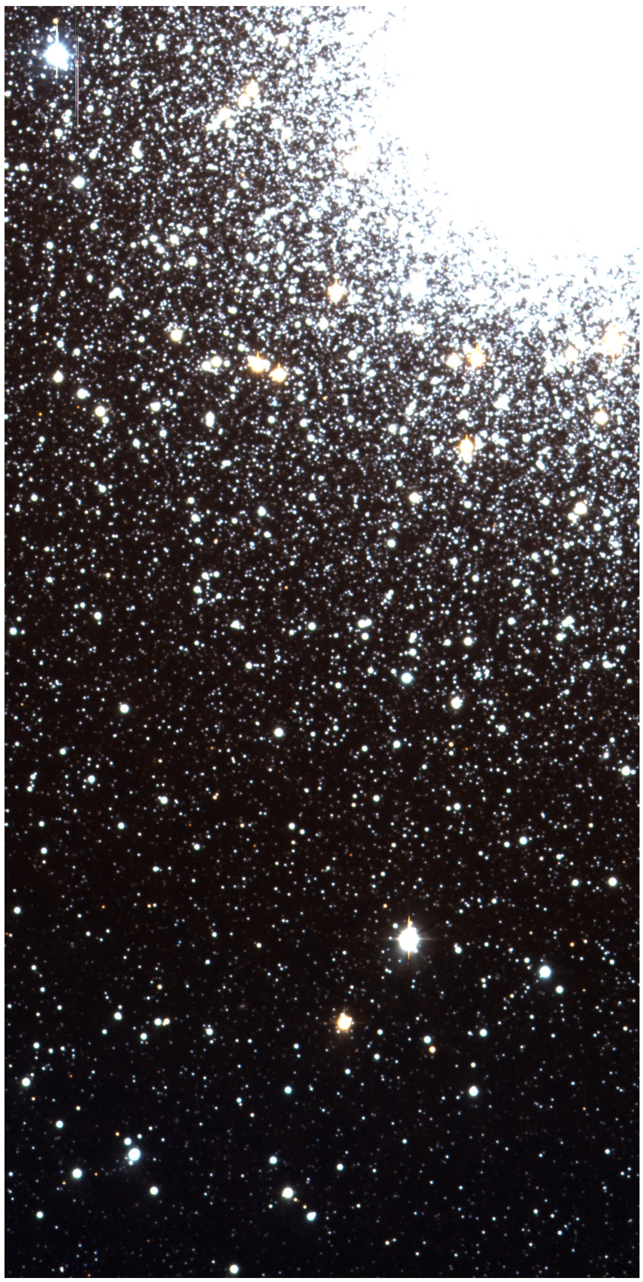
Groot onderzoek
A search for planets and other variables in 47 Tucanae

E.A.M. Westra
under the supervision of
prof. dr. P.D. Sackett

24th February 2003



Contents

Abstract

For this project stars of the globular cluster 47 Tucanae were observed. These observations were made using the 40 inch telescope at the Mount Stromlo Siding Springs Observatory, Australia, using the WFI detector.
806 images were taken using a special V+R filter. These images were processed using the Difference Image Analysis method (also known as image subtraction) [Woźniak 2000].
The original goal of this project was to find planets in this cluster, but no planetary transits were found in this project due to the used techniques. Future work on this data may find planets.
The goal of this project has been adjusted to find the obvious variable stars. 13 variable stars are found: 6 RR Lyrae, 5 binary star systems and 2 long period variables. Of these 13 variables 2 RR Lyrae and 2 binary star systems are already known by Kaluzny et al [Kaluzny et al 1998].

1 Introduction

1.1 General

A part of the Astronomy curriculum at the Kapteyn Institute is to do research. First, in the third year, a so called 'Klein Onderzoek' is done, which is a first acquaintance with research in Astronomy. At the end of the study much experience is obtained doing research in the 'sequel' of the 'Klein Onderzoek': the 'Groot Onderzoek'.
This 'Groot Onderzoek' talks about finding variable stars in the globular cluster 47 Tucanae using image subtraction. The project was conducted under the supervision of prof. dr. Penny D. Sackett.

1.2 Methods of discovery

After the discovery of Uranus, Neptune, Pluto and Charon people are now looking across the borders of our solar system. In order to find planets outside our Solar System it is possible to use many methods of finding a planet. The methods can be divided into two categories: direct and indirect.
The direct way is to point a telescope towards a star and wait until a planet is seen. But because the star will over-shine the planet it will be impossible to directly see the planet. However nowadays promising results are/will be made by using nulling-interferometry. This method uses the fact that two lightwaves in opposite phase will cancel each other out. In this way they can cancel out the light of the star and all that remains is the light of the planet. This method will be used in two projects: GENIE, Ground-based European Nulling Interferometer Experiment [1] (which will be online in approximately 2006), and Darwin [2] (which will be launched approximately in 2015).
There are several ways to detect planets around a star indirectly.
One method (using astrometry) is to look at the position of a star to see if it changes due to the gravitational pull of orbiting planets. An example of this type of research can be found in the work of Eckart et al [Eckart et al 1997]. Though they did not look for planets, but for black holes, the principle is the same.
But if the movement of the star is so small that it can't be seen using astrometry, it is possible to use spectrometry to determine the radial velocity, the velocity in the direction of the line of sight, of the star. For this method very high resolution spectra of a star must be taken. From the spectral lines the radial velocity is measured. The change in the radial velocity can be caused due to planets gravitationally pulling the star.
A final method is making use of occultation. In this case a planet will block an amount of light from the star when (partially) eclipsing it.

1.3 Transiting planets

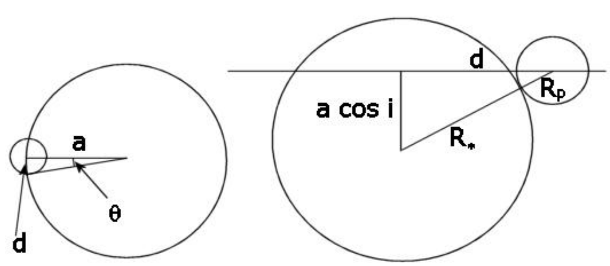


Figure 1: Model of transiting planet (taken from Sackett [Sackett 1998])

The type of planets this project was aiming for were Jupiter-like planets (radius $R_p \sim 0.1 R_\odot$ and mass $M_p \sim M_J$ at very close orbit to its star (semi-major axis $a < 0.1$ AU). These so called "hot-Jupiters" have a typical orbital period of several days. A Jupiter-like planet at a distance $a = 0.1$ AU of a sun like star would have a period of 0.032 year = 11.6^d, using

$$P = \sqrt{\frac{a^3}{M_*}} \tag{1}$$

with P the period in years, a the semi-major axis in AU and M_* the stellar mass in $M_\odot$. Planets of approximately the same mass, but closer to the star will (of course) have a shorter period.
The definition of the duration of the transit t_T is the part of revolution that the planet covers the star multiplied with the period. All the variables are indicated in figure 1 with i being the inclination of the planetary orbit with respect to the line of sight.

$$\text{part of revolution...} = 2 * \frac{\theta}{2\pi} \tag{2}$$

$$d = \sqrt{(R_p + R_*)^2 - a^2 \cos^2 i} \tag{3}$$

$$t_T = \frac{2\theta}{2\pi} P = \frac{P}{\pi} \arctan\left(\frac{d}{a}\right) = \frac{P}{\pi} \arctan \frac{\sqrt{(R_p + R_*)^2 - a^2 \cos^2 i}}{a} \tag{4}$$

with $a \gg R_* + R_p$ this becomes

$$t_T = \frac{P}{\pi} \sqrt{\left(\frac{R_* + R_p}{a}\right)^2 - \cos^2 i} \tag{5}$$

with t_T and P in days and R_*, R_p and a in AU. This equation has no solution when $\left(\frac{R_*+R_p}{a}\right)^2 - \cos^2 i < 0$. This is the case when $\left|\frac{R_*+R_p}{a}\right| < |\cos i|$. Both sides of the equation are ≥ 0 in the region of inclination ($\lesssim \frac{\pi}{2}$) used. So this means that $R_* + R_p < a \cos i$. And this is a logical result because (see figure 1) there won't be a planet covering the star when $a \cos i \geq R_* + R_p$.
The next thing to know about the transit is what the change in flux will be due to the planet covering a part of the stellar disc. This from the point of view of knowing which signal-to-noise (S/N) ratio is necessary for the observations. The maximum fractional change in the observed flux is given by

$$\text{maximum } \frac{|\delta F_\lambda|}{F_\lambda} = \frac{\text{light covered by planet}}{\text{light emitted by star and planet}} = \frac{\pi F_{\lambda,*} R_p^2}{\pi F_{\lambda,*} R_*^2 + \pi F_{\lambda,p} R_p^2}. \tag{6}$$

Since the light emitted by the planet is negligible compared to that of the star this equation simplifies to

$$\text{maximum } \frac{|\delta F_\lambda|}{F_\lambda} \simeq \left(\frac{R_p}{R_*}\right)^2. \tag{7}$$

This means that a detection of a Jupiter-like planet ($R_P = R_J \simeq 0.1 R_\odot$) at a sun like star ($R_* = R_\odot$) requires a S/N ratio of at least 100 to be detected. Also very important to know is that when a planet would survive the expanding of the envelope of a star that has become a giant ($R_* \sim 10 R_\odot$) a S/N ratio of 10^4 would be required. This is extremely hard (if not impossible) to get.

2 A search in 47 Tucanae

Other persons already had a look at 47 Tucanae, especially at the core [Gilliland et al 2000], and didn't find any planet. A reason why they think that they did not find any planets was the low metallicity of 47 Tucanae compared to the solar neighborhood. Metallicity when low could cause in a protoplanetary nebula less dust grains, which are the basics for planet formation. The less dust grains were formed in this stage the lower the frequency of planet formation.
Another reason they suggested was that the stellar density in/near the core of the globular cluster was so high ($\sim 10^3 M_\odot pc^{-3}$) that planet orbits would change drastically. They also suggested that in the crowded core tidal dissipation would destroy hot Jupiters. A final suggestion was that crowding limits planet formation.
As can be seen crowding is (probably) one of the influencing cause of a lack of planets in the core of 47 Tucanae. Therefore it would be better to look more in the outskirts of the cluster. For this telescopes with a large field of view were very useful.
Though these findings did not sound very promising to look at 47 Tucanae several reasons could be given to justify another look:
As already mentioned above the outskirts of 47 Tucanae are less dense and therefore the chances that a planet would come to formation would be increased.
This lower density also meant less crowding. The less crowding the less contaminated the light of single stars would be. The difference image analysis (DIA) of Woźniak [Woźniak 2000] was designed to work on more crowded areas.
Also more metal poor stars were present in the proposed field compared to the solar neighborhood. Metal poor stars are hotter and therefore smaller than 'normal' metallicity stars of the same mass. Being smaller would increase the depth of the dip caused by the planet transition (see section 1.3).
If the properties of this project are compared with that of Gilliland et al [Gilliland et al 2000] several differences could be found. The first was that they have used the Hubble Space Telescope. This meant continuous observation during all 8.3 days of their observation time. This project on the other hand was ground based so the observations were not continuous. But the allocated time was 30 days of which 18 have proven to be useful (which was reasonably good if normally only 50% would be useful due to weather conditions).
The observations of Gilliland et al were about 3 to 4 magnitudes below the turn-off, which corresponds with low mass stars ($\lesssim 0.65 M_\odot$), while this project was targeting at approximately 2 magnitudes below the turn-off, which are approximately solar mass stars ($\sim 1 M_\odot$).
To increase the signal-to-noise (S/N) ratio a special filter (see section 3.1) was made for this project increasing it with a factor of 2.
All in all these projects did not compete with each other, but they were complementary to one another.

2.1 Goal

The original goal of this project was to find planets in 47 Tucanae. During the course of the project it became clear that no planets were going to be found with the techniques used for this project. Therefore the goal was adjusted to find all variable stars in the data, but this was finally reduced to find all, what were called, "3σ-variables", stars that were clearly variable.
Of course, if a planet detection was found, that would get the major priority.
Because the beginning of this project took much time, only one CCD of the mosaic images (see section 3.1) was processed using the DIA method; this is CCD 3.

2.2 Collaborators

The observations for this project were done by Ken Freeman and Terry Bridges.
The software for the difference image analysis was provided by Przemyslaw Woźniak [Woźniak 2000]. He made suggestions about the parameters for the software.
The source code was run through by Penny Sackett to make it compatible with the local hardware. She also tested the software to see if it functioned and found doing so a variable star.
David Weldrake worked simultaneously on this project. He provided suggestions about i.a. variables he found, converting data into relative magnitudes.
Michelle Doherty provided via Terry Bridges a database of V- and I-magnitudes of the stars observed.

