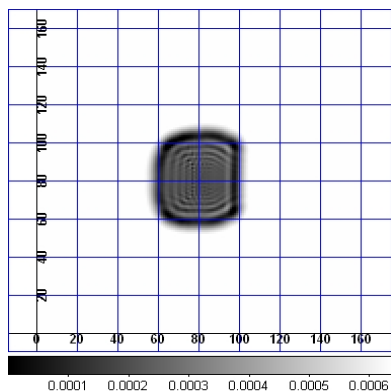
Method	Pos x	Pos y
CENTER/MOMENT	80.78	81.44
CENTER/IQE	80.56	81.11
CENTER/GAUSS	79.51	80.52

T: 8000K AngPos: 1.337



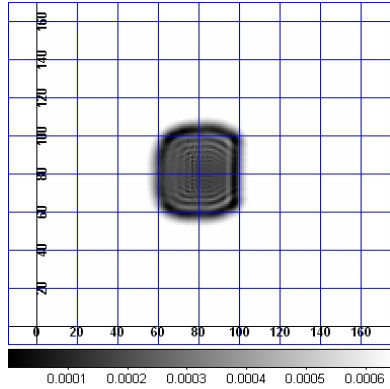
Method	Pos x	Pos y
CENTER/MOMENT	80.95	81.41
CENTER/IQE	80.78	81.00
CENTER/GAUSS	79.85	80.42

T: 4000K AngPos: 2.012



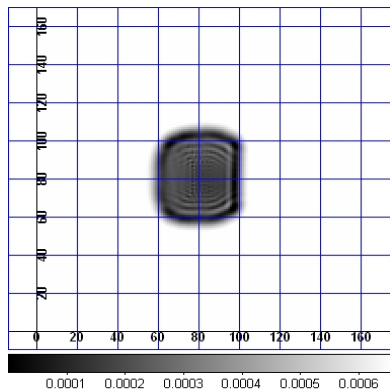
Method	Pos x	Pos y
CENTER/MOMENT	79.84	81.37
CENTER/IQE	79.65	81.38
CENTER/GAUSS	76.70	42.94

T: 5000K AngPos: 2.012



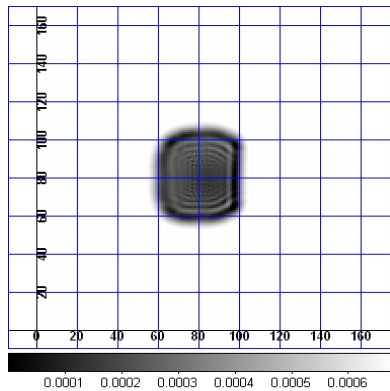
Method	Pos x	Pos y
CENTER/MOMENT	80.20	81.29
CENTER/IQE	80.16	81.15
CENTER/GAUSS	82.21	63.51

T: 6000K AngPos: 2.012



Method	Pos x	Pos y
CENTER/MOMENT	80.41	81.26
CENTER/IQE	80.48	81.12
CENTER/GAUSS	73.99	88.36

T: 8000K AngPos: 2.012



Method	Pos x	Pos y
CENTER/MOMENT	80.68	81.22
CENTER/IQE	80.87	80.93
CENTER/GAUSS	91.27	75.93

Appendix C

IDL Routines

C.0.2 Rebinning the images

Listing C.1: Rebin

```
pro rebin
nim = 16.
dim = 40.
output=dblarr(40,40)
name = strarr(1,nim)
openr,1,'orig/image.list'
readf,1,name
close,1
For k=0L,nim-1L do begin
    input='orig/'+name(0,k)
    print,name(0,k)
    z=readfits(input)
    FOR shiftx=0,4 do begin
        FOR shifty=0,4 do begin
            for i=0,dim-2 do begin
                for j=0,dim-2 do begin
                    print,i,j
                    output(i,j)=mean(z(i*4+shiftx:i*4+shiftx+3,j*4+shifty:
                        j*4+shifty+3))
                endfor
            endfor
            outfile='output/40_40_mean_x'+strtrim(shiftx,1)+'_y'+
                strtrim(shifty,1)+'_'+name(0,k)
            fits_write,output,outfile,output
        ENDFOR
    ENDFOR
ENDFOR
END
```

C.0.3 Spline shifting

Listing C.2: Spine shifting

```
pro spline
    ; creates images of 10 images shifted by 0,1
    Pixel

i=1
WHILE (i LT 10) DO BEGIN
    shift = (i)*0.1
    orig = mrdfits('data/4000_-0.420_
        -0.012.fits') ;EINLESEN
    splinex = findgen(160,160)
    xi=0
    WHILE (xi LT 160) DO BEGIN
        yi = FINDGEN(160)
        ADUi = orig(0:159, xi)
        t = FINDGEN
            (160)+shift
        trans = spline(yi,ADUi,t)
        splinex(*,xi)=trans
        xi=xi+1
        print,xi,'-te_Spalte_abgearbeitet',shift
    ENDWHILE
    c = string(
        shift,FORMAT='(f10.2)')
    outfilex = 'output/x+'+strtrim(c,1)+'_4000_-0.420_
        -0.012.fits'
    splinexy = findgen(160,160)
    xi=0
    WHILE (xi LT 160) DO BEGIN
        yi = FINDGEN(160)
        ADUi = splinex(xi,0:159)
        t = FINDGEN
            (160)+shift
        trans = spline(yi,ADUi,t)
        splinexy(xi,*)=trans
        xi=xi+1
        print,xi,'-te_Zeile_abgearbeitet'
    ENDWHILE
    outfilexy='output/xy+'+strtrim(c,1)+'_4000_-0.420_
        -0.012.fits'
    fits_write, outfilexy, splinexy
    i=i+1
ENDWHILE
```

END

C.0.4 Centroid algorithm

Listing C.3: Centroid

```

pro Centroid
;Thanks Torben!
    nim      = 25*16 ; 5 mal in X 5 mal in Y mal 4
               Temperaturen und 4 Positionen auf der CCD
    dim      = 40 ; Groesse des Bildes in pixel
    pos      =      fltarr(2,nim)
    name     = strarr(1,nim)
open,1,'data/Rebinned_40_40/image.list'
readf,1,name
close,1
for k=0L,nim-1L do begin
    input    = 'data/Rebinned_40_40/'+name(0,k)
    z        = readfits(input)
    xmin     = 0
               xmax           =      39
               ymin           =      0
               ymax           =      39
    deltax   =      xmax-xmin
    deltay   =      ymax-ymin
    b        =      z(xmin:
               xmax,ymin:ymax)
    cx       =      dblarr(deltax+1,
               deltay+1)
    cy       =      dblarr(deltax+1,
               deltay+1)
    for i=0, deltax do begin
for j=0,deltay do begin
        cx(i,j) =      float(i)*b(i,j)
        cy(i,j) =      float(j)*b(i,j)
    endfor
    endfor
    pos(0,k) =      (total(cx)/total(b))+
               xmin
    pos(1,k) =      (total(cy)/total(b))+
               ymin
endfor
openw,1,'output/float_RebinnedCentroidx.dat'
printf,1,pos(0,*)
close,1
openw,1,'output/float_RebinnedCentroidy.dat'

```

```

printf ,1 , pos (1 ,*)
close ,1
END

```

C.0.5 Symmetricity algorithm

Listing C.4: Symmetricity function

```

function Symmetricity , row , im1x , im1y , im2x , im2y

;Korrelationsmethode nach "Automatic Wavelength Measurement with
  a Digital Microdensitometer", R. Albrecht , H. Jenkner , W.W.
  Weiss , H.J. Wood, A&A 1977

x1=string(im1x,FORMAT='(i2)')
y1=string(im1y,FORMAT='(i2)')
x2=string(im2x,FORMAT='(i2)')
y2=string(im2y,FORMAT='(i2)')

tmp=mrdfits('data/Rebinned_40_40/40_40_mean_x'+strtrim(x1,1)+'
  _y'+strtrim(y1,1)+'_4000_-0.420_-0.012.fits')
tmp2=mrdfits('data/Rebinned_40_40/40_40_mean_x'+strtrim(x2,1)+'
  _y'+strtrim(y2,1)+'_4000_-0.420_-0.012.fits')

;
im=tmp(0:159,x)
im2=tmp2(0:159,x)

FOR TESTPUNKT=0,159,1 Do BEGIN
  IF (Testpunkt LE j) THEN BEGIN
    T=Testpunkt
    Summand=DBLARR(j)
    n=0
    WHILE (n LT j) DO BEGIN
      IF ((T+n) LT j AND (T-n) GE 0) THEN BEGIN
        Summand(n)=(im(T+n)-im(T-n))^2
      ENDIF
      IF ((T+n) GE j AND (T-n) GE 0) THEN BEGIN
        Summand(n)=(im(T-n))^2
      ENDIF
      IF ((T+n) LT j AND (T-n) LT 0) THEN BEGIN
        Summand(n)=(im(T+n))^2
      ENDIF
    
```

APPENDIX C. IDL ROUTINES

```
        IF ((T+n) GE j AND (T-n) LT 0) THEN BEGIN
            Summand(n)=0
        ENDIF
        Summe= TOTAL (Summand) ;Summe der quadrate aller
            Datenpunkte
        n=n+1
    ENDWHILE
    Result (Testpunkt)=Summe
ENDIF
IF (Testpunkt GT j) THEN BEGIN
    print , 'Fehler!_Testpunkt', Testpunkt , '_liegt_ausserhalb_'
    des_Wertebereichs: ', i
ENDIF
EndFOR
FOR TESTPUNKT=0,159,1 Do BEGIN
    IF (Testpunkt LE j) THEN BEGIN
        T=Testpunkt
        Summand=DBLARR(j)
        n=0
        WHILE (n LT j) DO BEGIN
            If ((T+n) LT j AND (T-n) GE 0) THEN BEGIN
                Summand(n)=(im2(T+n)-im2(T-n))^2
            ENDIF
            IF ((T+n) GE j AND (T-n) GE 0) THEN BEGIN
                Summand(n)=(im2(T-n))^2
            ENDIF
            IF ((T+n) LT j AND (T-n) LT 0) THEN BEGIN
                Summand(n)=(im2(T+n))^2
            ENDIF
            IF ((T+n) GE j AND (T-n) LT 0) THEN BEGIN
                Summand(n)=0
            ENDIF
            Summe= TOTAL (Summand) ;Summe der quadrate aller
                Datenpunkte
            n=n+1
        ENDWHILE
        Result2 (Testpunkt)=Summe
    ENDIF
    IF (Testpunkt GT j) THEN BEGIN
        print , 'Fehler!_Testpunkt', Testpunkt , '_liegt_ausserhalb_'
        des_Wertebereichs: ', i
    ENDIF
EndFOR
```

C.0.6 IDL Spline Routine

Listing C.5: IDL spline.pro routine

```
; $Id: //depot/idl/IDL_70/idldir/lib/spline.pro#1 $
;
; Copyright (c) 1983–2007, ITT Visual Information Solutions. All
; rights reserved. Unauthorized reproduction is prohibited
;
;+
; NAME:
;     SPLINE
;
; PURPOSE:
;     This function performs cubic spline interpolation.
;
; CATEGORY:
;     Interpolation – E1.
;
; CALLING SEQUENCE:
;     Result = SPLINE(X, Y, T [, Sigma])
;
; INPUTS:
;     X:     The abscissa vector. Values MUST be monotonically
;           increasing.
;
;     Y:     The vector of ordinate values corresponding to X
;           .
;
;     T:     The vector of abscissae values for which the
;           ordinate is
;           desired. The values of T MUST be monotonically
;           increasing.
;
; OPTIONAL INPUT PARAMETERS:
;     Sigma: The amount of "tension" that is applied to the
;           curve. The
;           default value is 1.0. If sigma is close to 0, (e
;           .g., .01),
;           then effectively there is a cubic spline fit. If
;           sigma
;           is large, (e.g., greater than 10), then the fit
;           will be like
;           a polynomial interpolation.
;
```