3 Technical details

3.1 General information

The images were taken with the 40 inch Wide Field Imager (WFI) at the Siding Springs Observatory in Australia. An image is constructed of a 2 by 4 mosaic of CCDs. Each of the CCDs consists of 2k by 4k pixels, which gives a total format of 8k by 8k pixels. Each pixel has a size of 15 micron, which gives a scale of 0.38"/pix at the Cass focus of the telescope. This results in a field-of view of 52 arcmin on a side (diagonal $\sim$1.2 deg).
The observing period was from September 30th 2000 up and until October 25th 2000. In this period there were made 5 images in V and 5 images in I. Data from these were used in this project only to provide the V- and I- magnitude and from there the V-I color. The V- and I-images are only a small part of the set.
Most of the images (806) were made in the special V+R filter. This filter was specially made for this project. There are a few reasons why this filter was made and why it has been made in V and R [Freeman (private communication)]:
The CCDs are at their best around V and R, and it is possible to double the number of photons per unit time with this filter (compared to V or R alone).
Then the question arises: why not use an even wider band? As 47 Tucanae is so far South, the airmass will always be larger than 1.3. The atmosphere disperses the light of stars. Close to the zenith this problem is negligible, but when observations are made closer to the horizon the distortion gets more important. Stellar images will look like short spectra with the direction of dispersion towards the zenith. So it is necessary to keep the band narrow enough so that the length of the spectrum is not significant compared with the seeing for the range of airmass being used (from 1.3 to more than 2). Therefore V+R is as wide as one can make it.
The filter is made of glasses cemented together as a sandwich. One glass provides the blue cutoff (same as for Cousins V) and the other the red cutoff (same as for Cousins R).
All of the images have an exposure-time of 300 seconds. CCD 3 had a gain of 1.96 e^-/adu and a read noise of 6.2 e^- [3].

3.2 Reduction

The unreduced images were reduced using IRAF (Image Reduction and Analysis Facility). First the images went through 'ccdproc' of the package 'mscred'. 'ccdproc' processes CCD images to correct and calibrate for detector readout defects, readout bias, zero level bias, dark counts, response, illumination and fringing. It also trims unwanted lines and columns of the CCD.
After this the images had to undergo linearity correction. In equation (8) the formula for this linearity correction [3] is given; a is the old value of the pixel, b the new value of the pixel and A_i are given in Table 2 [3].

$$b = a * (A_0 + A_1 * a + A_2 * a^2 + A_3 * a^3) \tag{8}$$

After this linearity correction the images had to be flatfielded. Flatfielding is used to correct pixel to pixel variations due to the different response of each pixel to light. The flatfielding was done using the twilight flatfield images. These images did also undergo 'ccdproc' and the linearity correction. Then they were checked whether or not they were saturated. For a list of the saturation level (see also section 3.3) see Table 1. None of the flatfield images were saturated.
At this point the flatfield images were combined using 'flatcombine'. Of CCD 1 through 6 [500:1500,1500:2500] was used, of CCD 7 [800:1700,2000:3000] and of CCD 8 [500:1300,2300:3000] [Bridges (private communication)]. These areas of the CCDs were used because the complete flatfield images were not uniform and the areas used seemed relatively uniform. The combination of the flatfields was done using the median.
When the twilight flatfield was finished (there are two types of flatfields: a twilight flatfield to correct for pixel to pixel variations and a dome flatfield for large scale variations), it was divided by a constant to be on average 1. The final step was to divide the data images by the flatfield to finalize the flatfielding. This division was done to correct for the response of each pixel.
With the reduced images the rest of the project was executed.

3.3 Seeing and saturation

CCD no.	saturation level	
	CGT[a]	this work
1	56000	55000/56000
2	53000	49000/51000[c]
3	42000	41000/43000
4	52000	?[b]
5	44000	43000/45000
6	56000	53000/55000[c]
7	54000	48000/50000[b]
8	55000	50000/52000[c]

Table 1: Saturation levels.

[a]at WFI performance information site [3]
[b]very hard to determine
[c]lower than CGT, saturated stars were found at level of CGT

CCD no.	A_0	A_1	A_2	A_3
1	1.0	0.0	3.86e-12	0.0
2	1.0	0.0	3.18e-12	0.0
3	1.0	0.0	8.43e-12	0.0
4	1.02753	-1.6555e-6	1.7791e-11	0.0
5	1.0	0.0	4.26e-12	0.0
6	1.0	0.0	5.93e-12	0.0
7	1.0	0.0	-5.50957e-12	2.39334e-16
8	1.0	0.0	5.37e-12	0.0

Table 2: Coefficients for linear correction for each of the CCDs [3]

Having the reduced images the seeing of the images and the saturation level could be determined.
The seeing of an image is the quality of the observing conditions at the time of observing. The higher the seeing the higher the turbulence in the Earth atmosphere. A lot of turbulence causes small erratic movement in the image position, which leads to a blurred, distorted and enlarged stellar image.
When the profile of a star is given (see figure 2) the seeing is the full width at the half of the maximum of the star (FWHM).

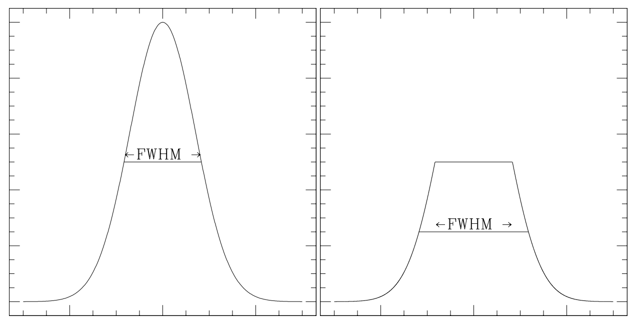


Figure 2: In the left figure the profile of a star is given with indicated its FWHM. In the right figure a the same star is indicated, but now it is saturated.

The seeing was determined using the package 'imexamine' of IRAF. An example of the output while applying imexamine to a star using the 'aperture sum' function is given here:

```
#   COL    LINE    COORDINATES
#     R    MAG    FLUX     SKY    PEAK    E    PA BETA ENCLOSED   MOFFAT DIRECT
 947.32 2037.90 947.32 2037.90
  14.62  14.73  12772.   505.1   307.9 0.01   84 3.00     5.30     4.88   4.88
```

In the final column below DIRECT the seeing is given in pixels.

The seeing was determined by measuring stars which had a profile like the star in figure 4(a). Stars like in figure 4(b) were just saturated. Saturation means that the CCD comes in the region below the maximum charge of electrons that the CCD pixel can handle. When more light falls onto the CCD charge begins to spill onto adjoining pixels. When even more light would fall onto the CCD the charge would spill onto a larger area of the CCD. Figure 4(c) is a good example for this. In this figure it is visible that in the center so much charge was built up that the analog to digital converter (the device in the CCD that converts charge into ADU, analog digital units) started to clip. This causes the star to look like it is wrapped in the center.

Using this it was possible to determine what the saturation level of a CCD was. Again 'imexamine' in IRAF was used. This time the 'statistics' function was used. Here below an output example is given. The output speaks for itself. The useful values are in this case in the 'max(imum)' column. First the seeing of several (unsaturated) stars was determined, then the seeing of stars that are on the edge of saturation was determined, so they had a profile just between figure 4(a) and 4(b). Then the maximum value of that star was determined. This was the required value. In Table 1 an overview can be found of the values found using this method and by Chris Tinney (CGT) [3].

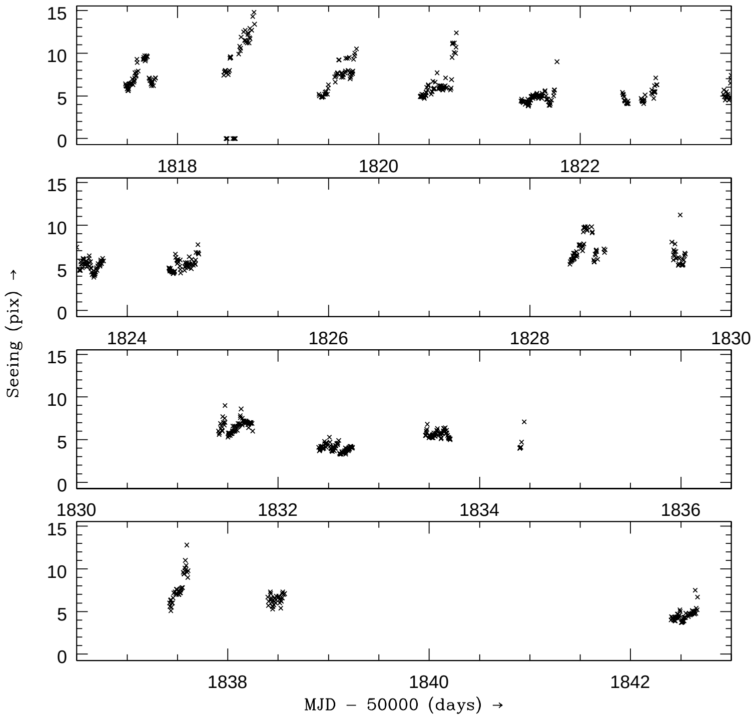
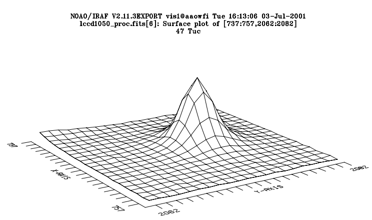


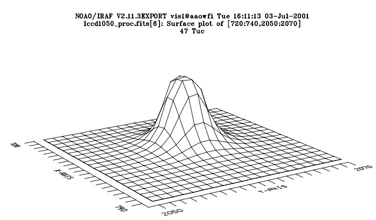
Figure 3: Seeing in pixels versus time in days. 1 pixel is equivalent to 0.38".

```
#             SECTION     NPIX     MEAN   MEDIAN   STDDEV      MIN      MAX
 [964:988,2043:2067]       625    1948.    1854.    347.1    1720.    5220.
[986:1010,2081:2105]       625    4108.    2024.    6935.    1767.   44463.
```

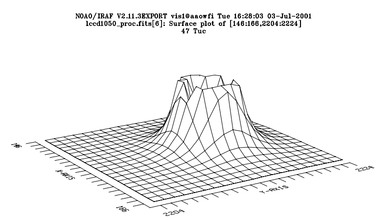
In figure 3 it is visible that the seeing varied a lot in time. It varied from 3.3 pixels on October 15th (MJD - 50000 = 1832) up to 14.8 pixels on October 1st (MJD - 50000 = 1818).



(a) A normal not saturated star. As one can see a nice PSF. (For a definition of PSF see section 4.2.)



(b) A star that is just saturated. The top is just flattening, but is not yet starting to wrap.



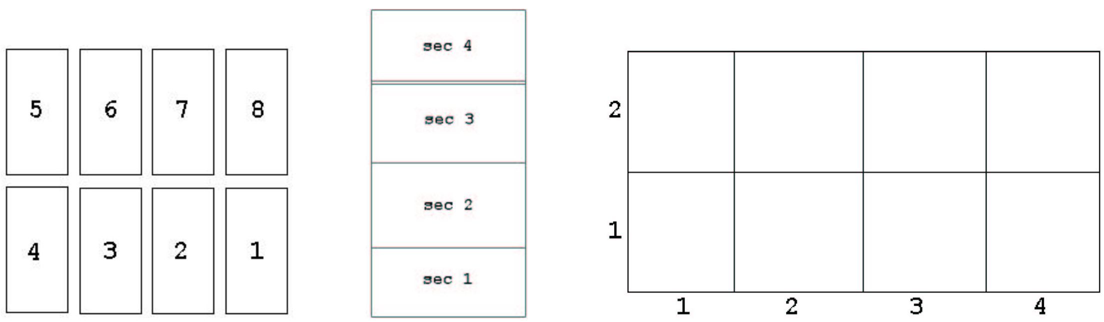
(c) A saturated star. The center of the star has wrapped to the inside.

Figure 4: Three different "kind" of stars: normal, just saturated and saturated.

4 Image processing

As mentioned in section 3.1 there are 8 CCDs per image. For this project, due to time constraints only one CCD was analyzed with image subtraction. The choice fell on CCD 3, because this was the most crowded CCD of all 8 and also the core of 47 Tucanae was located on this CCD. Because Woźniak claimed that his program would work well for crowded areas on images, unlike DoPhot, [Woźniak 2000] this would be a great testing opportunity.
The programs being talked about in this section are part of two script made by Woźniak [Woźniak 2000]. The first script prepared a so called template image (section 4.3), an image to which all the other images were compared. The second script was the photometric pipeline (section 4.4).
Together with all the used programs and parameter files the scripts could be changed to taste. This was done to make optimal use of the hardware used. Details of these changes can be found in the following sections.

4.1 Preparing the images



(a) Mosaic of **CCD**s

(b) CCD divided into **sections**

(c) Section divided into **subsections**

Figure 5: How the mosaic is divided into CCDs divided into sections divided into subsections

After the reduction process the images were given as mosaic fits files (figure 5(a)). These had to be split up into separate files. Each of these files represented one *CCD*. Due to hardware restrictions the CCD had to be divided into 4 *sections* (figure 5(b)) of a width of 2044 pixels and a height of 1024 pixels. Because the program was working with multiples of 512 pixels a CCD was divided as follows (only y-coordinates): pixel coordinates 1-1024, 1025-2048, 2049-3072 and 3071-4094 for sections 1 to 4 respectively. That the width was not a multiple of 512 was handled by the programs. Eventually the program divided the sections into 8 (4 by 2) *subsections*. Each file had its data in standard float format and had to be changed to the OGLE ushort format, so the programs was able to read the images.
From this point on the remaining processing of the images is described per *section*.

4.2 Creating a mask

Before the programs could do their job a mask had to be prepared. This mask was used in both the template and pipeline script. It prevents the programs from using certain pixels. For example when a star was oversaturated the star didn't look like a normal star (see figure 4(c)). Some of those pixels could not be used by the programs because they had a value that meant saturation, i.e. the pixel value was too high. But in other pixels around a saturated star the programs could (miraculously) see a PSF. The PSF (point spread function) is the response of an imaging system to a point source, in this case a star. As a result of this the programs found a lot of variables which were not real. To eliminate these problems together with the completely saturated core which fell precisely in the used CCD, the mask was used.

For the mask the best seeing image was used as in this image the most stars were saturated so it contained the most saturated pixels. This was the reference frame (see section 4.3 for the definition of the reference frame.) The saturation level was chosen to be 40,000. This because the saturation value of CCD 3 (given in Table 1) was approximately 42,000 if all the saturated stars had to be a bit below this value. Also stars with pixel values in this region turned out to be of magnitudes of approximately 17 and brighter. These stars would be giants and therefore a planet would be extremely hard to be detected (see section 1.3).
A method of making this mask was to make first an image in which all pixels were set to 0 except those in the reference image which were 40,000 or higher. Those were set to 1. Using IRAF this can be done using

```
imexpr '(a > 40000)' mask.fits a=lccd0817 dims=2044,1024
```

The next step was to convert this file to a list of all coordinates and the values of those coordinates. This was achieved using

```
listpixels mask.fits > mask.txt
```

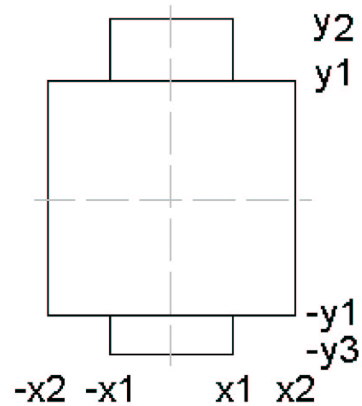


Figure 6: Shape of pixel mask. Values of x1 etc are in source code of the mask making program in appendix A.

All the coordinates and pixel values were saved in a file. Outside IRAF the compiled source code of the mask making program (see appendix A) was run using the list as input data. The program created several files which were IRAF scripts. It was necessary to only start the first one, because it would start the others by itself. Meanwhile a new empty image had to be created (filled with only 1's).

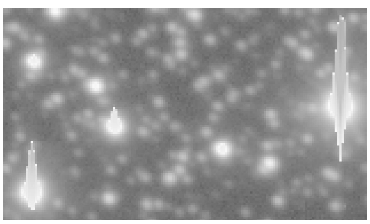


Figure 7: Example of 3 magnitudes of spikes. Grayscale is logarithmic to enhance the spikes.

The effect of the script was to make in the mask for **each** pixel that was saturated, i.e. above 40,000, the shape as given in figure 6. The saturated pixel is located on the crossing of the two dashed lines.
The most important reason that the pixel mask had this shape was that very saturated stars started to show 'spikes' (see figure 7). In figure 7 it is visible that a barely saturated star (left of the middle) has only a small spike in the upward direction. As there are more saturated stars the spikes will be bigger. Not only the spike in the upward direction will grow, but also a spike in downward direction will become visible and grow when a star is more saturated.
Because most of the pixels of a saturated star are more than 40,000 the complete star will be blocked. Due to the fact that there are some pixels in the spike that are above 40,000 the spike will also be blocked.
When the IRAF scripts were run the mask was almost completed. It only had to be converted to the OGLE ushort format and given the proper name. At this point the mask was made properly.

4.3 Template

At this point the images were in a format (and size) to be read by the script. The next thing that had to be done was to create a *template*. The template was the result of stacking several of the best of images (lowest seeing and low background). For this template 37 images were used (see Table 3).
To achieve this, first a coordinate reference image (hereafter *reference*) was chosen to which all other images were to be resampled. The image numbered *lccd0817* was used as the reference image. The choice of this image was based on the fact that it was of a very good series of low seeing images. In particular this image had the best seeing of all images, viz. 3.3 pixels. After this the PSF of the template was determined in order to perform profile photometry on the difference images.
Profile photometry uses a 2 dimensional function to fit the profile of a star, usually a Gaussian function. To get the brightness of the star, the function is integrated over those 2 dimensions.
Another method to get photometry makes use of counting the values of the pixels within a certain radius of the centroid of the star. This is called aperture photometry.
It should be clear that in a very crowded area profile photometry is better than aperture photometry, because with aperture photometry too many pixels of a star are contaminated by light of neighboring

lccd0294	lccd0798	lccd0816	lccd0822	lccd0829	lccd1038	lccd1057
lccd0295	lccd0799	lccd0817	lccd0823	lccd0830	lccd1039	
lccd0778	lccd0800	lccd0818	lccd0824	lccd0831	lccd1051	
lccd0779	lccd0801	lccd0819	lccd0826	lccd0832	lccd1052	
lccd0780	lccd0804	lccd0820	lccd0827	lccd0834	lccd1054	
lccd0786	lccd0815	lccd0821	lccd0828	lccd0836	lccd1055	

Table 3: Overview of images used for the template

stars.
Essential differences were made in the template script made by Woźniak. The process of making a template and the pipeline were completely severed from each other. For more details look in appendix B and Woźniak [Woźniak 2000].
At the end of the template script, PSF coefficients, the mask and the reference image per subsection are placed in appropriate directories and not to forget the template. At this point the images were ready for the next steps of the pipeline.

4.4 Pipeline

The next step in the process was the pipeline. In here all the images (including the images used for the template) were undergoing the process of image subtraction.
To separate the process of making a template and the main part of the reduction a new structure of file locations was needed. This new structure was also a result of testing the program on these images in advance. From these tests it was clear that the computers locally available had too little memory to execute some programs of the pipeline using all images. Therefore the dataset was cut up into two *half*s. The distribution of images over the halfs was as follows: the first image was put in half 1, the second in half 2, the third in half 1, etc. This would be better than just dividing the complete dataset in two, because with this method if there would be a variable, there would not be too much of a change in its luminosity. So it was possible to regard these halfs as two sets of observations taken at almost the same time.
Within these two halfs the structure of subsections was maintained.
Because some information was needed from the template creating process, this information was distributed over both halfs.
The reduction continued with resampling all the images to the pixel grid of the reference. Then the seeing of the images was determined. The reference image was made to match the seeing of each image and then the difference images were created.
The next step was to find variable stars. This was done using a crude photometry on groups of variable pixels, which had the shape of the PSF. Of this group the centeroid was determined. Also some simplistic profile and aperture photometry on the template at the location of the variable was done.
The final step was the actual photometry at the location of the found variables.

4.5 Periodogram [Press et al 1992]

The data produced by the pipeline could just be plotted and reviewed by eye, but to go through thousands of lightcurves manually would be a tedious job. To speed up the process the computer was used to go through the data more rapidly.
Finding planetary transits or other types of variability, like e.g. binary star systems or RR Lyrae stars, would involve some kind of periodicity. When a computer is used to find periodicity almost immediately Fast Fourier Transformation (FFT) comes into mind. FFT is based on the normal Fourier transformation to get a spectrum.
The only problem with FFT is that it works only on evenly spaced data. In the field of Astronomy (especially the optical branch) this is almost impossible to get. Just think about gaps produced by having the Sun at the sky during the day. Also it is difficult to observe exactly every e.g. 5 minutes.
To solve this problem interpolation can be used: choose at which evenly spaced points (in time) the data have to be distributed onto, interpolate to these points and then use FFT. If it is only for a few missing points then interpolation is pretty easy to do. But when there is a whole gap in your data (like daytime) it becomes far more difficult.

Therefore another method has been developed by Lomb [Press et al 1992]. It evaluates the data only at times that are actually measured. To go into the mathematical details of this so called *Lomb normalized periodogram* would go too deep. For more mathematical information Numerical Recipes [Press et al 1992] should be read.
The result of this computational method is an array with an increasing sequence of frequencies and an array with the values of the Lomb normalized periodogram at those frequencies. Returned is the index of the frequency at which the value of the Lomb normalized periodogram is at its maximum. The last thing that is returned, is an estimate of the significance of that maximum against the hypothesis of being random noise. So a small value indicates a significant signal.

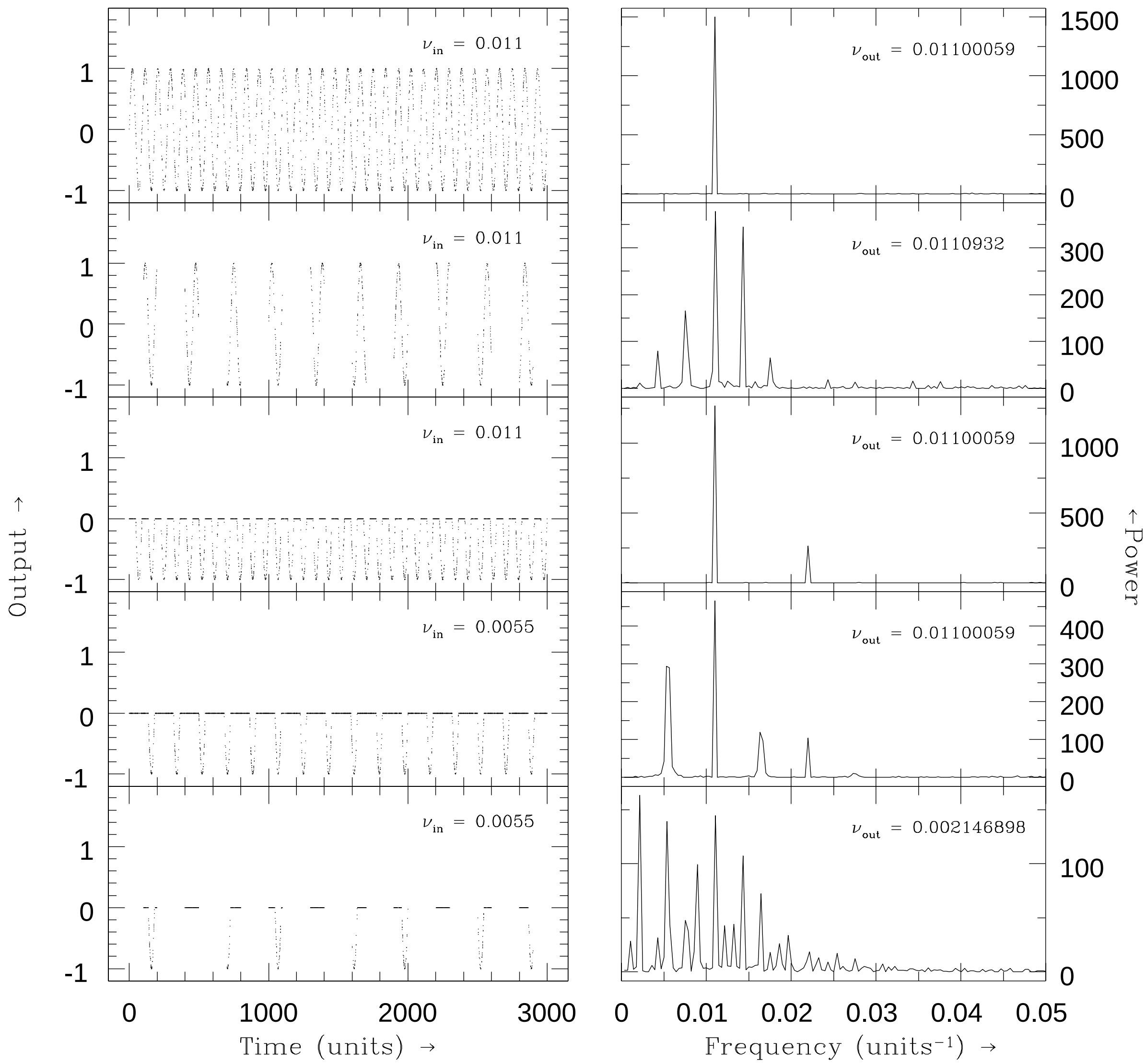


Figure 8: Results of testing the Lomb periodogram routine from the Numerical Recipies [Press et al 1992]. In the left column the input of the program has been plotted. In the right column its corresponding has been plotted. From top to bottom: plain sine-function, sine-function with gaps, topped sine-function, topped sine-function with only every other valley and topped sine-function with only every other valley with gaps.