APPENDIX C. IDL ROUTINES

```
; KEYWORDS:
;   DOUBLE: Set this keyword to force computations to be done
;             using double-precision arithmetic.
;
; OUTPUTS:
;   SPLINE returns a vector of interpolated ordinates.
;   Result(i) = value of the function at T(i).
;
; RESTRICTIONS:
;   Abcissa values must be monotonically increasing.
;
; EXAMPLE:
;   The commands below show a typical use of SPLINE:
;
;           X = [2.,3.,4.]           ;X values of original
;   function
;           Y = (X-3)^2           ;Make a quadratic
;           T = FINDGEN(20)/10.+2   ;Values for interpolated
;   points.
;
;           3.9.                   ;twenty values from 2 to
;
;           Z = SPLINE(X,Y,T)       ;Do the interpolation.
;
; MODIFICATION HISTORY:
;   Author: Walter W. Jones, Naval Research Laboratory, Sept
;   26, 1976.
;   Reviewer: Sidney Prahl, Texas Instruments.
;   Adapted for IDL: DMS, March, 1983.
;   CT, RSI, July 2003: Added double precision support and
;   DOUBLE keyword,
;   use vector math to speed up the loops.
;   CT, RSI, August 2003: Must have at least 3 points.
;
;—
```

```
function spline ,x,y,t,sigmaIn , DOUBLE=double

  compile_opt idl2

  on_error ,2                               ;Return to caller if an
  error occurs

  n = N_ELEMENTS(x) < N_ELEMENTS(y)
  if (n le 2) then $
    MESSAGE, 'X_and_Y_must_be_arrays_of_3_or_more_elements.'
```

```

dbl = (N_ELEMENTS(double) gt 0) ? KEYWORD_SET(double) : $
      (SIZE(x,/TYPE) eq 5) || (SIZE(y,/TYPE) eq 5) || (SIZE(t
        ,/TYPE) eq 5)

sigma = (n_params(0) lt 4) ? (dbl ? 1d : 1.0) : $
        (sigmaIn > (dbl ? 0.001d : 0.001)) ;in range?

xx = dbl ? DOUBLE(x) : FLOAT(x)
yy = dbl ? DOUBLE(y) : FLOAT(y)
tt = dbl ? DOUBLE(t) : FLOAT(t)
yp = dbl ? DBLARR(2*n) : FLTARR(2*n) ;temp storage

delx1 = xx[1] - xx[0] ;1st incr
dx1 = (yy[1] - yy[0])/delx1

nm1 = n - 1L
np1 = n + 1L

delx2 = xx[2]-xx[1]
delx12 = xx[2]-xx[0]
c1 = -(delx12+delx1)/delx12/delx1
c2 = delx12/delx1/delx2
c3 = -delx1/delx12/delx2

slpp1 = c1*yy[0]+c2*yy[1]+c3*yy[2]
deln = xx[nm1]-xx[nm1-1]
delnm1 = xx[nm1-1]-xx[nm1-2]
delnn = xx[nm1]-xx[nm1-2]
c1 = (delnn+deln)/delnn/deln
c2 = -delnn/deln/delnm1
c3 = deln/delnn/delnm1
slppn = c3*yy[nm1-2]+c2*yy[nm1-1]+c1*yy[nm1]

sigmap = sigma*nm1/(xx[nm1]-xx[0])
dels = sigmap*delx1
exps = exp(dels)
sinhs = 0.5d*(exps - 1./exps)
sinhin = 1./(delx1*sinhs)
diag1 = sinhin*(dels*0.5d*(exps+1./exps)-sinhs)
diagin = 1./diag1
yp[0] = diagin*(dx1-slpp1)
spdiag = sinhin*(sinhs-dels)
yp[n] = diagin*spdiag

```

APPENDIX C. IDL ROUTINES

```

; Do as much work using vectors as possible.
delx2 = xx[1:*] - xx
dx2 = (yy[1:*] - yy)/delx2
dels = sigmap*delx2
exps = exp(dels)
sinhs = 0.5d *(exps - 1./exps)
sinhin = 1./(delx2*sinhs)
diag2 = sinhin*(dels*(0.5d*(exps+1./exps))-sinhs)
diag2 = [0, diag2 + diag2[1:]]
dx2nm1 = dx2[nm1-1] ; need to save this to calc yp[nm1]
dx2 = [0, dx2[1:*] - dx2]
spdiag = sinhin*(sinhs-dels)

; Need to do an iterative loop for this part.
for i=1L,nm1-1 do begin
    diagin = 1./(diag2[i] - spdiag[i-1]*yp[i+n-1])
    yp[i] = diagin*(dx2[i] - spdiag[i-1]*yp[i-1])
    yp[i+n] = diagin*spdiag[i]
endfor

diagin = 1./(diag1-spdiag[nm1-1]*yp[n+nm1-1])
yp[nm1] = diagin*(slppn-dx2nm1-spdiag[nm1-1]*yp[nm1-1])
for i=n-2,0,-1 do $
    yp[i] = yp[i] - yp[i+n]*yp[i+1]

m = n_elements(t)
subs = replicate(long(nm1),m) ;subscripts
s = xx[nm1]-xx[0]
sigmap = sigma*nm1/s

j = 0L
for i=1L,nm1 do begin ;find subscript where xx[subs] > t(j)
    > xx[subs-1]
    while tt[j] lt xx[i] do begin
        subs[j]=i
        j++
        if j eq m then goto ,done
    endwhile
endfor
done:
subs1 = subs - 1
del1 = tt-xx[subs1]
del2 = xx[subs] - tt
dels = xx[subs]-xx[subs1]

```

```

    exps1 = exp(sigmoid*del1)
    sinh1 = 0.5*(exps1 - 1./exps1)
    exps2 = exp(sigmoid*del2)
    sinh2 = 0.5*(exps2 - 1./exps2)
    exps = exps1*exps2
    sinhs = 0.5*(exps - 1./exps)
    spl = (yp[subs]*sinh1+yp[subs1]*sinh2)/sinhs + $
          ((yy[subs]-yp[subs])*del1+(yy[subs1]-yp[subs1])*del2)/
          dels
    return , (m eq 1) ? spl[0] : spl
end

```

C.0.7 IDL Shift Routine

Listing C.6: IDL shift.pro routine

```

; $Id: //depot/idl/IDL_70/idldir/lib/shift_diff.pro#1 $
;
; Copyright (c) 2006-2007, ITT Visual Information Solutions. All
; rights reserved. Unauthorized reproduction is prohibited
;
;
;+
; NAME:
;     SHIFT_DIFF
;
; PURPOSE:
;     Apply a shift-difference filter to an image array.
;
; CATEGORY:
;     Image Processing.
;
; CALLING SEQUENCE:
;     Result = SHIFT_DIFF(Array [, /ADD_BACK]
;                          [, DIRECTION=direction]
;                          [, <CONVOL keywords>])
;
; INPUTS:
;     Array = A 2D array of any basic type except string.
;     Keywords:
;     ADD_BACK: Set this keyword to cause the original input
array
;                to be added back to the difference array
generated
;                by the shift-difference operator. This is often
useful

```

APPENDIX C. IDL ROUTINES

```
;           for sharpening the image.
;   DIRECTION: An integer between 0 and 7 to specify the
;   direction of
;           the shift:
;           0 1 2
;           3 x 4
;           5 6 7
;   This procedure also accepts all keywords accepted by
;   CONVOL.
;   The scale_factor argument to CONVOL is left at it's
;   default
;   value to 1 since the kernel needs no scaling.
;
; OUTPUTS:
;   If the ADD_BACK keyword is set, the function returns an
;   array of
;   the same type and dimensions as the input array.
;
;   If the ADD_BACK keyword is not set, the function returns
;   an array
;   of the same dimensions as the input array, but with a type
;   described
;   as follows:
;           BYTE --> INT
;           INT  --> LONG
;           UINT --> LONG
;           ULONG --> LONG64
;           ULONG64 --> LONG64
;   For all other input types, the output type is the same
;   as the input type.
;
; SIDE EFFECTS:
;   None.
;
; RESTRICTIONS:
;   None.
;
; PROCEDURE:
;   Calls the CONVOL function with the appropriate kernel.
;   See the
;   CONVOL function for information on the treatment of edge
;   pixels.
;
; EXAMPLE CALLING SEQUENCE:
;   TVSCL, SHIFT_DIFF(DIST(200), DIRECTION=4)
```

```

;   should produce an image with black gradients on the left
;   and white gradients on the right.
;
; MODIFICATION HISTORY:
;   Sep 2006 – Initial Version
;–
function shift_diff , arr_in , ADD_BACK=add_back , $
                                DIRECTION=direction_in , $
                                _REF_EXTRA=_extra

    compile_opt idl2
    on_error , 2                ;Return to caller if an error
                                occurs
    CATCH, err                  ;Catch so we can identify
                                ourself
    if err ne 0 then begin
        CATCH, /CANCEL
        MESSAGE, !ERROR_STATE.MSG
    endif

;; Set up kernel , based on direction
if N_ELEMENTS(direction_in) gt 0 then begin
    direction = (direction_in > 0) < 7
    index = direction gt 3 ? direction + 1 :
            direction
endif else $
    index = 0
    kernel = INTARR(3,3)
    kernel[4] = 1
    kernel[index] = -1

;; Use a larger , signed integer type.
type = SIZE(arr_in , /TYPE)
case type of
    1: calctype = 2    ; byte --> int
    2: calctype = 3    ; int --> long
    12: calctype = 3   ; uint --> long
    13: calctype = 14  ; ulong --> long64
    15: calctype = 14  ; ulong64 --> long64
    else: calctype = type
endcase
    arr = FIX(arr_in , TYPE=calctype)

    result = CONVOL(TEMPORARY(arr) , kernel , _STRICT_EXTRA=
                    _extra)

```

```

;; Add back original data if requested
if KEYWORD_SET(add_back) then begin

    result += arr_in

    ;; Clip and convert back to original type.
case type of
    1: clip = [0, 255]
    2: clip = [-32768, 32767]
    12: clip = [0, 65535u]
    13: clip = [0, 4294967295UL]
    15: clip = [0, 9223372036854775807ULL]
    else: ; no clipping needed
endcase

if (N_ELEMENTS(clip) gt 0) then begin
    mn = MIN(result, MAX=mx)
    if (mn lt clip[0]) then result >= clip[0]
    if (mx gt clip[1]) then result <= clip[1]
endif

    result = FIX(result, TYPE=type)
endif

return, result
end

```

C.0.8 How to count the number of stars

Listing C.7: How to count the number of stars

```

starnumber=0
found=0
For i=0,2158 do begin
    for j=0,2052 do begin
        if z(i,j) LT cut AND z(i+1,j+1) GE cut then begin
            starnumber = starnumber +1
            print, 'z(i,j):_ ',z(i,j), '_z(i+1,j+1):_ ',z(i+1,j
                +1)
            print, 'starnumber:_ ',starnumber+1, '_i:_ ',i, '_j:_ '
                ',j
        endif
    endfor
endfor

```

C.0.9 How to find the edge

Listing C.8: How to find the edge

```
                                ; linke
                                kante
                                erreicht

    if z(i,j) LT cut AND z(i,j+1) GE cut then begin
        print , 'linke_Kante:_', L_K_i, L_K_j
    endif
        ; rechte kante erreicht
    if z(i,j) GE cut AND z(i,j+1) LT cut then begin
        print , 'rechte_Kante:_', R_K_i, R_K_j
    endif
```