Figure 8 gives a series of test data and its periodogram. In the left column the test data are plotted and in the right column the periodogram to match. To see if the program worked and how to interpret the results, input with predictable outcome were given.
The basis of the input was a sine-function with frequency $\nu_{in} = 0.011\ unit^{-1}$. The data on the x-axis was created using the `random()` function of C++. The data on the y-axis were the result of $\sin(2\pi\nu_{in}x)$.

In the periodogram on the x-axis the frequency was plotted versus at the y-axis the power of that frequency.
A simple sine-function should result in a periodogram with a strong signal at this frequency. And indeed, the frequency found by the program (ν_{out}) was 0.01101246. This was very close to the original frequency.
The next step was to check the influence of major gaps (like daytime) in the data. To do this daytime was simulated. A rather reasonable estimate for the length of observation in comparison to the length of no observation is 1 to 2. In this case the length of a day has been set to 300 units. Putting this into the program it gave back a frequency at 0.0110923 $unit^{-1}$. In the periodogram two side peaks were visible. The frequencies of these sidepeaks (from left to right: 0.0042938, 0.00751414, 0.0143126 and 0.017533 $unit^{-1}$) are separated by approximately $\frac{1}{\text{length of a day}}$ (= 0.00333...).
Until so far complete sinoids were used. A transiting planet would not cause increasing luminosity, it would only cause a dip. A first step to simulate this could be to use the negative part of the sine-function. This was plotted in the third row from above. In the periodogram the recovered frequency (ν_{out}) was 0.01100059 $unit^{-1}$. But again other peaks arose. These were overtones. These peaks arise at $n \cdot \nu_{out}$ with $n = 2, 3, \ldots$. It is logical that they decrease with increasing frequency, as these frequencies are less important.
But this was not even close of being a transit, because the time it takes a planet to eclipse the star is not equal to the time to do the rest of the revolution. Just for testing purposes it was assumed that the revolution would take three times the duration of the complete eclipse. So only every other negative part of the sine-function was passed through; the rest was set to 0. The recovered frequency was 0.01100059 $unit^{-1}$. This was the input frequency of the sine-function, but the real frequency was 0.5 times the frequency of the sine-function, because only every other dip was passed through. This was the cause of the side peaks. The most left peak was the real frequency and the others were overtones.
The final step was to see how a transiting planet would look like when it was observed, i.e. with gaps in the data. Some of the simulated transits occurred during daytime. The recovered frequency was 0.002146898 $unit^{-1}$. This was not even close to either the frequency of the sine-function or the actual frequency, which was 0.5 times frequency of the sine-function. But if one looked a bit closer, the frequency of 0.002146898 $unit^{-1}$ corresponded to a period of 465 units. This was close to the average of the distance between the peaks. I.e. when the difference between the first and second peak and the difference between the second and third peak were taken, the average of these two values was 456 units. In other words: the beat-frequency of two periodic signals was found.
This was only about the recovered frequency. In the periodogram the central frequency of approximately 0.011 $unit^{-1}$, side peaks due to the gaps in the data and some upper harmonics were recovered.
If all periodograms in figure 8 are compared to each other, a general trend is discovered: when modifications are made to a sine-function, like e.g. making gaps or chopping off the upper parts, the power of the periodogram decreases.

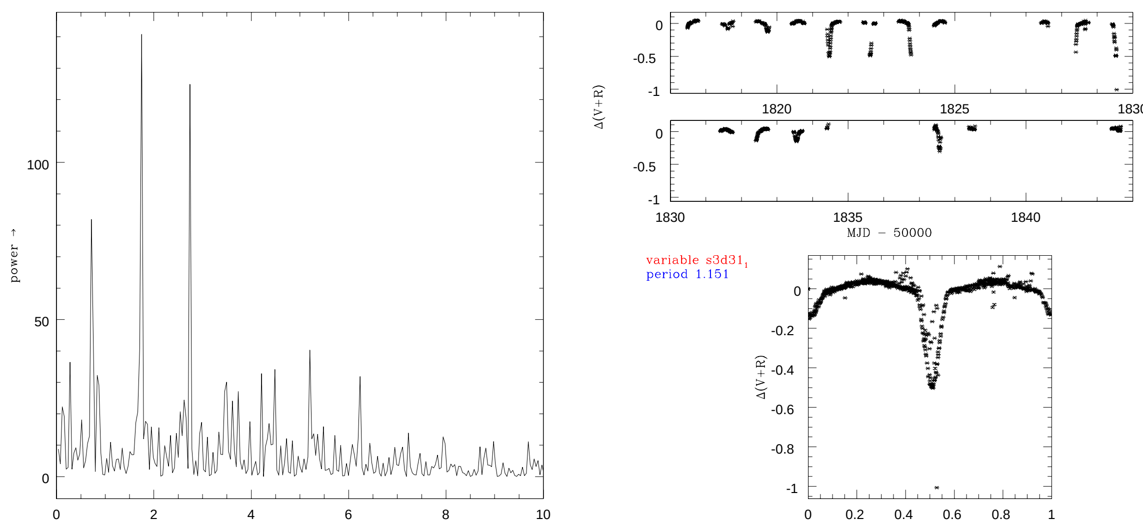


Figure 9: Periodogram, lightcurve and phase wrapped lightcurve of the variable in section 3 subsection 3 1

In figure 9 an example of a periodogram is given. It is the periodogram of the variable found in section

3 subsection 3 1 (see section 5.1). The highest peak in the periodogram is at a frequency of 1.747424 days^{-1}, which corresponds to a period of 0.5722709 days. In the right of figure 9 the lightcurve and the phase wrapped (see section 4.6) lightcurves are plotted. It is clearly visible from the lightcurve that there is a signal.
In section 4.6 this variable is used to describe how to get the period more accurate than this using this method.

4.6 Phase wrapping

As mentioned earlier in section 4.5 a lot of stellar lightcurves, like those of binary star systems and RR Lyrae, have some kind of periodicity. To enhance the typical lightcurve of these stars 'phase wrapping' can be used.
The idea of phase wrapping a lightcurve is based on wrapping the lightcurve per period. I.e. the result is a diagram in which the flux is plotted as a function of the phase (similar to the phase of a sine-function). Written in a semi-mathematical way:

$$\text{phase} = \frac{\text{date}}{\text{period} \times \text{ no. of periods to display}} - \text{integer part of latter term} \tag{9}$$

with number of periods to display usually being 1.
Because phase wrapping a lightcurve uses the period of the found variables, the found period of the Lomb periodogram program (see section 4.5) can be used.
From the results in that section it could be concluded that the determined period was not accurate enough. It is possible to improve this by plotting a phase wrapped lightcurve with the found period. By adjusting the period manually (and the number of periods when the found period was due to overtones) a better phase wrapped lightcurve and therefore also a more accurately known period (see figure 10 for an example) is retrieved.

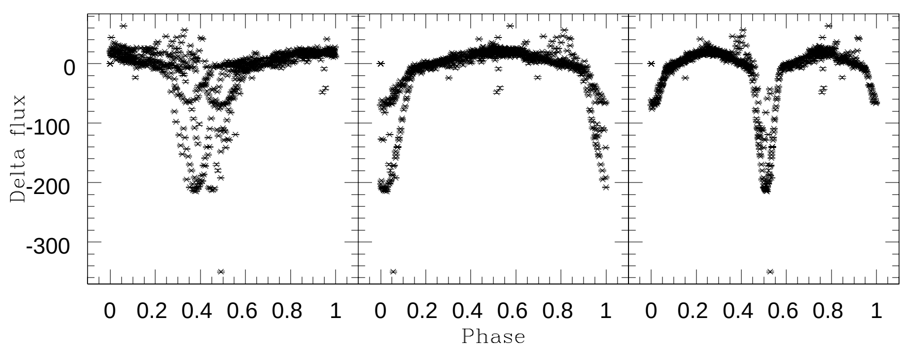


Figure 10: From left to right an example of how phase wrapping the lightcurve helped in improving the period. In the left figure the original period (0.5722709 days) was used as input. A very clear signal was already visible. The middle figure had the right period (0.5755 days), but showed different levels in the dip. It did not seem to be caused by scatter. So in the right image the period was doubled (1.151 days) and the lightcurve of a binary star system was revealed. In figure 9 the periodogram of this variable is given. The doubled period is also present in the periodogram at a frequency of 0.873712 days^{-1} (just right from the peak at 0.714855).

4.7 Counts to magnitude

As can be seen in figure 10 the change in flux is given in a strange unit, which is not a 'standard' measure of the change of flux. This delta flux unit (dfu) can be converted to a relative magnitude change.
The dfu multiplied with the exposure time (also a setting in the template and pipeline scripts) gives the change in flux in counts. Counts and dfu's are related to each other by counts = EXP_TIME $\times$ dfu, with EXP_TIME the exposure time of the images in seconds (for these observations 300). When the number of counts of the star are known, the change in magnitudes can be found using

$$\Delta\text{flux} = -2.5\log_{10}\left(\frac{\text{counts}_*}{\text{count}_* + (\text{dfu}_* \times \text{EXP\_TIME})}\right) \tag{10}$$

with Δflux the change in flux in (V+R) in magnitudes, $count_*$ the counts of the star on the reference image and dfu_* the change in flux of the star in dfu (result from the pipeline).
The counts can also be used to calculate the percentage change in (V+R). This is calculated using the maximum change in dfu (from the highest value to the lowest) multiplied with EXP_TIME. This gives the maximum change in counts. This compared to the counts of the star on the reference image gives the percentage change.

5 Variables

5.1 Discovered variables

Location	x-coordinate in subsection (pix)	y-coordinate in subsection (pix)	period (days)	type	OGLEGC
sec 1 subs 4 2	20.8	296.3	0.3646948	RR-Lyrae(?)	
sec 1 h2 subs 3 1 [c]	178.8	310.8	1.5248424	RR-Lyrae(?)	
sec 2 subs 1 2	124.3	434.8	0.4463	Binary system	253
sec 2 subs 2 2	425.8	249.5	0.3633	RR-Lyrae	
sec 2 h1 subs 4 1 [c]	280.9	289.4	0.4290	Binary system	
sec 3 subs 2 1 (1)	292.9	305.8	0.37884	Binary system	
sec 3 subs 2 1 (2)[b]	78.2	323.1	0.5251	RR-Lyrae	255
sec 3 subs 2 2	485.7	235.1	> 16[d]	?	
sec 3 subs 3 1	435.4	314.5	1.151	Binary system	228
sec 3 subs 4 2	88.7	128.6	0.2971	RR-Lyrae	223
sec 4 subs 3 2 (1)	398.9	74.0	0.139994	RR-Lyrae	
sec 4 subs 3 2 (2)[a]	221.1	73.0	0.300464	Binary system	
sec 4 subs 3 2 (3)[b]	449.4	41.7	9.9	?	

Table 4: Information about the variable stars. Location means where a star can be found on the reference image. Sec `a` (h`b`) subs `c d` (`e`) means that the star can be found in section `a` (only in (half `b`) if given) in subsection `c d` and it is the `e`-th variable star there. The x- and y-coordinates are the coordinates of the star in subsection `c d`. The period is given in days. Type stands for the type of variability that causes the star to fluctuate. OGLEGC means the number of the star in the OGLE general catalog given in Kaluzny et al [Kaluzny et al 1998].

[a]Discovered by Sackett first
[b]Discovered by Weldrake first
[c]Only found in one part of images. See section 4.4 for more details.
[d]See section 5.2 to understand why period > 16 days.

Location	% change in (V+R)[a]	V	I	(V-I)	Remarks
sec 1 subs 4 2	11 ± 4	19.35	19.03	0.32	
sec 1 h2 subs 3 1	20 ± 6	19.35	18.98	0.37	
sec 2 subs 1 2	41 ± 2	16.82	16.29	0.59	$V_{max} = 16.77$, $V_{min} = 17.12$, V-I = 0.57[b]
sec 2 subs 2 2	14 ± 4	19.65	19.19	0.46	
sec 2 h1 subs 4 1	8 ± 2	19.13	18.23	0.90	
sec 3 subs 2 1 (1)	30 ± 3	16.72	16.23	0.49	
sec 3 subs 2 1 (2)	35 ± 5	n/a[b]	n/a[a]	n/a[a]	$V_{mean} = 19.8$, V-I = 0.50[c]
sec 3 subs 2 2	10 ± 3	n/a[b]	n/a[a]	n/a[a]	
sec 3 subs 3 1	43 ± 4	n/a[b]	n/a[a]	n/a[a]	$V_{max} = 15.90$, $V_{min} = 16.30$, V-I = 0.34[c]
sec 3 subs 4 2	33 ± 3	n/a[b]	n/a[a]	n/a[a]	$V_{mean} = 17.6$, V-I = 0.33[c]
sec 4 subs 3 2 (1)	16 ± 3	n/a[b]	n/a[a]	n/a[a]	
sec 4 subs 3 2 (2)	13 ± 3	n/a[b]	n/a[a]	n/a[a]	
sec 4 subs 3 2 (3)	14 ± 3	n/a[b]	n/a[a]	n/a[a]	

Table 5: More information about the variable stars, like the V and I magnitude and the (V-I) color. In the remarks column some extra data about the stars are mentioned.