C.0.10 Creating a list of stars

Listing C.9: Creating a list of stars

```
pro starlist
folder=' ../ '
filename=' AN0_FULLIMAGE_20071016T184715_A1 '
    fieldsize_x=2160
    fieldsize_y=2054
    cut=2000
    cut_s=string(cut,FORMAT='(f10.2)')
    input=folder+filename+'.fits'
    z1=readfits(input)

;##### count number of stars #####
starnumber=0
found=0
    For i=0,fieldsize_x-2 do begin
        for j=0,fieldsize_y-2 do begin
            if z(i,j) LT cut AND z(i+1,j+1) GE cut then begin
                starnumber = starnumber +1
            endif
        endfor
    endfor
endfor

;##### write list of stars #####
Star_List=dblarr(4,starnumber+1)
starnumber=0
found=0
```

```

For i=0,fieldsize_x-2 do begin
  for j=0,fieldsize_y-2 do begin
    if z(i,j) LT cut AND z(i+1,j+1) GE cut then begin
      starnumber = starnumber +1
      Star_List(0,starnumber)=starnumber
      Star_List(1,starnumber)=i
      Star_List(2,starnumber)=j
      Star_List(3,starnumber)=z(i+1,j+1)
    endif
  endfor
endfor
openw,1,folder+'starlist_'+filename+'_cut_'+strtrim(cut_s,1)+'
  dat'
printf,1,Star_List
CLOSE,1
END

```

C.0.11 Creating a list of filtered stars

Listing C.10: Creating a filtered list of stars

```

pro starlist_filter
  folder='..\ '
  openr,1,folder+'\
    starlist2_AN0_FULLIMAGE_20071016T173859_A1_cut_2000.00.dat'
  id=0d
  x_pos=0d
  y_pos=0d
  intensity=0d
  breite_x=0d

  starnumber=825
  Star_List=dblarr(5,starnumber+1)

  i=0

WHILE NOT EOF(1) DO BEGIN
  readf,1,id,x_pos,y_pos,intensity,format='(6x,F8.4,7x,F8.4,7x,
    F8.4,7x,F8.4) '
  Star_List(0,i)=id
  Star_List(1,i)=x_pos
  Star_List(2,i)=y_pos
  Star_List(3,i)=intensity
  Star_List(4,i)=breite_x
  i=i+1
ENDWHILE

```

```

CLOSE,1
a=Star_List

i=0
for i=0,starnumber-1 do begin
  while a(1,i) EQ a(1,i+1) and a(1,i+1) NE 0 do begin
    print , 'test',i, a(1,i), a(1,i+1)
    a(4,i)=a(4,i)+1 ; breite von a wird erhöht um 1
    a(*,i+1)=a(*,i) ; inhalt von a(i) wird nach a(i+1) kopiert
    a(*,i)=0 ; inhalt von a(i) wird auf 0 gesetzt
    i=i+1 ; i wird um eins erhöht
  endwhile
endfor

j=0
for i=0,starnumber do begin
  if a(1,i) ne 0 then begin ; Werte ungleich Null nach oben
    kopieren
    a(*,j)=a(*,i)
    j=j+1; Nullstellen zählen
  endif
endfor

a2=dblarr(5,j)
a2(*,*)=a(*,0:j-1)

starnumber_neu=j
j=0
for i=0,starnumber_neu-1 do begin
  if a2(4,i) GE 6 then begin ; Werte ungleich Null nach oben
    kopieren
    a(*,j)=a(*,i)
    j=j+1; Nullstellen zählen
  endif
endfor
a3=dblarr(5,j)
a3(*,*)=a(*,0:j-1)

openw,1, folder+' \starlist_filtered.dat'
printf,1, a3
CLOSE,1

end

```

C.0.12 Creating list of stars with their positions

Listing C.11: Creating list of stars with their positions

```
pro centroid_list

folder='..\ '
;filename = 'AN0_FULLIMAGE_20071016T173859_A1_cut_2000.00 '
filename='AN0_FULLIMAGE_20071016T184715_A1_cut_2000.00 '
input=folder+filename+'. fits '
z=readfits(input)
  openr,1, folder+'starlist_filtered.dat'
x_pos=0d
y_pos=0d
intensity=0d
id=0d
breite_x=0d
breite_y=0d
starnumber=29
Star_List=dblarr(7, starnumber)

i=0
WHILE NOT EOF(1) DO BEGIN
  readf,1,id,x_pos,y_pos,intensity,breite_x,format='(6x,F8
    .4,7x,F8.4,7x,F8.4,7x,F8.4,7x,F8.4)'
  Star_List(0,i)=id
  Star_List(1,i)=x_pos
  Star_List(2,i)=y_pos
  Star_List(3,i)=intensity
  Star_List(4,i)=breite_x
  Star_List(5,i)=0
  Star_List(6,i)=0
  i=i+1
ENDWHILE
CLOSE,1

for k=0,starnumber-1 do begin
  xmin=Star_List(1,k)
  xmax=xmin+17
  ymin=Star_List(2,k)
  ymax=ymin+18

  deltax=xmax-xmin
  deltay=ymax-ymin
  b=z(xmin:xmax,ymin:ymax)
```

```

cx=fltarr (deltax+1,deltay+1)
cy=fltarr (deltax+1,deltay+1)
For i=0, deltax do begin
    for j=0,deltay do begin
        cx(i,j)=(float(i)+1)*b(i,j)
        cy(i,j)=(float(j)+1)*b(i,j)
    endfor
endfor
center_x=(total(cx)/total(b))+xmin-1
center_y=(total(cy)/total(b))+ymin-1
Star_List(5,k)=center_x
Star_List(6,k)=center_y
endfor

;openw,1,folder+'starlist_centroid_LRa01_T1738_2.dat'
openw,1,folder+'_'+filename+'_centroid.dat'
for i=0,starnumber-1 do begin
    printf,1,i,Star_List(1,i),Star_List(2,i),Star_List(5,i),
        Star_List(6,i)
endfor
CLOSE,1

end

```

C.0.13 Determining the shift between two images

Listing C.12: Determining the shift between two images

```

pro shiftfind
folder='..\ '
filename='AN0_FULLIMAGE_20071016T173859_A1_cut_2000.00'
filename1='AN0_FULLIMAGE_20071016T184715_A1_cut_2000.00'
input1=folder+filename+'.fits'
input2=folder+filename1+'.fits'
a=readfits(input1)
b=readfits(input2)
fieldsize_x=2160
fieldsize_y=2054
limit_x=1080 ; noise ~1030 adu right from cut, left ~1130
a1=a(0:limit_x-1,*)
b1=b(0:limit_x-1,*)

;Step 1: Find Best Pixel
;*****
sum4=dblarr(10,10)
for i=0,9 do for j=0,9 do begin

```

```

k=i-5
l=j-5
a3=shift(b1,k,l)
z3=a3-a1
sum4(i,j)=total(abs(z3))
endfor
d=min(sum4,pos)
ind=array_indices(sum4,pos)
print, 'best_shift_in_px:', ind[0]-5, ind[1]-5, 'at_a_minimum_value
of:', d

```

C.0.14 Subpixel shifts, "Overlay" method, parts 2 to 5

Listing C.13: Determining the shift between two images, part2

```

; Step 2: spline to scan subpixel interval I
orig=a1(760:790,1730:1760)
shift_orig2=shift(b1(760:790,1730:1760), ind[0]-3, ind[1]-3)
diff=dblarr(29,29)
for i=1,29 DO for j=1,29 do BEGIN ; i walking index of the shift
    step, 29 is the upper limit
    shift_x=(i)*0.1 ; step size: 0.1
    shift_y=(j)*0.1
    spline_im = findgen(30,30) ; shifted image, image size: 30,30
    for xi=0,29 DO BEGIN ; spline(shift) generating of X
        yi=FINDGEN(30)
        ADUi=shift_orig2(xi,0:29)
        t=FINDGEN(30)+shift_x
        trans=spline(yi,ADUi,t)
        spline_im(xi,*)=trans
    endfor
    spline_im2 = findgen(30,30)
    for xi=0,29 DO BEGIN ; spline(shift) generating of Y
        yi=FINDGEN(30)
        ADUi=spline_im(0:29,xi)
        t=FINDGEN(30)+shift_y
        trans=spline(yi,ADUi,t)
        spline_im2(*,xi)=trans
    endfor
    dif=orig(0:29,0:29)-spline_im2(0:29,0:29)
    diff(i-1,j-1)=total(abs(dif))
endfor
d=min(diff,pos1)
ind1=array_indices(diff,pos1)
print, 'best_shift_in_subpixels_of_step_size_0.1:', ind1[0], ind1
[1], 'at_a_minimum_value_of:', d

```


Listing C.14: Determining the shift between two images, part3 of 5

```
*****
; Step 3: spline to scan subpixel interval II
diff=dblarr(59,59)
for i=1,59 DO for j=1,59 do BEGIN
  shift_x=(i)*0.05 ; step size: 0.05
  shift_y=(j)*0.05
  spline_im = findgen(30,30)
  for xi=0,29 DO BEGIN
    yi=FINDGEN(30)
    ADUi=shift_orig2(xi,0:29)
    t=FINDGEN(30)+shift_x
    trans=spline(yi,ADUi,t)
    spline_im(xi,*)=trans
  endfor
  spline_im2 = findgen(30,30)
  for xi=0,29 DO BEGIN
    yi=FINDGEN(30)
    ADUi=spline_im(0:29,xi)
    t=FINDGEN(30)+shift_y
    trans=spline(yi,ADUi,t)
    spline_im2(*,xi)=trans
  endfor
  dif=orig(0:29,0:29)-spline_im2(0:29,0:29)
  diff(i-1,j-1)=total(abs(dif))
endfor
bla1=min(diff, pos1)
ind1=array_indices(diff, pos1)
print, 'best_shift_in_subpixels_of_step_size_0.05:', ind1[0], ind1
      [1]
print, '_____at_a_minimum_value_of:_', bla1

*****
; Step 4 spline to scan subpixel interval III
diff=dblarr(119,119)
for i=1,119 DO for j=1,119 do BEGIN
  shift_x=(i)*0.025 ; step size: 0.025
  shift_y=(j)*0.025
  spline_im = findgen(30,30)
  for xi=0,29 DO BEGIN
    yi=FINDGEN(30)
    ADUi=shift_orig2(xi,0:29)
    t=FINDGEN(30)+shift_x
    trans=spline(yi,ADUi,t)
```

APPENDIX C. IDL ROUTINES

```

    spline_im(xi,*)=trans
endfor
spline_im2 = findgen(30,30)
for xi=0,29 DO BEGIN
    yi=FINDGEN(30)
    ADUi=spline_im(0:29,xi)
    t=FINDGEN(30)+shift_y
    trans=spline(yi,ADUi,t)
    spline_im2(*,xi)=trans
endfor
dif=orig(0:29,0:29)-spline_im2(0:29,0:29)
diff(i-1,j-1)=total(abs(dif))
endfor
bla1=min(diff, pos1)
ind1=array_indices(diff, pos1)
print, 'best_shift_in_subpixels_of_step_size_0.025:', ind1[0], ind1
    [1]
print, '_____at_a_minimum_value_of: ', bla1

;*****
;Step 5 spline to scan subpixel interval IIII
diff=dblarr(299,299)
for i=1,299 DO for j=1,299 do BEGIN
    shift_x=(i)*0.01 ; step size: 0.01
    shift_y=(j)*0.01
    spline_im = findgen(30,30)
    for xi=0,29 DO BEGIN
        yi=FINDGEN(30)
        ADUi=shift_orig2(xi,0:29)
        t=FINDGEN(30)+shift_x
        trans=spline(yi,ADUi,t)
        spline_im(xi,*)=trans
    endfor
    spline_im2 = findgen(30,30)
    for xi=0,29 DO BEGIN
        yi=FINDGEN(30)
        ADUi=spline_im(0:29,xi)
        t=FINDGEN(30)+shift_y
        trans=spline(yi,ADUi,t)
        spline_im2(*,xi)=trans
    endfor
    dif=orig(0:29,0:29)-spline_im2(0:29,0:29)
    diff(i-1,j-1)=total(abs(dif))
endfor
bla1=min(diff, pos1)