[a]see section 4.7 for an explanation of this value
[b]n/a: no magnitude data of these stars are available. They are not in the database produced by Doherty [Doherty 2001].
[c][Kaluzny et al 1998]

For each section of the images it took the pipeline, which was the most time consuming part of the DIA method, approximately 3 days to complete (extra time due to preparing the data and going through all

the output was not taken into account in those 3 days). The results of running the pipeline (and all other scripts) are given in Table 4 and 5. In the first Table the location, period and type of the variable are given. In the final column the OGLEGC [Kaluzny et al 1998] number of the star is given if it was crossidentified (see section 6.1 for more details). In the second Table the V and I magnitudes of the stars are given together with the (V-I) color. These magnitude data are taken from her database created by Michelle Doherty [Doherty 2001] (again see section 6.1). When no magnitude data are given it was not in the database. In the final column some extra data of the stars were given when they have been found by Kaluzny et al [Kaluzny et al 1998] too.
The lightcurves of the images are plotted in section 5.2 in figures 11 to 23.

5.2 Lightcurves

In this section the lightcurves of the discovered variables are plotted. The upper area of each figure is the complete lightcurve. On the x-axis the time in days is plotted, on the y-axis the change in (V+R)-magnitude (section 4.7) is plotted. In the figure (and caption) the name of the variable is given, also the corrected period is indicated. The lower area of the figure is the phase wrapped (section 4.6). More information of the variables, like location and magnitude, can be found in Table 4 and 5.
Stars that are also known by an OGLEGC number can be found in Kaluzny [Kaluzny et al 1998].
The variables classified as binary star system were classified by eye. The same applies for the RR Lyrae, though in section 6.2 more information about the identification of the RR Lyrae can be found.
Because no magnitude data of the two long period variables is available, only a suggestion for their type (variable K-giants) can be given.

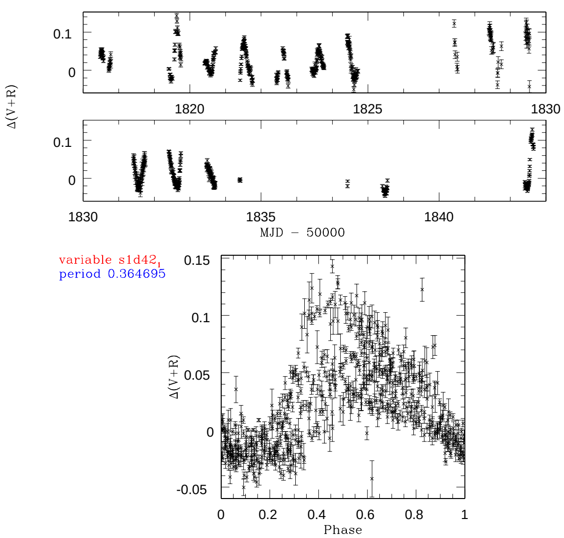


Figure 11: Variable in section 1 subsection 4 2. This is probably an RR Lyrae in the SMC. See section 6.2 for more details.

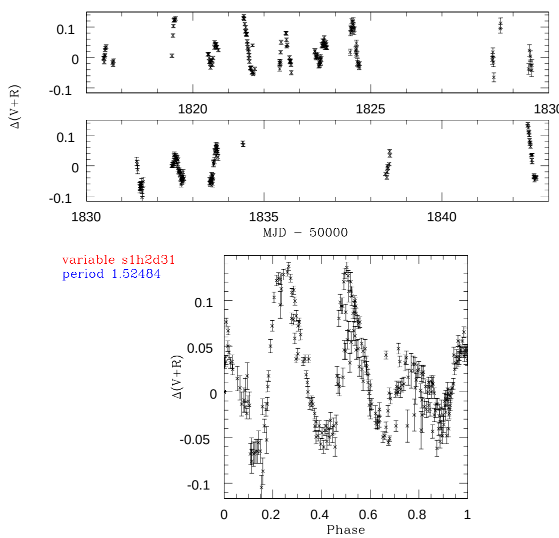


Figure 12: Variable in section 1 half 2 subsection 3 1. This is probably an RR Lyrae in the SMC. The phase-wrapped plot is probably of 4 times the actual period. See section 6.2 for more details.

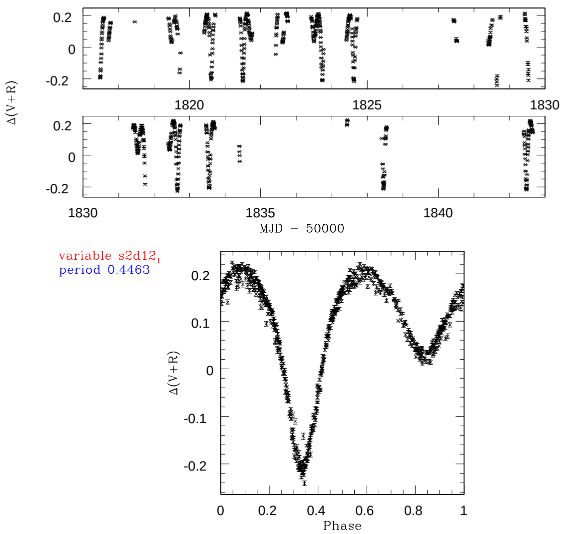


Figure 13: Variable in section 2 subsection 1 2. This is a binary star system. It is also known as OGLEGC 253.

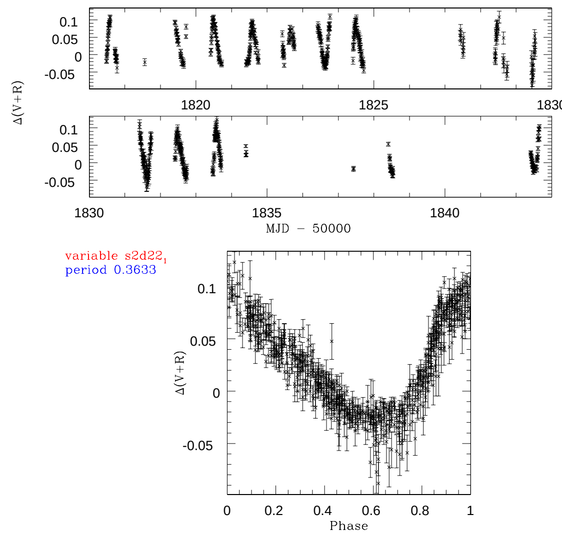


Figure 14: Variable in section 2 subsection 2 2. This is an RR Lyrae in the SMC.

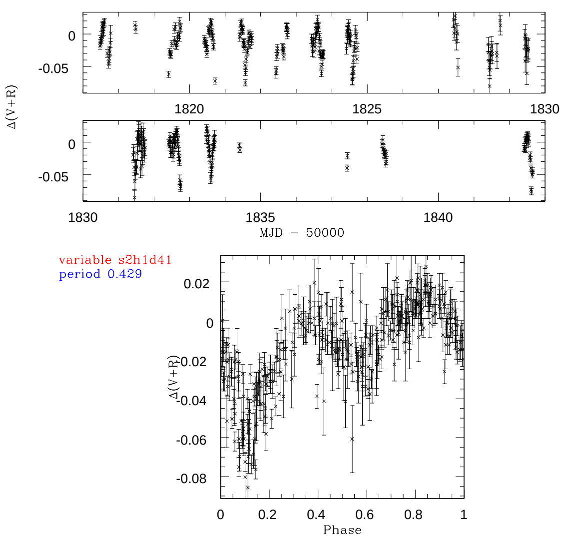


Figure 15: Variable in section 2 half 1 subsection 4 1. Though it is only found in one half, it is a binary star.

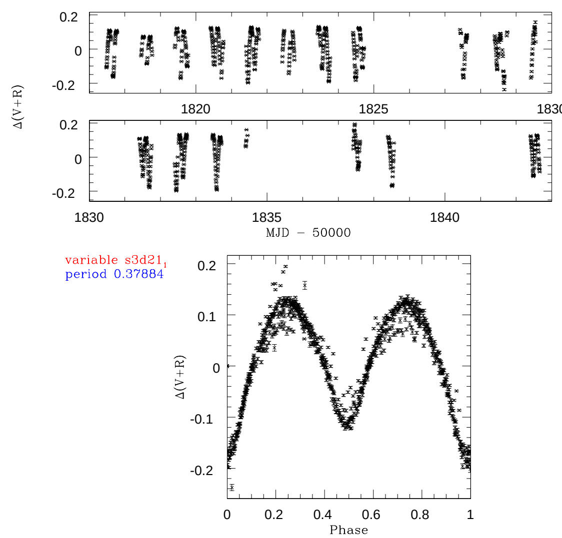


Figure 16: Variable in section 3 subsection 2 1 (1). This is a binary star system.

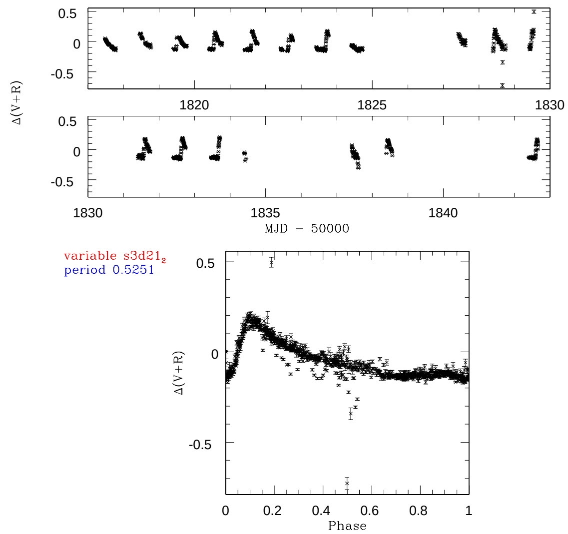


Figure 17: Variable in section 3 subsection 2 1 (2). This is an RR Lyrae found in the SMC. It is also known as OGLEGC 255. It is first found by David Weldrake [Weldrake (private communication)].

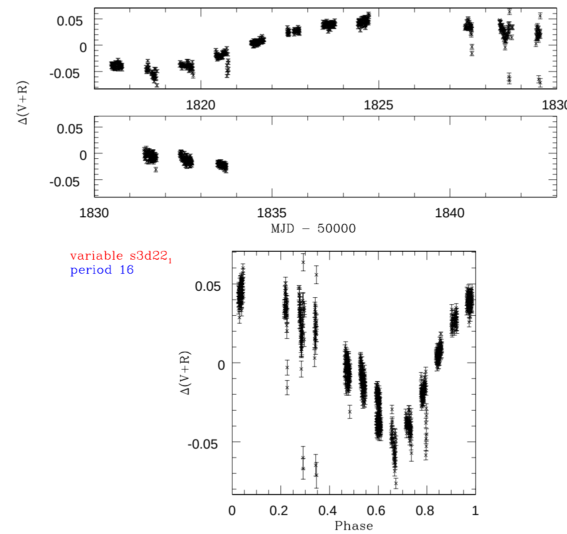


Figure 18: Variable in section 3 subsection 2 2. This is a variable with a period longer than 16 days. A more accurate determination of the period is not possible, because more data are not available. There is a lot of scatter in both plots. For enhancement of the variation the data from date 1836 and later were removed. These data contained too much scatter. It may be a K giant, but no magnitude data on this star are present.

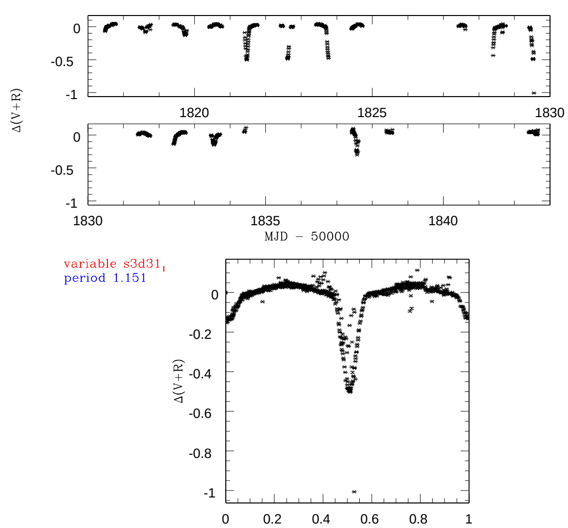


Figure 19: Variable in section 3 subsection 3 1. This is a binary star system, which is also known as OGLEGC 228.

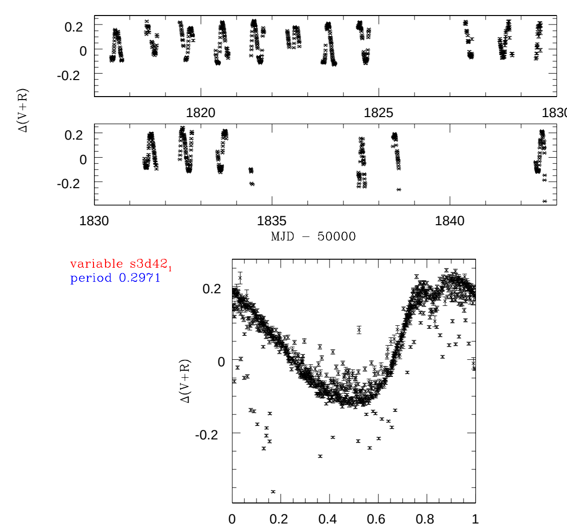


Figure 20: Variable in section 3 subsection 4 2. This is an RR Lyrae in the halo of our Galaxy. It is also know as OGLEGC 223.

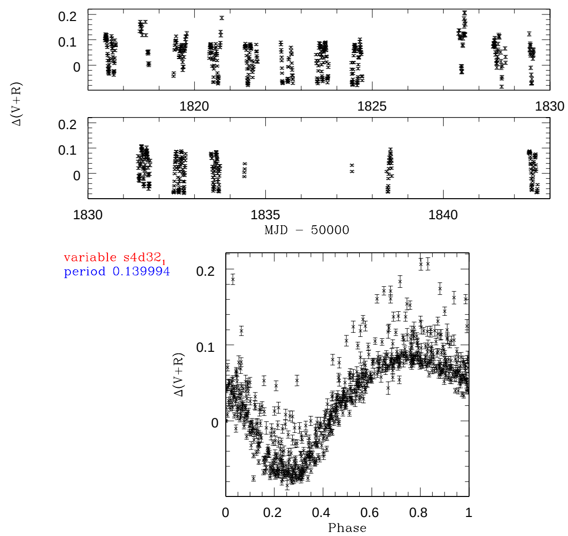


Figure 21: Variable in section 4 subsection 3 2 (1). This is an RR Lyrae in the SMC.

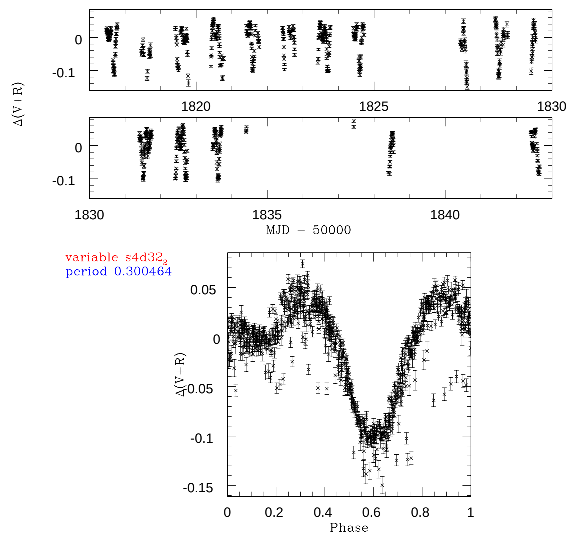


Figure 22: Variable in section 4 subsection 3 2 (2). This is a binary star system. It is first found by Penny Sackett [Sackett (private communication)].

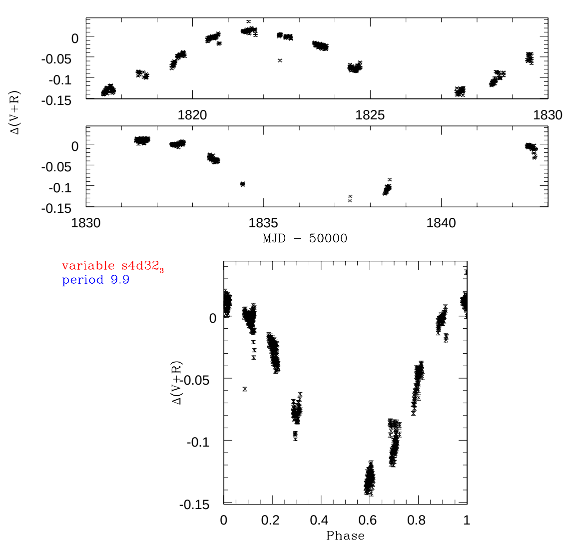
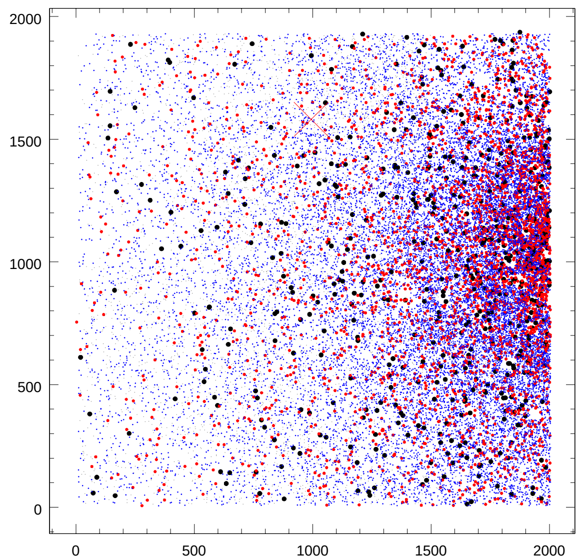


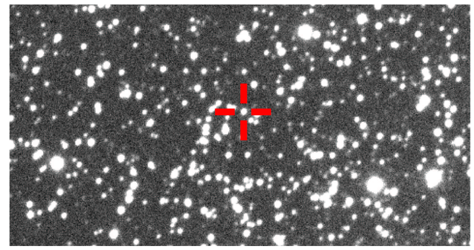
Figure 23: Variable in section 4 subsection 3 2 (3). This is a long period variable. Probably a K-giant, but no magnitude data on this star are present. It is found first by David Weldrake [Weldrake (private communication)].

6 Analysis

6.1 Color-magnitude diagram



(a) Plot of field 104A of [Kaluzny et al 1998]. The big black dots are stars with magnitude > 15, the red dots < 17.5 and > 15, the blue dots < 20 and > 17.5. The big red diagonal cross (though hard to find, it is located at x = 993.95, y = 1581.29) represents OGLEGC214 [Kaluzny et al 1998].



(b) Area around OGLEGC214 (in red cross) on the reference image of this data set.

Figure 24: Cross identification of OGLEGC214 using Kaluzny et al [Kaluzny et al 1998] and the reference image

Having so many stars in a field it was possible with the V- and I-images to determine the V- and I-magnitudes (and therefore also the (V-I) color) of almost 7000 stars in CCD 3. These data were collected by Michelle Doherty[Doherty 2001].
These data plotted in a color-magnitude diagram (CMD) gave a nice view of the location of the found variables in comparison to the zero age main sequence of 47 Tucanae.
For a very rough estimate of the age and metallicity models for evolutionary tracks and isochrones for low- and intermediate-mass stars of Girardi et al [Girardi et al 2000] were used. These models are suitable to model star clusters and galaxies by means of population synthesis. The values in their models were for the Johnson-Cousins UBVRIJHK broadband photometric system, as used for the mentioned V- and I-images.
The values given are absolute magnitudes. To bring these values to the distance of 47 Tucanae the distance modulus $(m - M)_V$ was used. In Hesser et al [Hesser et al 1987] a value of 13.4 for $(m - M)_V$ was given. This corresponds to a distance of approximately 4.8 kpc.
With these data a CMD can be made with therein plotted the isochrones of a cluster. In figure 25 the isochrones of a cluster with an age of 10 to 15 Gyr at a metallicity of Z = 0.004 (blue lines) and with an age of 12 to 20 Gyr at a metallicity of Z = 0.008 (red lines) are plotted. These values were the best fits on the turn-off. Doing this with the uncalibrated CMD resulted in isochrones absolutely not corresponding to the data points. This meant (together with the CMD of Kaluzny et al [Kaluzny et al 1998]) that the data from Doherty [Doherty 2001] were uncalibrated.
The calibration for CCD 3 was done using the V- and I-data from Kaluzny et al [Kaluzny et al 1998]. The only problem was that their stars had to be cross-identified with those on CCD 3. This was done in the most old-fashioned way imaginable: have some printouts of the locations of the stars and some indication for their magnitude and find which stars match.
An example of this is given in figure 24. In the left figure the stars of field 104A are plotted with OGLEGC 214 indicated with a big red cross (on this image hard to find). Together with using SIMBAD [4] and Aladin Previewer[5] to make an optical image of that area around OGLEGC 214, it was possible to find

this star on CCD 3, which is indicated in the right figure of figure 24. From that point it was a matter of finding corresponding stars around OGLEGC 214, looking up the calibrated magnitude of those stars in Kaluzny et al [Kaluzny et al 1998] and the uncalibrated magnitudes in Doherty [Doherty 2001]. Using 12 stars this gave an average difference of $V_{Kal} - V_{Doh} = -2.24 \pm 0.03$, $I_{Kal} - I_{Doh} = -2.87 \pm 0.04$ and $(V-I)_{Kal} - (V-I)_{Doh} = 0.63 \pm 0.03$.
Using these values on the data of Doherty [Doherty 2001] they give a calibrated CMD for CCD 3 (figure 25).

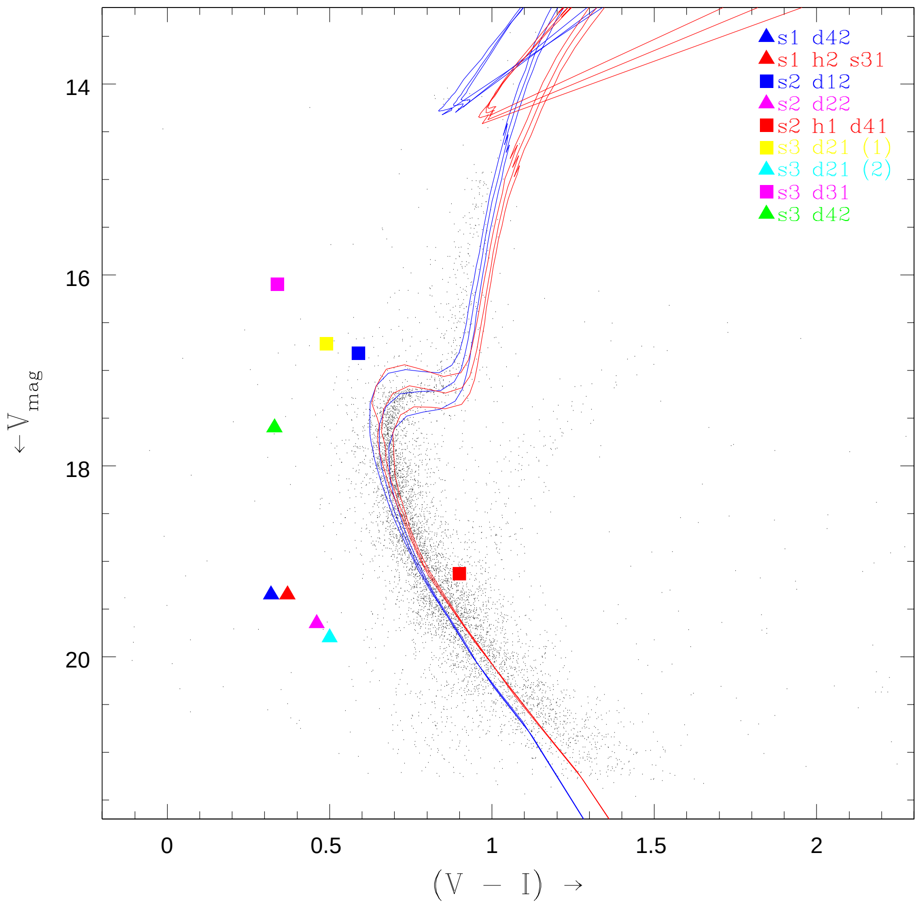


Figure 25: CMD of CCD 3. It has indicated the found variable stars. Triangles are RR Lyrae stars and squares are binary systems. The blue lines are the isochrones with an age of 10 to 15 Gyr at a metallicity of Z = 0.004. The red lines are the isochrones with an age of 12 to 20 Gyr at a metallicity of Z = 0.008.