```

```

ind1=array_indices(diff, pos1)
print, 'best_shift_in_subpixels_of_step_size_0.01:', ind1[0], ind1
      [1]
print, '_____at_a_minimum_value_of:_', bla1
end

```

C.0.15 Turner method

Listing C.15: Turner Method

```

pro turner
; Measured star :

xA=804.10185D
yA=1821.6459D
; literature values :
alphaA= 102.0002812D
deltaA=0.3220531D

; reference stars :
x_1=679.16707D
y_1=1883.834D
alpha_1=101.9624567D
delta_1=0.2420525D

x_2=774.18959D
y_2=1749.0629D
alpha_2=102.0479296D
delta_2=0.3049731D

x_3=834.08549D
y_3=1900.9467D
alpha_3=101.9481471D
delta_3=0.3401889D

alpha1=double(alpha_1*!DPI/360)
delta1=double(delta_1*!DPI/360)
alpha2=double(alpha_2*!DPI/360)
delta2=double(delta_2*!DPI/360)
alpha3=double(alpha_3*!DPI/360)
delta3=double(delta_3*!DPI/360)
Alpha=double(alphaA*!DPI/360)
Delta=double(deltaA*!DPI/360)

```

APPENDIX C. IDL ROUTINES

```
q1=double(atan(double(tan(delta1))*double(cos(alpha1-Alpha))))
q2=double(atan(double(tan(delta2))*double(cos(alpha2-Alpha))))
q3=double(atan(double(tan(delta3))*double(cos(alpha3-Alpha))))

eta1=double(tan(Delta-q1))
eta2=double(tan(Delta-q2))
eta3=double(tan(Delta-q3))

xi1=(double(tan(alpha1-Alpha))*double(sin(q1)))/double(cos(Delta-q1))
xi2=(double(tan(alpha2-Alpha))*double(sin(q2)))/double(cos(Delta-q2))
xi3=(double(tan(alpha3-Alpha))*double(sin(q3)))/double(cos(Delta-q3))

x1=x_1-xA
y1=y_1-yA
x2=x_2-xA
y2=y_2-yA
x3=x_3-xA
y3=y_3-yA

a=((xi1-xi2)*(y2-y3)-(xi2-xi3)*(y1-y2))/((x1-x2)*(y2-y3)-(x2-x3)*(y1-y2))
b=((xi1-xi2)-a*(x1-x2))/(y1-y2)
c=xi1-a*x1-b*y1
d=((eta1-eta2)*(y2-y3)-(eta2-eta3)*(y1-y2))/((x1-x2)*(y2-y3)-(x2-x3)*(y1-y2))
e=((eta1-eta2)-d*(x1-x2))/(y1-y2)
f=eta1-d*x1-e*y1

A1=1/15*c*1/double(cos(Delta))

delta1=f

alphaobj=double(Alpha+A1)
deltaobj=double(Delta+delta1)
print , q1 , q2 , q3
print , xi1 , xi2 , xi3
print , y1 , y2 , y3
print , a , b , c , d , e , f
print , alphaobj*double(360/! dpi)
print , deltaobj*double(360/! dpi)
```