6.2 Period-luminosity relation

For RR Lyrae stars it is known that RR Lyrae have the same luminosity independent of its period [Zeilik et al 1998]. To test this for the found variables the average magnitude of all 5 (of which some suspected to be) RR Lyrae was taken ($V_{av} = 19.15$ and $I_{av} = 18.75$). As immediately can be seen in figure 26, where these values are plotted as a dotted line, the RR Lyrae in section 3 subsection 4 2 deviates a lot. Together with the star being about 3 magnitudes brighter than the other RR Lyrae this suggests that this star is not a member of 47 Tucanae. In Table 4 it is clear that this star was already known by Kaluzny et al [Kaluzny et al 1998]. They state that the star is an RR Lyrae in the halo of our Galaxy.
The 4 remaining stars have average magnitudes of $V_{av} = 19.54$ and $I_{av} = 18.75$. Because RR Lyrae stars usually have an absolute V magnitude around 0.5 to 1 [Zeilik et al 1998] these stars are also not members of 47 Tucanae, when a distance modulus of 13.4 is applied. The distance modulus of the Small Magellanic Cloud (SMC), which is very close to 47 Tucanae on the sky, is determined by Udalski et al [Udalski et al 1998] to be approximately 18.08. This is approximately 0.4 magnitudes lower than the generally accepted values. Using this knowledge this will likely place the 4 RR Lyrae in the SMC.
Another reason to assume that these 4 stars are not members of 47 Tucanae is that RR Lyrae stars are brighter and redder than main sequence stars [Zeilik et al 1998].

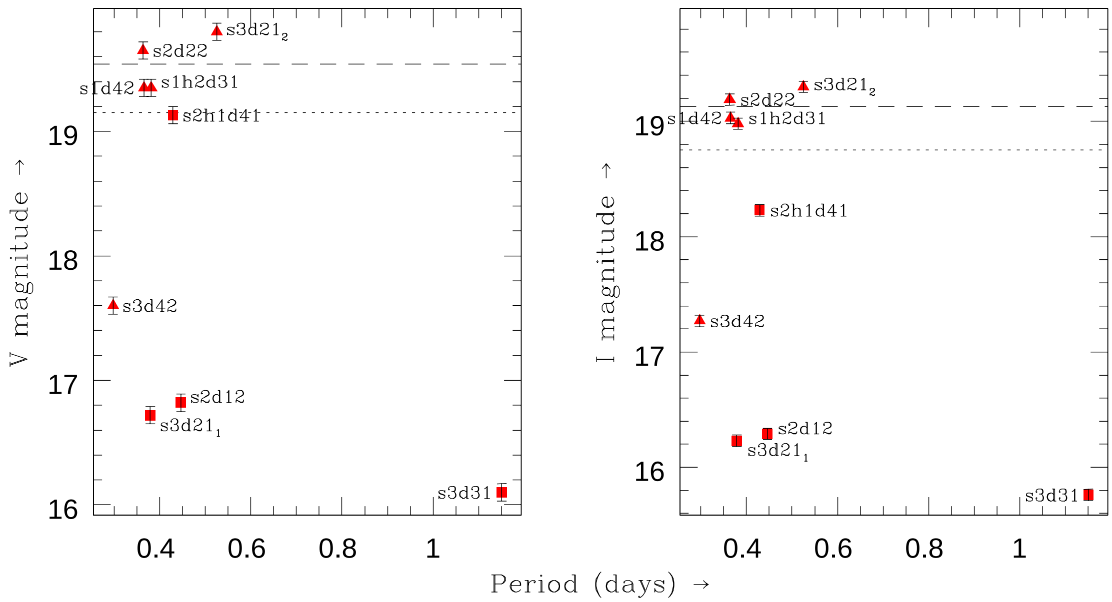


Figure 26: Period versus luminosity. Red triangles are RR Lyrae variables, red squares are binary systems. The dashed line is the average magnitude of the RR Lyrae belonging to 47 Tucanae, the dotted line is the average magnitude of all RR Lyrae.

As it was very certain for the second variable in section 3 subsection 2 1 (also known as OGLEGC 255) to be an RR Lyrae [Kaluzny et al 1998], this will make the variables in section 1 subsection 4 2 and section 1 half 2 subsection 3 1 likely to be RR Lyrae. The latter star is plotted in figure 26 using $\frac{1}{4}$ of its period given in Table 4, because the longer period was used to show the general trend in variation of this variable's lightcurve.

6.3 CMD-CCD relation

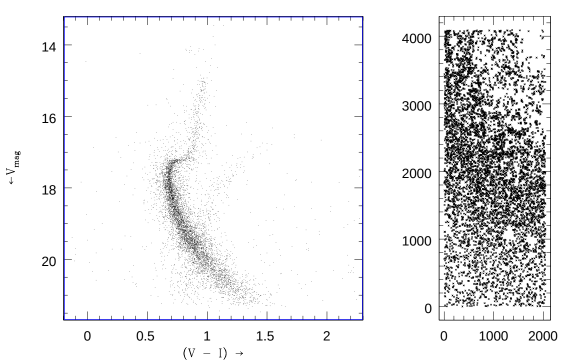


Figure 27: Distribution of all the stars

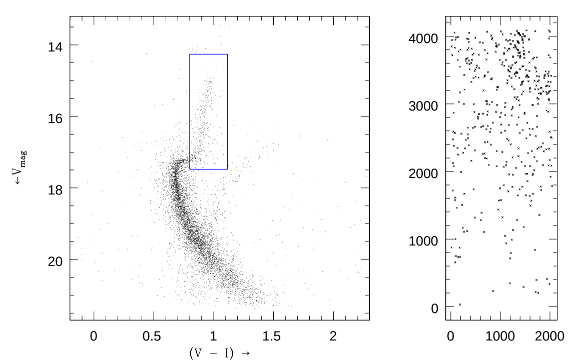


Figure 28: Red Giant Branch

It might be possible to see how the different type of stars like main sequence stars, red giants stars etc are distributed over the sky. To do this the location on the CCD of stars in those specific areas of the CMD can be plotted.

In figure 27 the locations of all stars in the CMD are plotted. Several things are noticeable.

Firstly the core is clearly visible. There are no data on this area of the CCD, because the core is so crowded and bright that the CCD was saturated.

Secondly some other white spots are found. These are due to foreground stars saturation the CCD at those locations.

Thirdly there is a strong gradient visible in the CCD. From the top right corner to the lower left corner the amount of stars per square pixel decreases. If there would not be saturation effects, it would be a lot more clear. This gradient is also visible on the image on the cover of this report.

The first specific area of the CMD is the Red Giant Branch (RGB). In figure 28 this area is indicated by a blue box. The location of the stars on the CCD is next to it. Clearly visible is the same gradient as talked about before. That there are more red giants in the center of the cluster (~ 0.1 red giant per star) than in the outskirts (~ 0.03 red giant per star) could be an indication that more massive stars were formed in the center of the cluster. The more massive stars evolve faster than lighter stars. Therefore there could be more red giants in the core than in the outskirts.

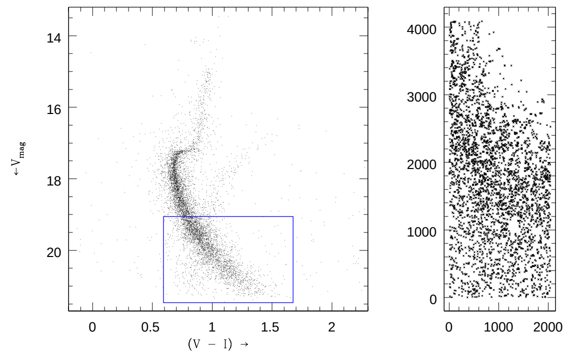


Figure 29: Lower area of the Main Sequence

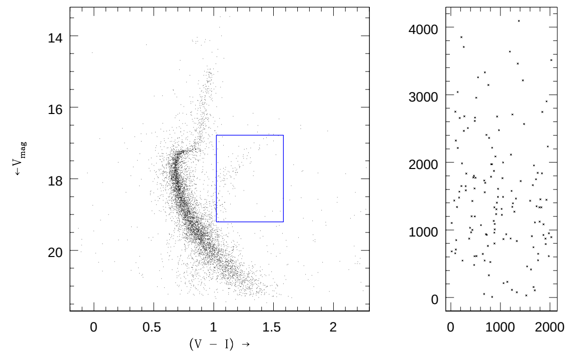


Figure 30: Small Magellanic Cloud

The next area on the CMD that is interesting is the main sequence (MS). There is some contamination due to the presence of the SMC. The RGB of the SMC was detected to be running through the MS of 47 Tucanae on the CMD. But the majority of the stars plotted in figure 29 are stars of the MS of 47 Tucanae. The stars of the RGB of the SMC are boxed in figure 30.

The stars of the MS are clearly less present near the core and more present in the outskirts, but decreasing in density when going further away from the core as the overall density of stars decreases outwards.

The final group on the CMD are the stars belonging to the RGB of the SMC. These stars are (almost) homogeneously distributed over the CCD. This is logical because at the sky the SMC is bigger than 47 Tucanae and the stars of the SMC that are in the neighborhood of 47 Tucanae are on the outskirts of the SMC, where a more homogeneously distribution of stars can be expected than in the center of the SMC.

7 Discussion

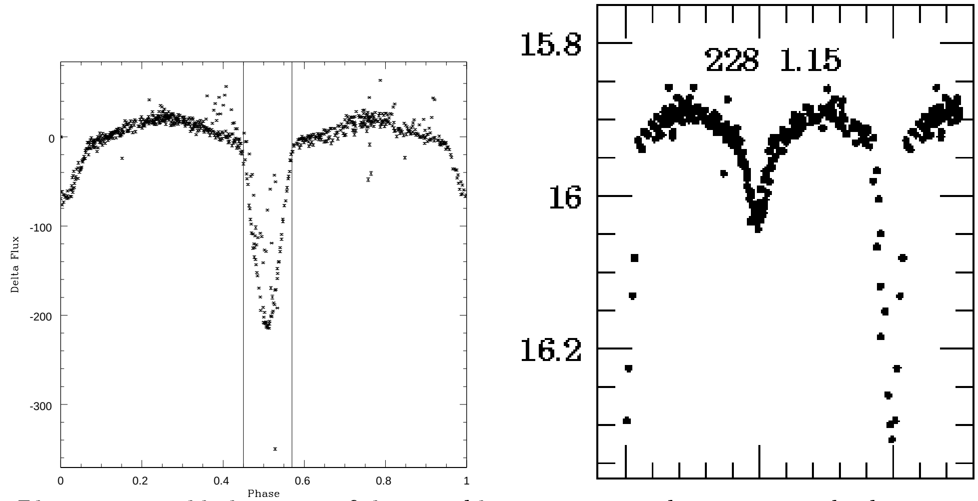


Figure 31: Phase-wrapped lightcurves of the variable in section 3 subsection 3 1, also known as OGLEGC 228 [Kaluzny et al 1998]. In the left figure the time span in which the most variation was, is indicated. The right figure was taken from Kaluzny et al [Kaluzny et al 1998].

7.1 Magnitudes

The V and I data taken by Doherty [Doherty 2001] have normal errors in the values: less then 0.01 magnitudes for stars brighter than a magnitude of approximately 19.5 and up to 0.5 for stars as dim as approximately 21.5 for both V and I. This means for the color (V-I) that the error will be twice the size of the error of the single magnitude.
As the V and I magnitudes were not taken simultaneously (they were taken 13 minutes apart), an error is introduced that is due to the variation of the star caused by its variability. To illustrate this, the variable in section 3 subsection 3 1 was used, because the magnitude data are known by Kaluzny et al [Kaluzny et al 1998], see figure 31.
In this variable the greatest change in flux happened twice from 0.45 of the phase until 0.57. So in half of this time ($\frac{0.57-0.45}{2} = 0.06$ phase) the maximum change per unit phase was achieved (from 16.33 to 15.93 = 0.40 magnitude). This meant $\frac{0.40 \text{ magn}}{0.06 \text{ phase}} = \frac{0.40}{0.06 \cdot 1.151 \cdot 24 \cdot 60} = 4.0\ 10^{-3} \frac{\text{magn}}{\text{min}} = \frac{0.05 \text{ magn}}{13 \text{ min}}$. So there is an extra uncertainty for this variable of approximately 0.05 magnitudes.
This to show that the variable stars on the CMD in figure 25 have a rather large error in comparison to the other stars.

7.2 Scattering in lightcurves

Some of the variables have a lot of scatter in their lightcurves. This is well visible in phase wrapped lightcurves. Very good examples of this are the variables in section 4 subsection 3 2 the first and second variable.
When the points of images with values above a certain seeing are left out of the phase wrapped plot, the scatter is reduced very much. This concerns all the variables except the ones in section 1 subsection 4 2, section 1 half 2 subsection 3 1 and section 2 half 1 subsection 4 1. Coincidently these are 3 of the 4 with the lowest range in the delta flux. There might be a relation between the seeing and the scattering. Also 2 of these variables are found in only 1 half of the data. Follow up work may be done about this (see section 9) to confirm the existence of these 2 variables.
The seeing level at which the other lightcurves cleaned up much is at 8 pixels (= 3.0"). But taking out the images with seeing above 6 pixels (= 2.3") clears the images even more in most cases.
Definitely seeing is of an important influence on the DIA method.