end

Appendix D

Centroid results for all rebinned PSFs

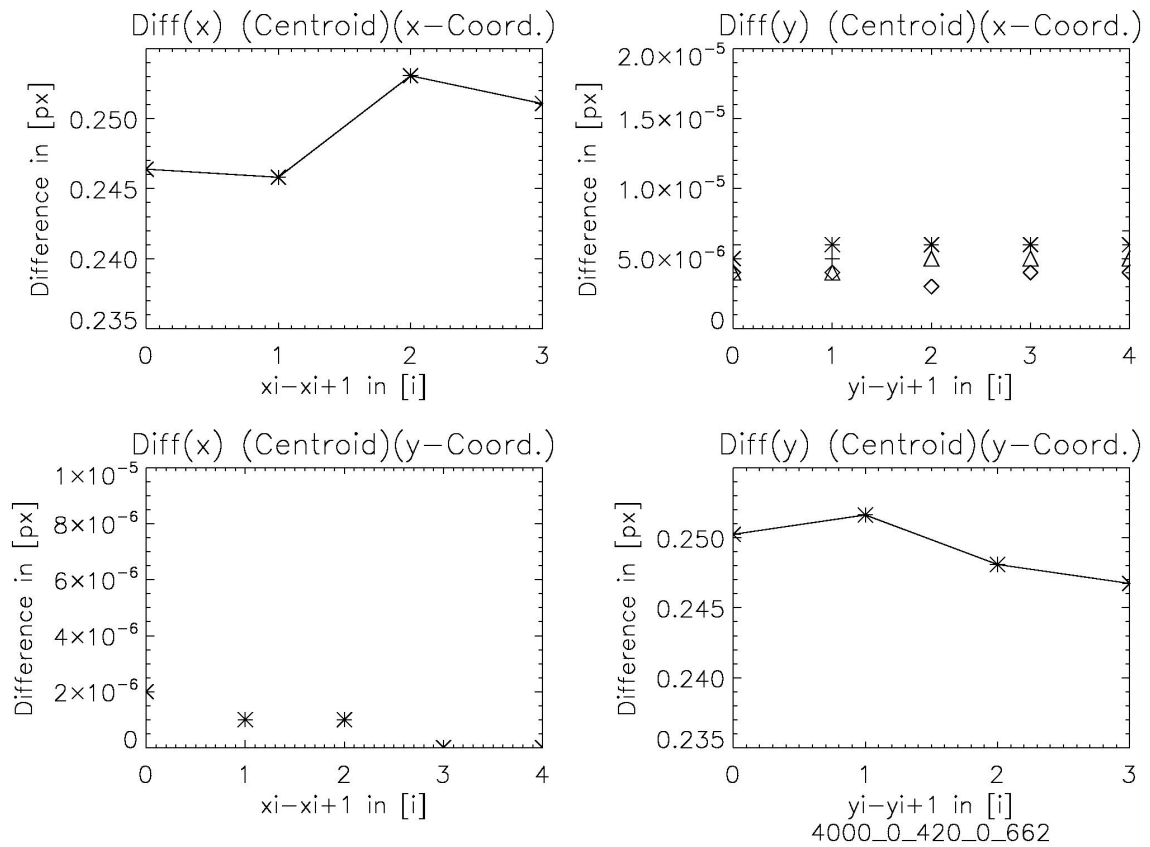


Figure D.1: 4000 K 0.662

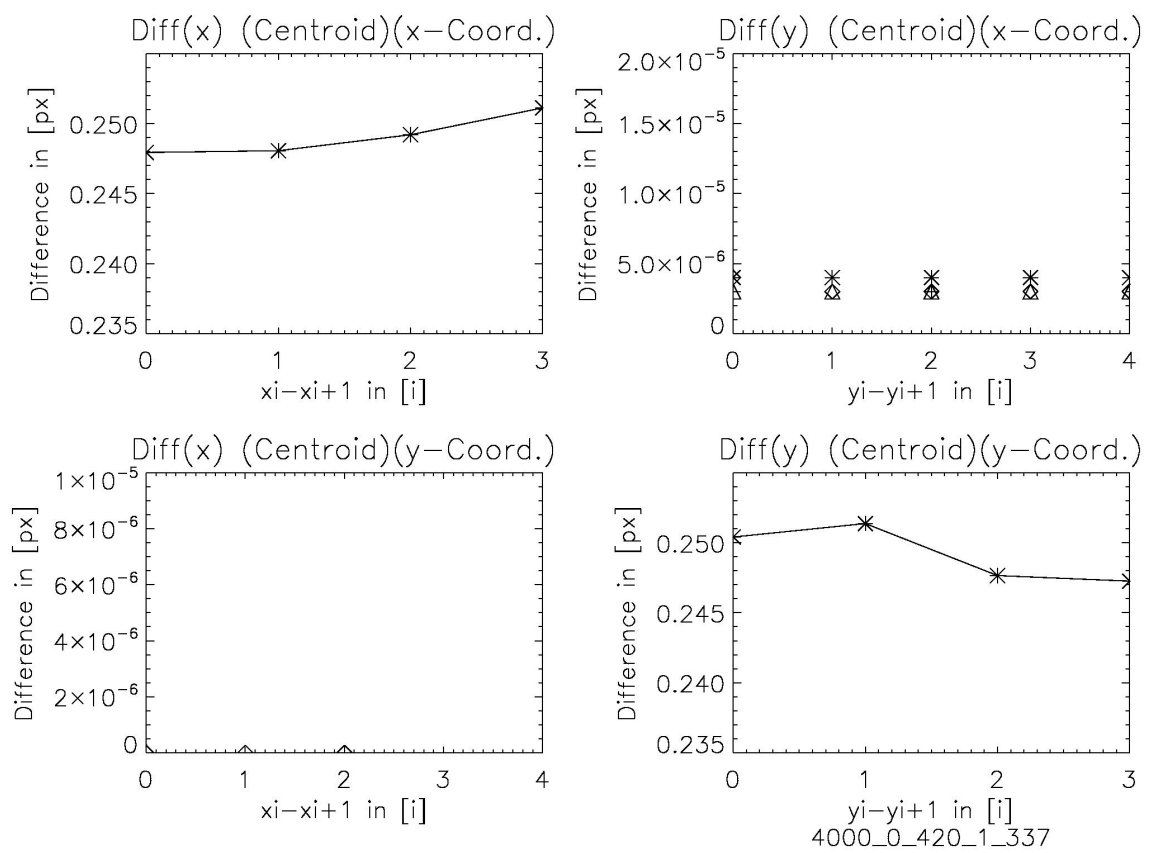


Figure D.2: 4000 K 1.337

APPENDIX D. CENTROID RESULTS FOR ALL REBINNED PSFS

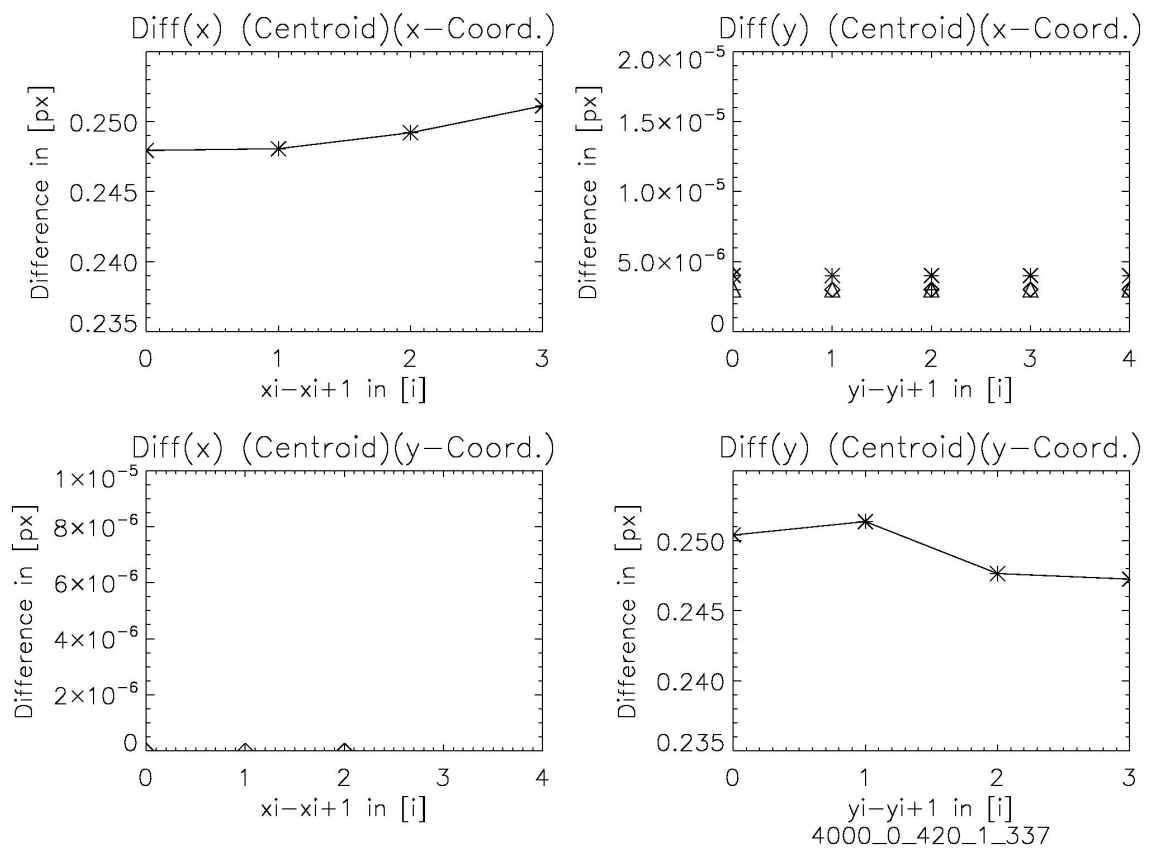


Figure D.3: 4000 K 1.337

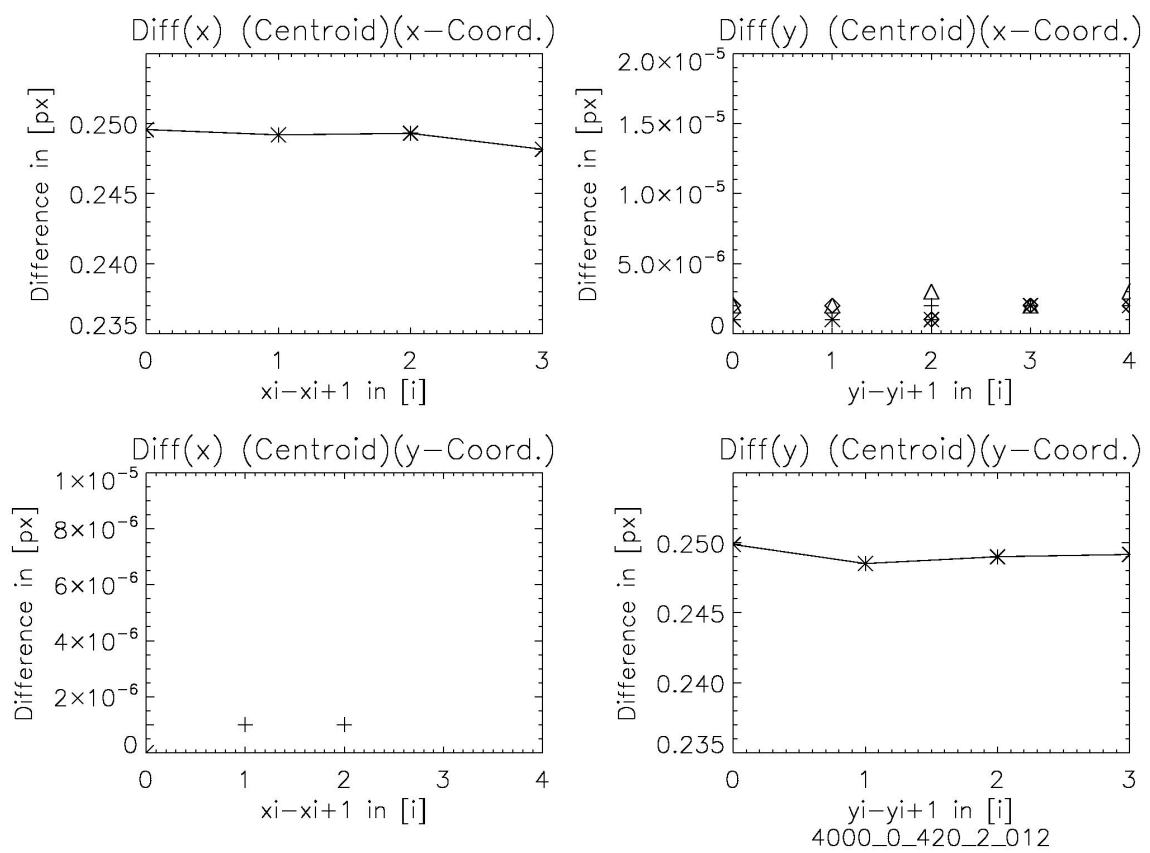


Figure D.4: 4000 K 2.012

APPENDIX D. CENTROID RESULTS FOR ALL REBINNED PSFS

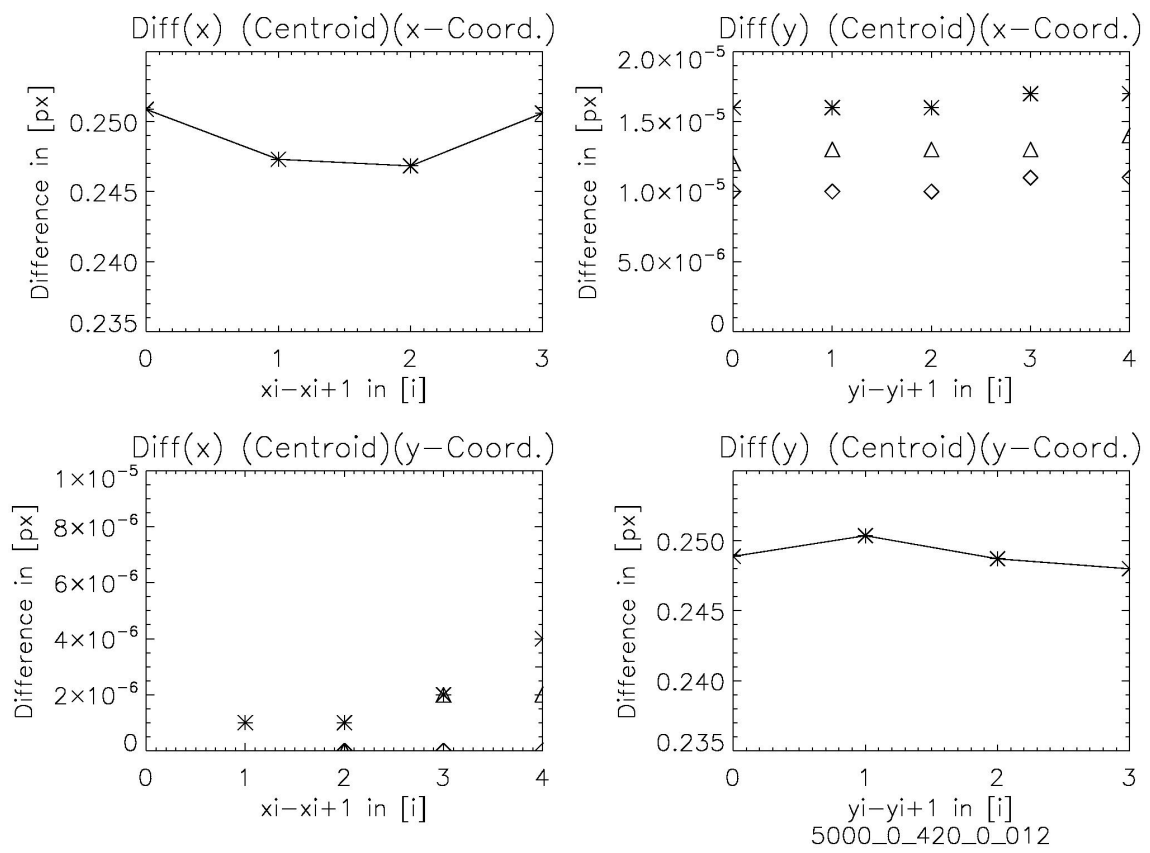


Figure D.5: 5000 K 0.012

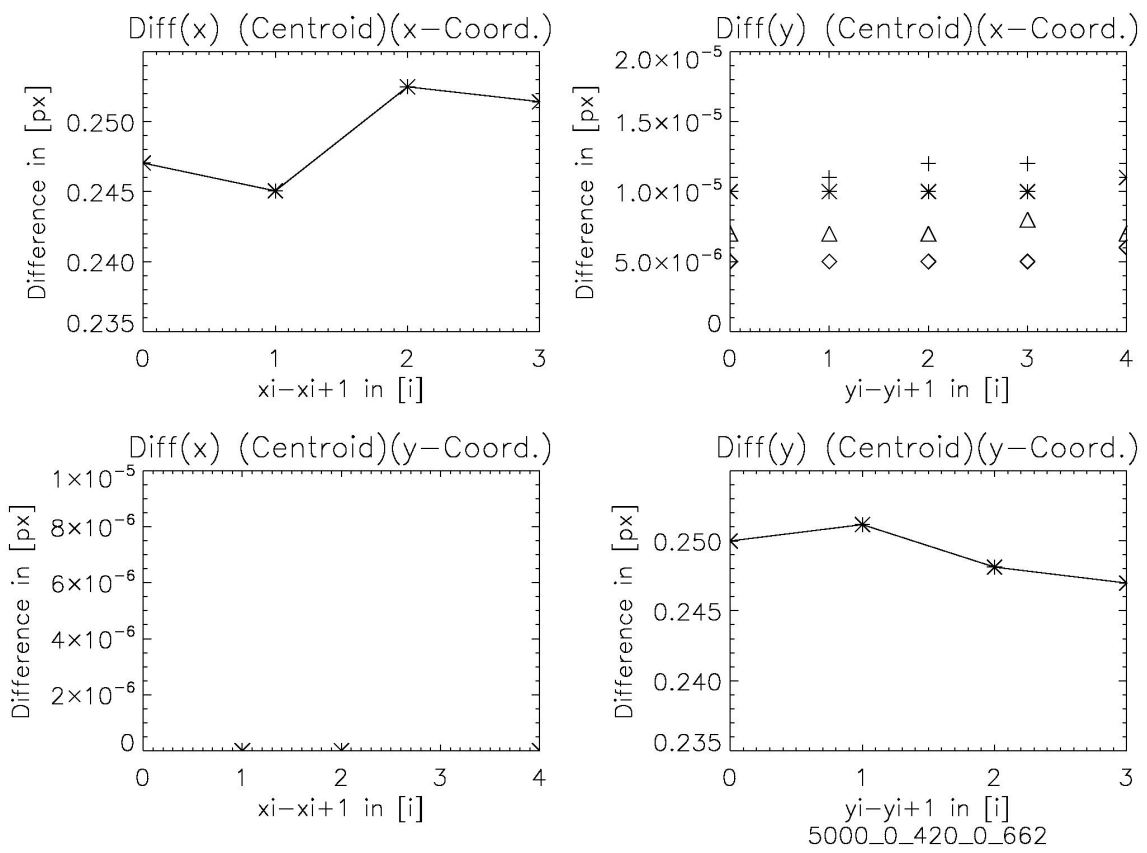


Figure D.6: 5000 K 0.662

APPENDIX D. CENTROID RESULTS FOR ALL REBINNED PSFS

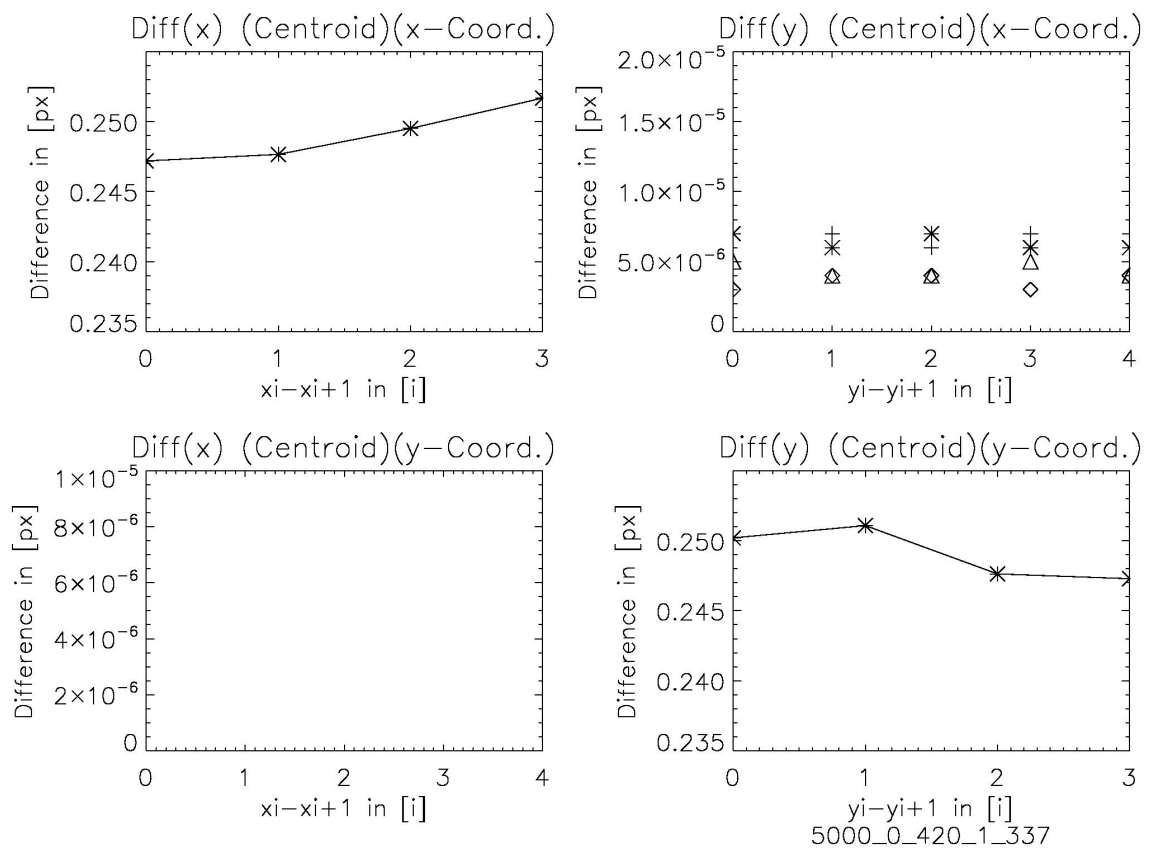


Figure D.7: 5000 K 1.337

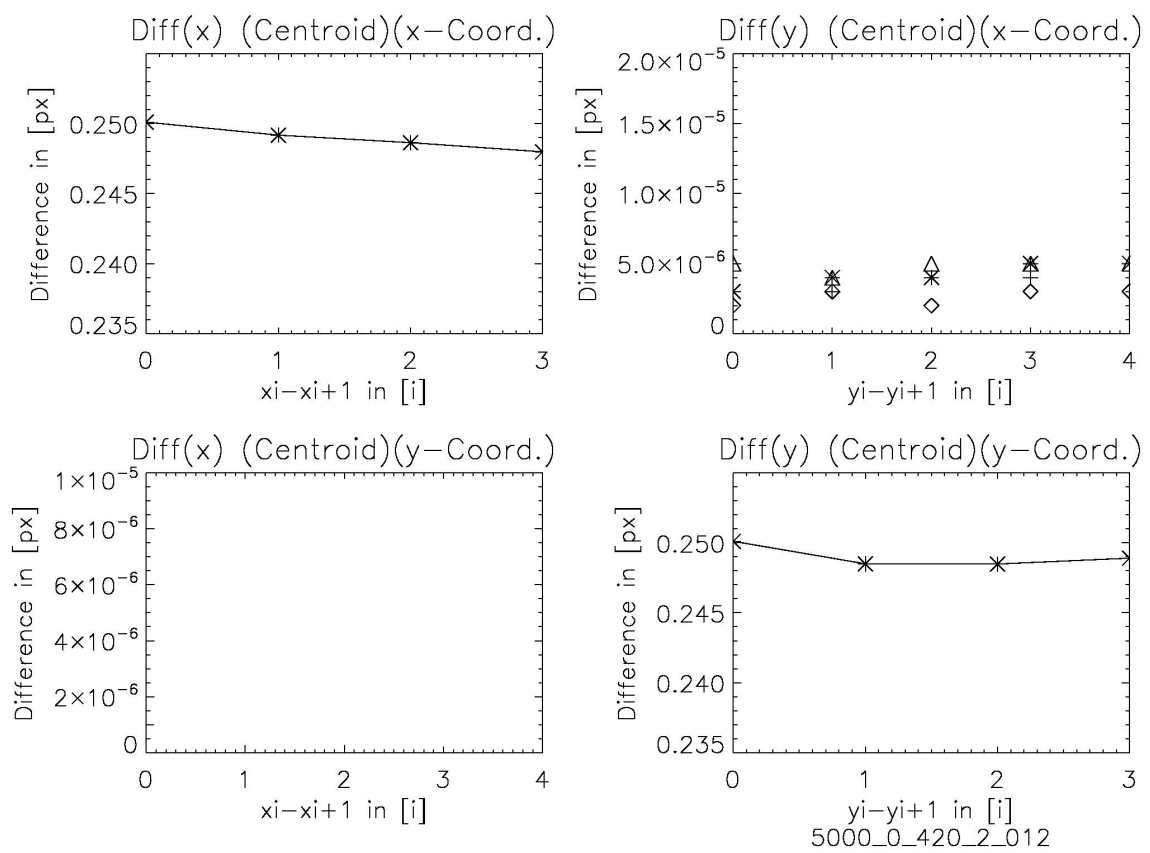


Figure D.8: 5000 K 2.012

APPENDIX D. CENTROID RESULTS FOR ALL REBINNED PSFS

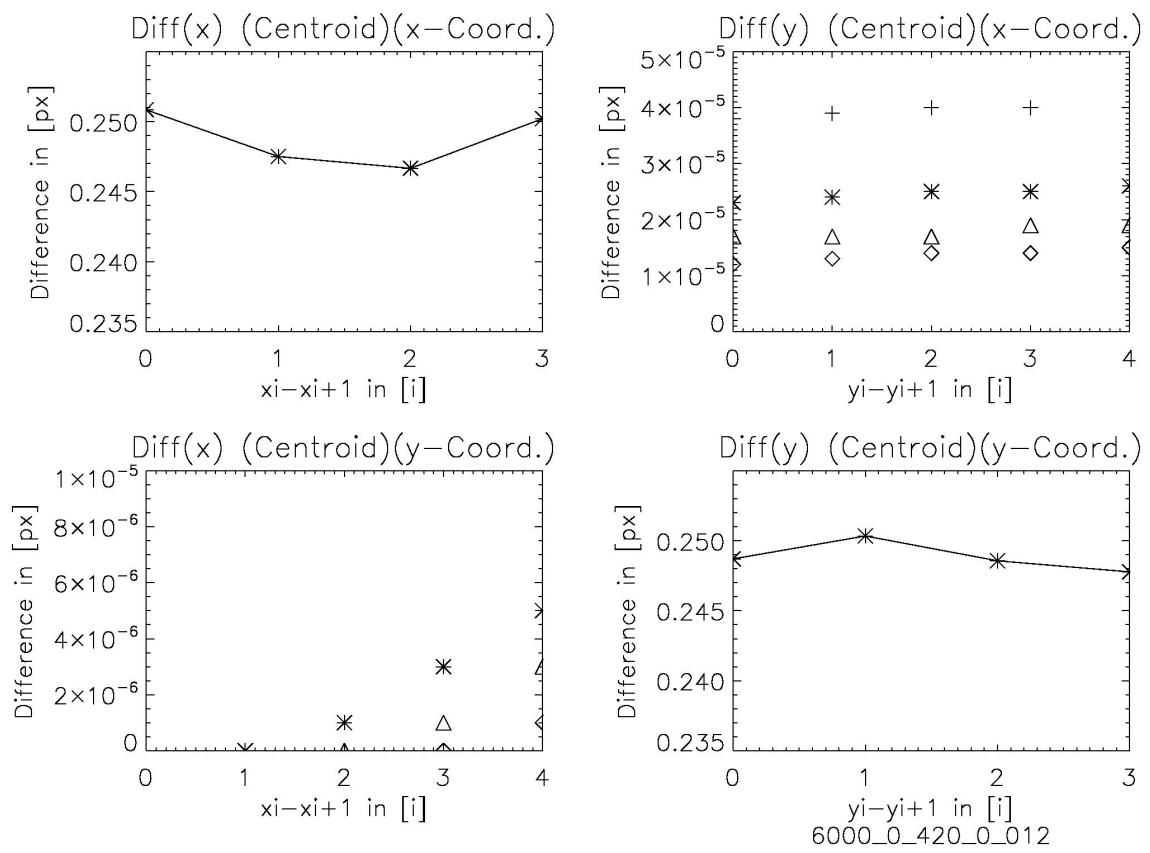


Figure D.9: 6000 K 0.012

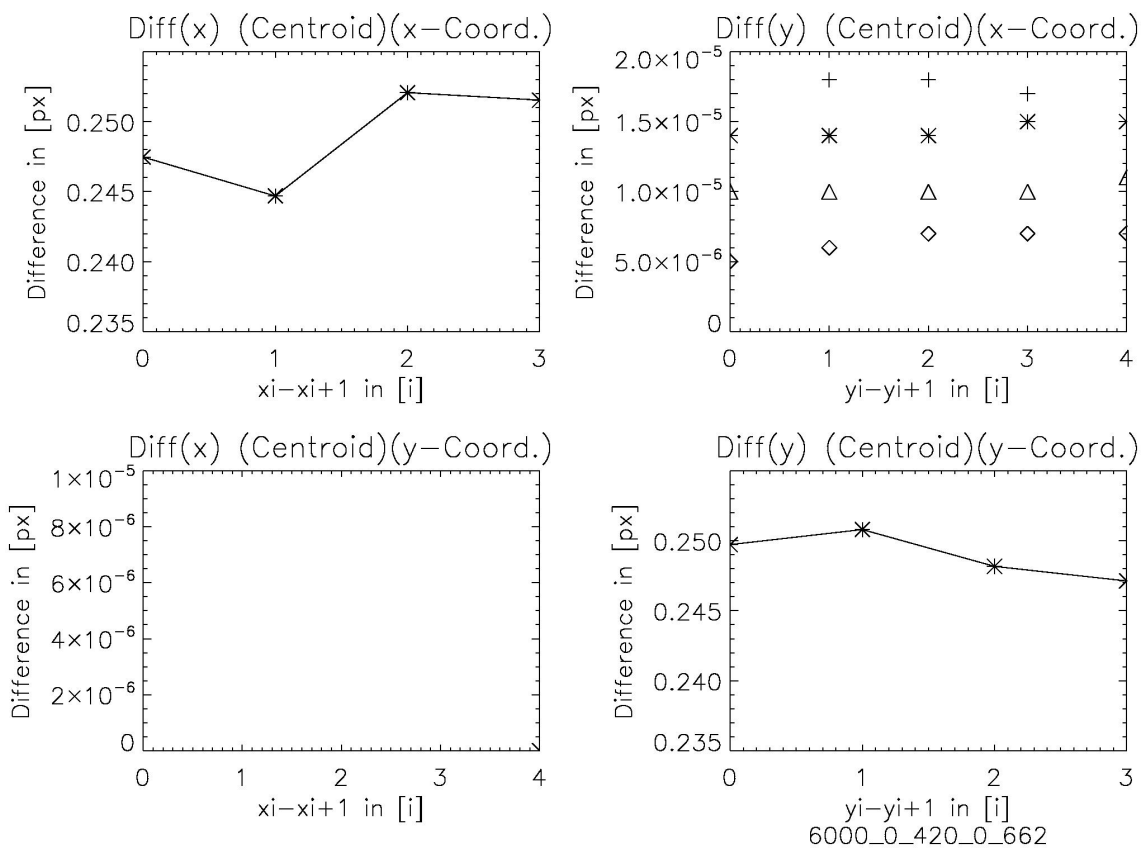


Figure D.10: 6000 K 0.662

APPENDIX D. CENTROID RESULTS FOR ALL REBINNED PSFS

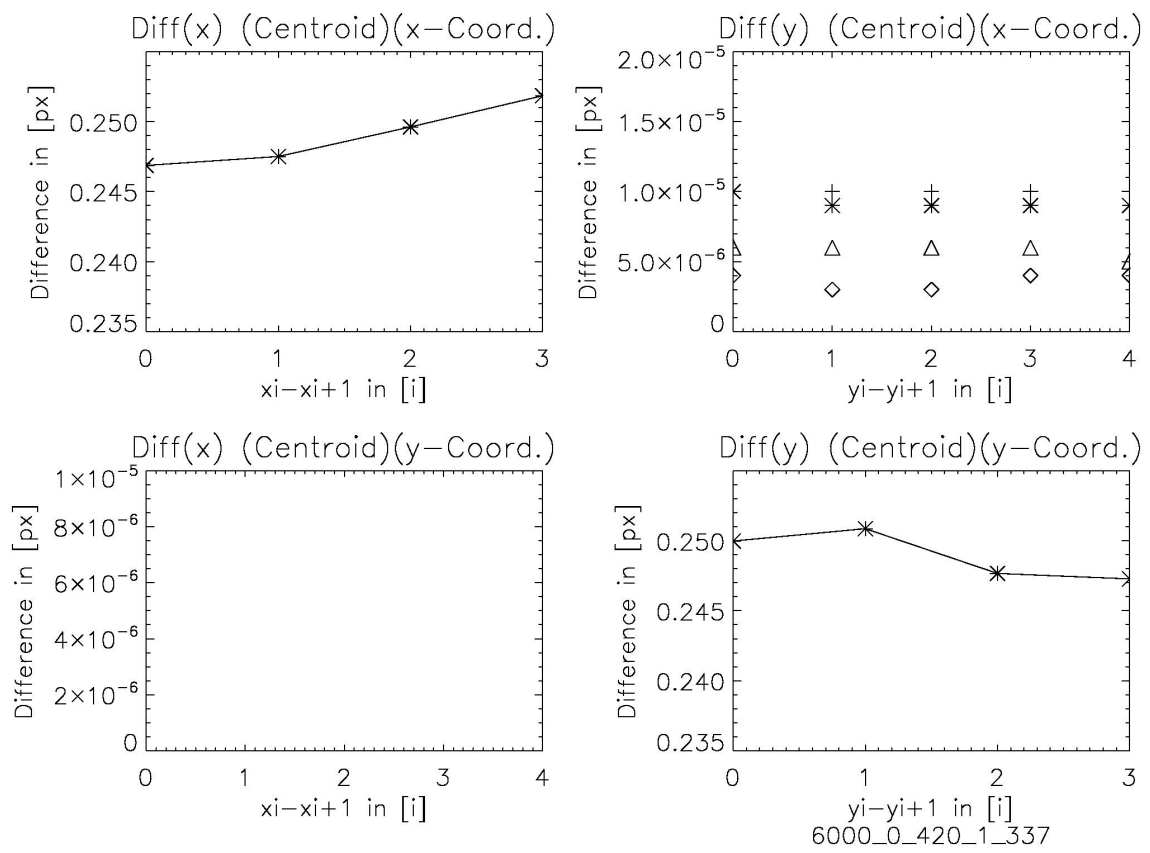


Figure D.11: 6000 K 1.337

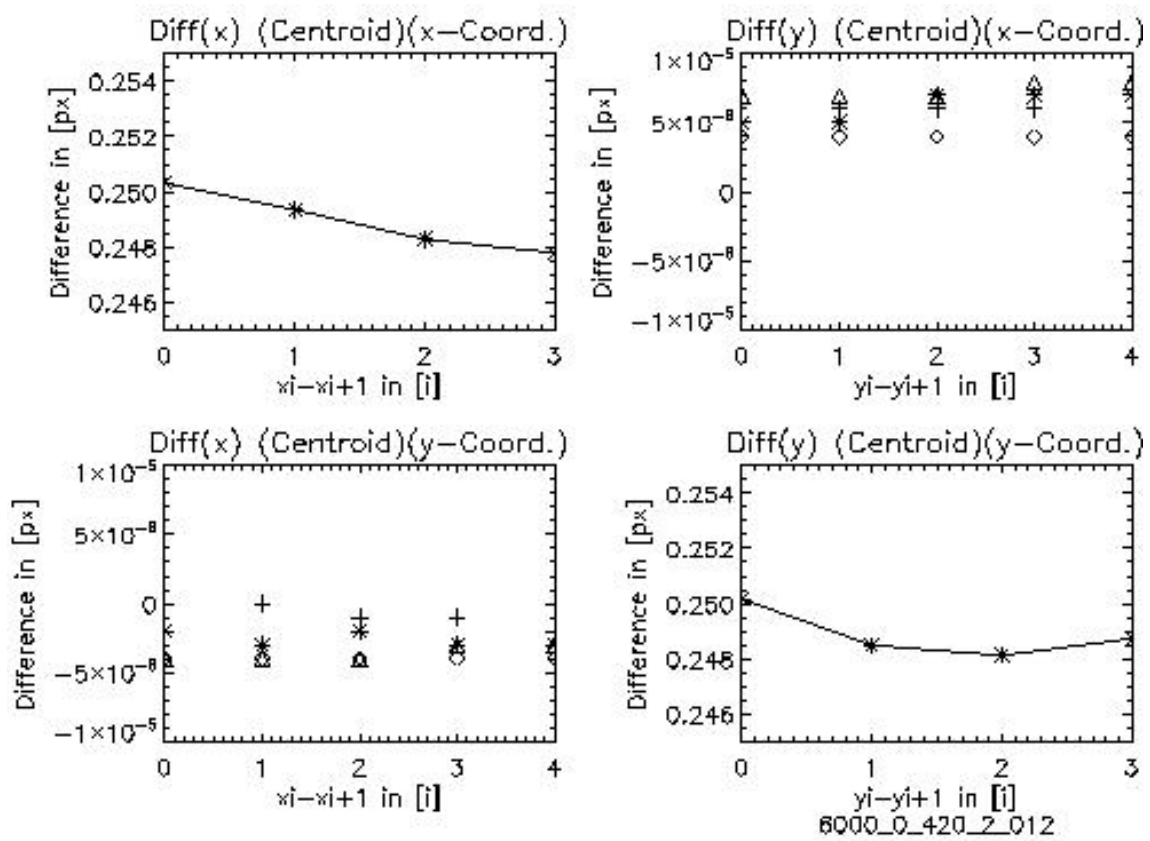


Figure D.12: 6000 K 2.012

APPENDIX D. CENTROID RESULTS FOR ALL REBINNED PSFS

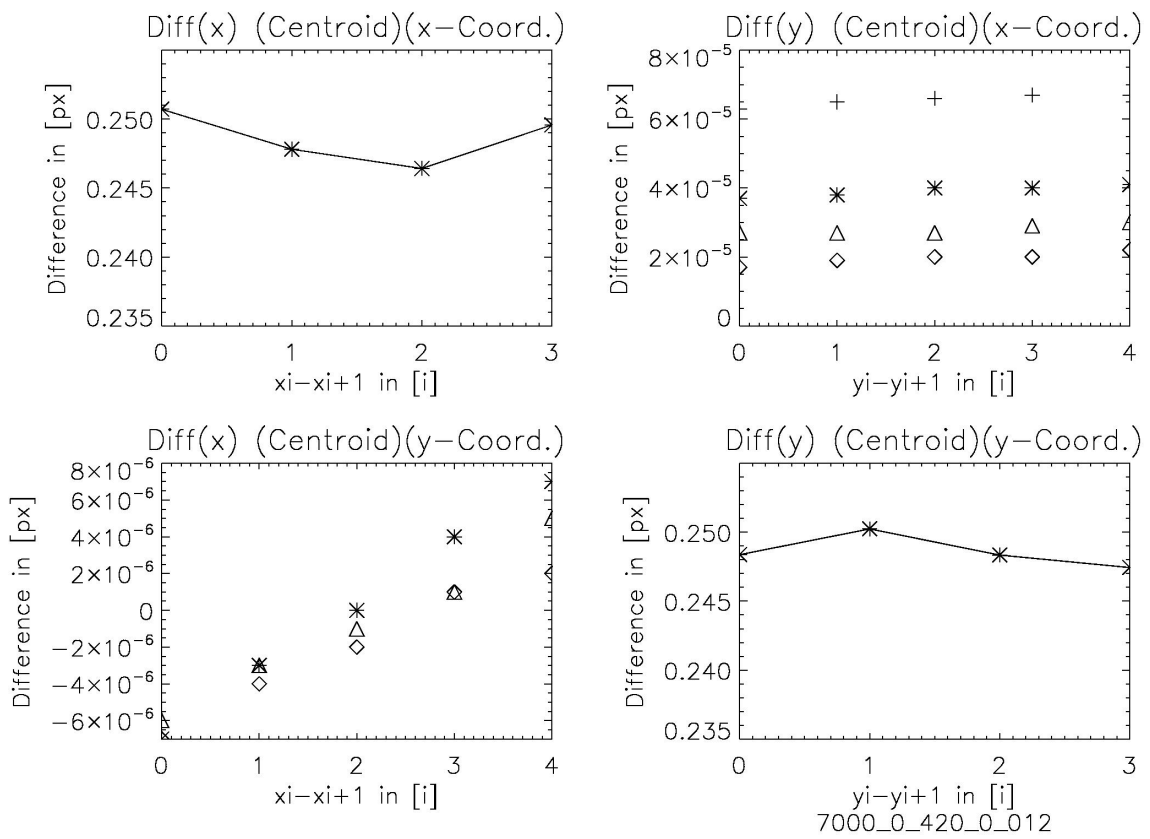


Figure D.13: 7000 K 0.012

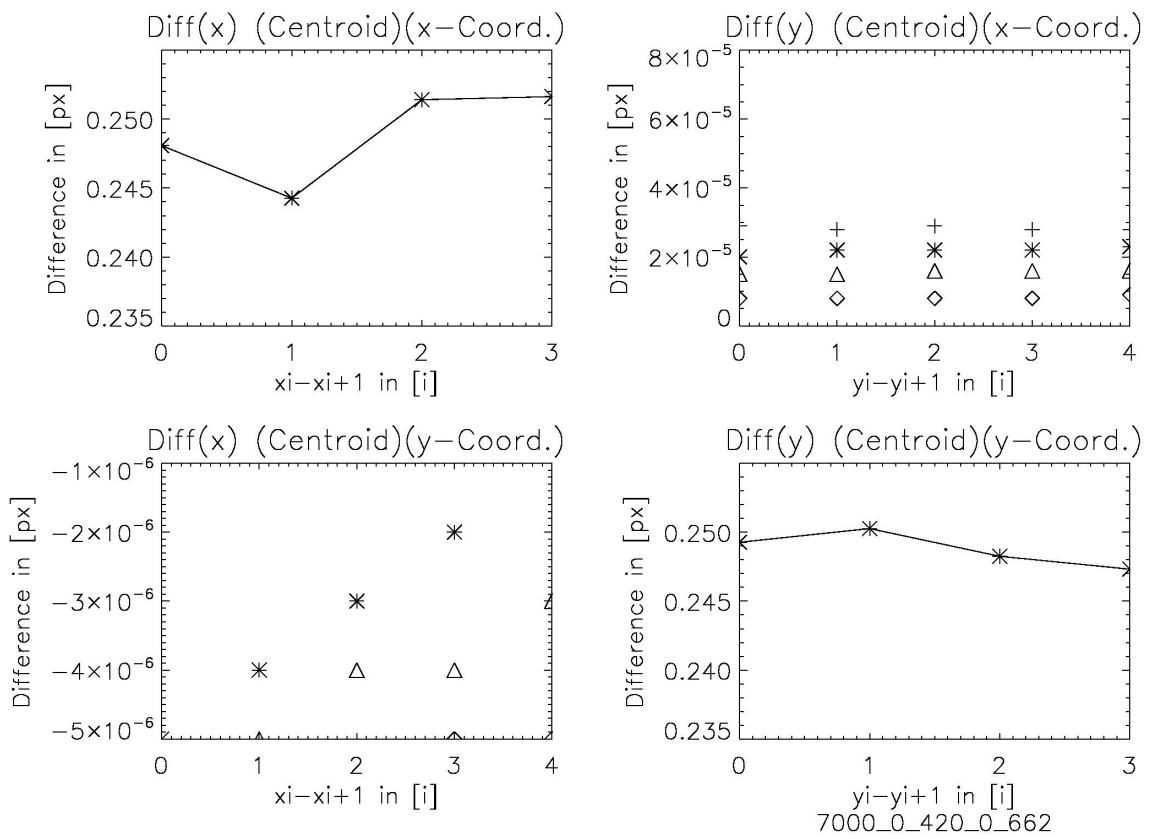


Figure D.14: 7000 K 0.662

APPENDIX D. CENTROID RESULTS FOR ALL REBINNED PSFS

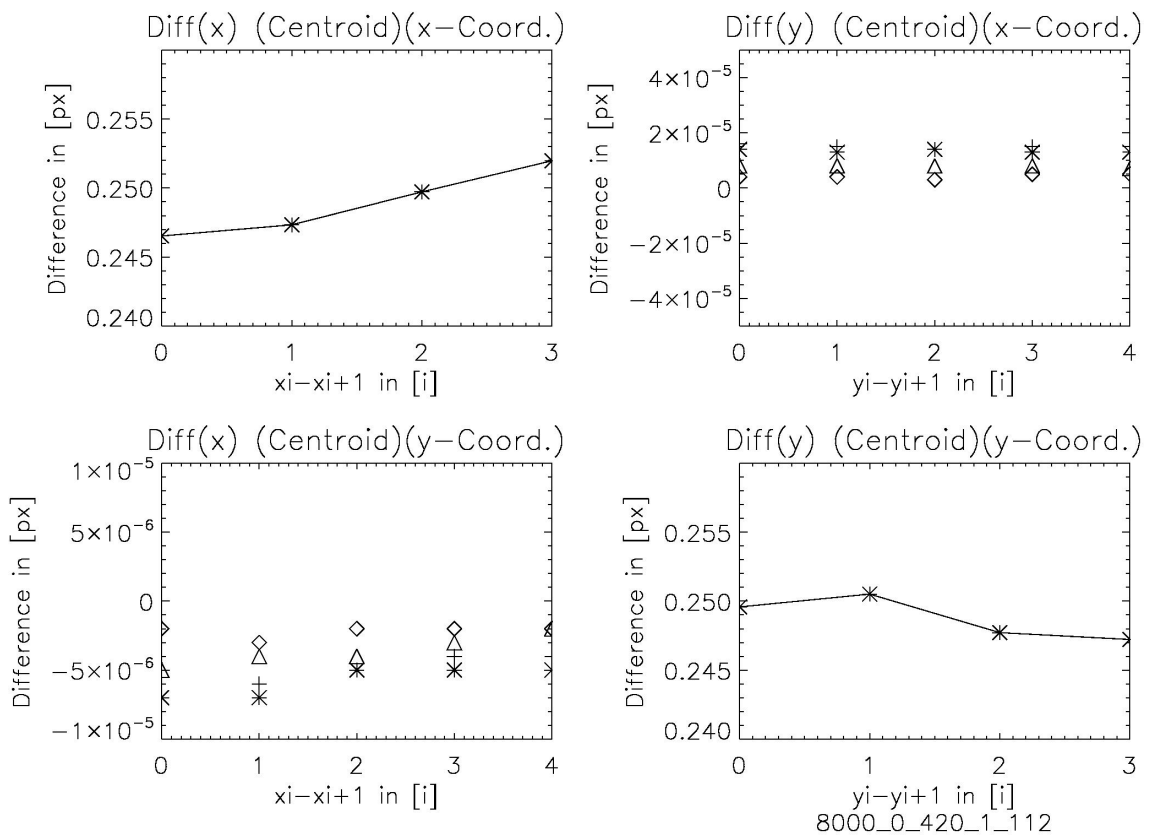


Figure D.15: 8000 K 1.112

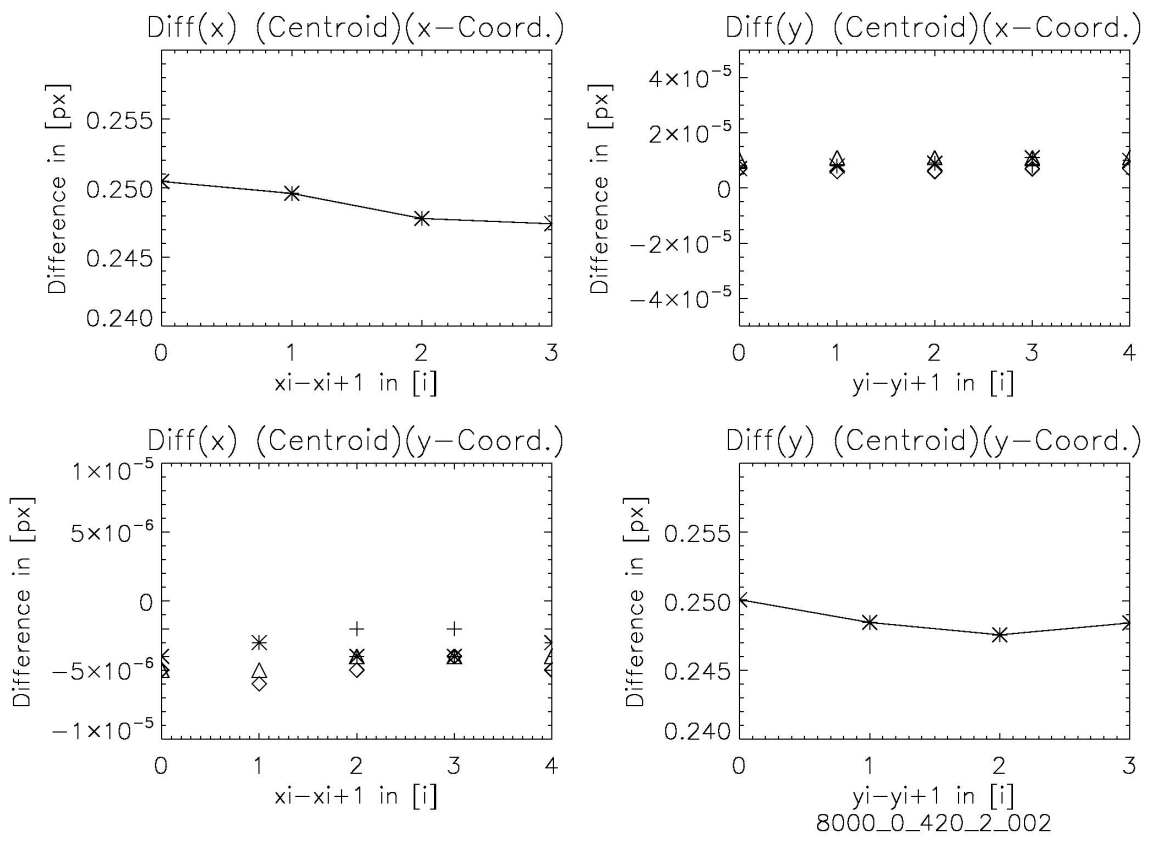


Figure D.16: 8000 K 2.012

Curriculum Vitae



Gabriel Stöckle

born December 27, 1976

marital status: unmarried

nationality : german

adress: Im Giesental 26
D-79112 Freiburg

contact: stoeckle@astro.univie.ac.at/gabriel.stoeckle@web.de
(+43)0699/11388029

Education

1996 Abitur, Deutsch-Französisches Gymnasium, Freiburg i. Br. main subjects : French, German, mathematics, biology, chemistry

1997 - 1999 Academic studies of physics at Albert-Ludwigs University, Freiburg

1999 - 2000 Academic studies of physics at Ruprecht-Karls University, Heidelberg

2001 - dato Academic studies of astronomy at Vienna University, "Alma Mater Rudolphina Vindobonensis", Wien

Publications

"100 Stunden Astronomie - Zufriedene Gesichter in Deutschland", Thomas Posch und Gabriel Stöckle, "Sterne und Weltraum", pp. 96-98, Issue 7/09

Conferences, Talks

CoRoT Week 7, Granada, Dec 14-17, 2004

” Gibt es Leben auf anderen Planeten? “, Kinderuni Wien, 2005

” Modern Astrometry/Parallax determination with CoRoT“, Space Technology Education Conference, Braunschweig, May 9-12, 2006

1st CoRoT International Symposium, Paris, Feb 02-05, 2009

Work experience

07/08 1994,95,96 Ganter Brewery, Freiburg, holiday job

07 1996 - 06 1997 Military service at EUROCORPS, Strasbourg (F)

07 1997 - 07 1999 Artists agency Tournado, Freiburg

10 1998 - 07 1999 Institute for applied mathematics, Freiburg , Webadmin

11/12 2000 MediDOK Softwareentwicklung, Heidelberg

07-10 2001+02 2002 PEARL Consulting&Communication, Buggingen near Freiburg:
network- and database administration

07-09 2002 Working student, tax office Müllheim near Freiburg

01 2003 - 03 2004 Show operator, planetarium Vienna

07-09 2003 + 02 2004 General Electric Network Solutions, Freiburg , feasibility studies,
visualization of land registry data, contracting authority: department of the interior
of the state of Hesse, Wiesbaden

03 2004 - dato, diploma thesis under surveillance of Univ.-Prof. Maria G. Firneis and
Univ.-Prof. DI Dr. Werner W. Weiss ”Feasibility of Parallaxes with COROT“

03 2006 - dato, marketing department of Silver Server, Vienna

Computer literacy

Content Management Systems: TYPO3, Mambo

Software development: Rational Suite

Programming languages: IDL, C, PHP, PERL, Shell, VB, Java, Pascal, Fortran

Network technology: implementation and support of students hostels networks in stu-
dents hall of residence, Starkfriedheim, Vienna with 250 users

Miscellaneous

Languages: French (first foreign language), English

Social commitment: students representative for the faculty of earth sciences (WS 2005-WS 2007), students representative of the branch of astronomy (WS 2005-WS 2007), Vienna University, representative of students hall of residence Starkfriedheim, Vienna (WS 2001 - SS 2005),

Personal interests: music, astronomy, football, hiking