7.3 Sensitivity

In Table 5 the percentage change over a complete period of the variable stars is given. The variable star that had the least change was the variable in section 2 half 1 subsection 4 1 ($\sim$ 8%). If the change of only the second dip of this variable would be measured it would be $\sim$ 2%. This would be in the region of being able to detect planets.
If this was calculated the other way around, for a star of approximately 23000 counts (V roughly estimated lies around 20 mags) a 2% dip would require 1.5 dfu. For comparison the total change of the variable in section 2 half 1 subsection 4 1 is around 7.3 dfu and the second dip is around 2 dfu.
Also the errorbars in most of the lightcurves have a size of 0.5 dfu. So a 2% dip ($\sim$ would be 1.5 dfu) would be a 3σ detection.

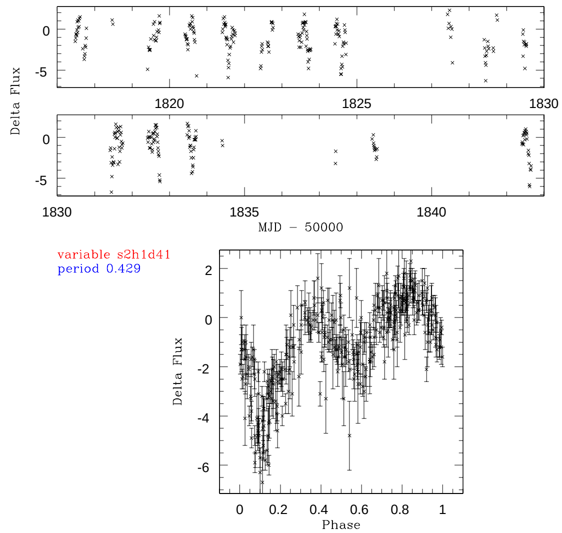


Figure 32: Variable in section 2 half 1 subsection 4 1. The y-axis is in dfu units.

8 Conclusions

As already mentioned in section 2.1 it became clear that the techniques used for this project would not produce a planet detection.
In section 7.3 is mentioned that the observed images in combination with the DIA method are sensitive enough to detect planets. That no planetary transits are found, is due to using the periodogram-method to find periodicity. This method might be replaced by/used next to a matched filter technique.
Despite the conclusion that the used techniques didn't provide planetary transits, 13 variable stars are found of which 4 are already known and 9 are newly found variable stars. Of these 9 variables 4 are RR Lyrae stars of which at least 3 can be placed in the SMC, 3 are binary star systems and 2 long period variables are found, which are variable K-giants.
It is also worth mentioning that testing the DIA method has been successfully, because variable stars are found.

9 Future Work

The first thing that should be done is to apply the DIA method to all the 7 other CCDs.
Other future work on this project is not so far in the future as can normally be expected. At this moment David Weldrake finished doing observations on 47 Tucanae last (European) autumn. The duration of his run was approximately 4 weeks and resulted in approximately 1200 images all with a seeing of 2.8" or less. The seeing is one of the recommendations that emerges from this project: low seeing will probably result in less scatter in the lightcurves. Maybe there is a way in the DIA method to decrease the influence of seeing on the results.
Another suggestion that comes from this project is to get more out of the DIA method. There can be played with some settings to optimize the process. For this project there was too little time to do this.
To find planetary transits a matched filter should be used instead of/next to the periodogram-method. The matched filter will specifically be looking for these kind of transits, while the periodogram only looks for some kind of periodicity in lightcurves.
Crowding is a reason to look at an other area of the sky with also a lot of stars, but with a less dense population of stars. This density should be so low that in the past the gravitational influences of stars on each other was much smaller than in 47 Tucanae must have been.
Also other methods, like nulling interferometry, should be used.

10 Acknowledgements

The following people and institutes have to be thanked for their support:

- Penny Sackett as my supervisor and for all the help during this project,
- Terry Bridges as observer and as my supervisor in Australia and for all his help during the project, also for the color-magnitude data,
- Ken Freeman for observing and providing information,
- David Weldrake for checking my results as a part of his thesis and again for his hints and tips on this project,
- Michelle Doherty (via Terry Bridges) for creating a database with V- and I-magnitudes of stars in 47 Tucanae and allowing the use of it,
- system management (Eite Tiesinga in Groningen and Helen Davies and Katrina Tapia-Sealey in Australia) for their support in soft- and hardware,
- Chris Ormel for suggesting Girardi et al 2000,
- Marco Beijersbergen for helping me on the composite images,
- the Anglo-Australian Observatory for providing a place to work,
- the Australian Telescope National Facility (ATNF) for providing the roof in Australia.

This project has been made possible by financial support of

- the Anglo Australian Observatory (AAO),
- the Opleidingsinstituut Sterrenkunde in Groningen,
- the Groninger Universiteits Fonds (GUF),
- the Kapteynfonds.

And not to forget of course on both the private and financial support:

- my parents.

References

[Bridges (private communication)] Bridges, private communication about the image reduction process

[Doherty 2001] Doherty, M., 2001, 47 Tucanae - A colour magnitude diagram (private communication via Terry Bridges)

[Eckart et al 1997] Eckart, A., Genzel, R., 1997, MNRAS, **284** 576-598, Stellar proper motions in the central 0.1 pc of the Galaxy

[Freeman (private communication)] Freeman, private communication about filters.

[Gilliland et al 2000] Gilliland et al, 2000, ApJ, Vol 545 Issue 1, L47-L51, A lack of planets in 47 Tucanae from an HST search

[Girardi et al 2000] Girardi et al, 2000, A&AS, **141** 371-383, Evolutionary tracks and isochrones for low- and intermediate-mass stars: From 0.15 to 7 $M_{\odot}$, and from $Z = 0.0004$ to 0.003

[Hesser et al 1987] Hesser et al, 1987, PASP, **99** 739-808, A CCD color-magnitude study of 47 Tucanae

[Howell 2000] Howell, S.B., 2000, 53, Handbook of CCD astronomy

[Illingworth 2000] editor Illingworth, V., 2000, p 223, Collins Dictionary Astronomy Second Edition

[Kaluzny et al 1998] Kaluzny et al, 1998, A&AS, **128** 19-28, The optical gravitational lensing experiment. Variable stars in globular clusters. IV Fields 104A-E in 47 Tucanae

[Press et al 1992] Press, W.H., et al, 1992, Numerical Recipes in C, The Art of Scientific Computing, Second Edition, chapter 13.8, 575-584

[Sackett 1998] Sackett, P.D., 1998, Planets outside the Solar System: Theory and observations, Eds. J.M. Mariotti and D. Alloin

[Sackett (private communication)] Sackett, private communication about variable detection

[Udalski et al 1998] Udalski, A., et al, 1998 AcA, **48** 1-17, Optical Gravitational Lensing Experiment. Distance to the Magellanic Clouds with the Red Clump stars: are the Magellanic Clouds 15% closer than generally accepted?

[Weldrake (private communication)] Weldrake, private communication about variable detection

[Woźniak 2000] Woźniak, P.R., 2000, AcA, vol **50** 421-450, Difference Image Analysis of the OGLE-II Bulge Data. I. The method

[Zeilik et al 1998] Zeilik, Gregory, 1998, Introductory Astronomy and Astrophysics (fourth edition), chapter 18-2, 353-356

[1] Internet http://unisci.com/stories/20021/0328022.htm

[2] Internet http://www.strw.leidenuniv.nl/annrep00/node64.html

[3] Internet http://www.aao.gov.au/wfi/wfi_perform.html

[4] Internet http://simbad.u-strasbg.fr/sim-fid.pl

[5] Internet http://aladin.u-strasbg.fr/AladinPreview

A Source code of masking script

Written by Eduard Westra.

```
#include <iostream>
#include <string>
#include <iomanip>
#include <fstream>
#include <sstream>

using namespace std;

int main() {
  const double nx = 2044., ny = 1024.;
  double x, y, val;
  string basename = "maskeer";
  int linecounter = 0, filecounter = 0;
  ofstream of;
  string filename, nextfilename;
  double x1 = 5., x2 = 10.;
  double y1 = 8., y2 = 16., 13.;

  while(cin >> x >> y >> val) {
    if (val == 1.) {
      if((linecounter % 1000) == 0) {
        filecounter++;
        ostringstream ostr, ostr1;
        ostr << filecounter;
        ostr1 << filecounter+1;
        string extension = basename;
        extension.append(ostr.str());
        filename = extension.append(".cl");
        extension = "cl < ";
        extension.append(basename);
        extension.append(ostr1.str());
        nextfilename = extension.append(".cl");
        of.open(filename.c_str());
      }
      of << "imreplace mask.fits[" << ((x-x1  < 1) ? 1 : x-x1)  << ":"
           << ((x+x1  > nx) ? nx : x+x1)  << ","
           << ((y+(y1+1.)  > ny) ? ny : y+(y1+1.))  << ":"
           << ((y+y2 > ny) ? ny : y+y2) << "]" << " 0\n";// << endl;
      of << "imreplace mask.fits[" << ((x-x2 < 1) ? 1 : x-x2) << ":"
           << ((x+x2 > nx) ? nx : x+x2) << ","
           << ((y-y1  < 1)  ? 1  : y-y1)  << ":"
           << ((y+y1  > ny) ? ny : y+y1)  << "]" << " 0\n";// << endl;
      of << "imreplace mask.fits[" << ((x-x1  < 1) ? 1 : x-x1) << ":"
           << ((x+x1  > nx) ? nx : x+x1)  << ","
           << ((y-y3 < 1)  ? 1  : y-y3) << ":"
           << ((y-(y1+1.)  < 1)  ? 1  : y-(y1+1.))  << "]" << " 0\n";// << endl;
      linecounter++;
      if((linecounter % 1000) == 0) {
        of << nextfilename << endl;
        of.close();
      }
    }
  }
  return 0;
}
```

B Template script

Written by Przemyslaw Woźniak, modified by Eduard Westra.

```
#!/bin/tcsh -f

set DATA_DIR = "/ari2/users/westra/final_sec1/DataDir"
set WORK_DIR = "/ari2/users/westra/final_sec1/WorkDir"
set BIN_DIR  = "nice +1 /ari2/users/westra/final_sec1/bin"

set FIELD = "47Tuc_ccd3_sec1"
set refim = "lccd0817_ccd3_sec1"

cd $WORK_DIR

###############################################################################
awk '{printf"%scr.fts\n",$1}' tplimages.txt >! tplcrossimages.txt
awk '{printf"%sr.fts\n",$1}' tplimages.txt >! stackimages.txt
awk '{printf"s_%sr.fts\n",$1}' images.txt >! difimages.txt
###############################################################################

set nx  = 512
set ny  = 512
set nx0 = 2048
set ny0 = 1045

set edge  =  5
set marg  =  0
set kerhw =  7

@ xload = $nx0 / $nx
@ yload = $ny0 / $ny

#######################################
set images = ` cat images.txt `
set tplim  = ` cat tplimages.txt `

foreach tpl ( $tplim )
    echo $tpl
    ${BIN_DIR}/super_cutfits ${DATA_DIR}/${tpl}.fts ${tpl}cr.fts 512 1535 1 1024
end

${BIN_DIR}/super_cutfits ${DATA_DIR}/${refim}.fts ${refim}cr.fts 512 1535 1 1024
${BIN_DIR}/cross ${WORK_DIR}/cross.par "${refim}cr.fts" -f ${WORK_DIR}/tplcrossimages.txt >!
                 ${WORK_DIR}/tplshifts.txt
#######################################

echo >! ${FIELD}.db
echo >! ${FIELD}.cat
echo >! trimfiles.txt
echo >! var_${FIELD}.coo

@ nxw = $nx + 2 * $marg
@ nyw = $ny + 2 * $marg

set tplimages = ` cat tplimages.txt `
set images2   = ` cat tplimages.txt | fgrep -v ${refim} `

set ix = 1
set iy = 1

while( $iy <= $yload )
```

```
while( $ix <= $xload )
   set prefix = "${ix}_${iy}"

   mkdir dif_${prefix}
   cd dif_${prefix}
   set dir = ` pwd `
   echo $dir

   @ xl = 1 + $nx * ($ix - 1) - $marg
   @ xu =     $nx *  $ix      + $marg

   @ yl = 1 + $ny * ($iy - 1) - $marg
   @ yu =     $ny *  $iy      + $marg

   ${BIN_DIR}/super_cutfits ${WORK_DIR}/ONE_ushort.fts blank_ushort.fts $xl $xu $yl $yu

   foreach im ( $tplimages )
      set arg = ` fgrep ${im} ${WORK_DIR}/tplshifts.txt `

      @ n11 = $xl - $arg[2]
      @ n12 = $xu - $arg[2]
      @ n21 = $yl - $arg[3]
      @ n22 = $yu - $arg[3]

      echo ${im}: $n11 $n12 $n21 $n22

      ${BIN_DIR}/super_cutfits ${DATA_DIR}/${im}.fts ${im}c.fts $n11 $n12 $n21 $n22

      ${BIN_DIR}/sfind ${WORK_DIR}/sfind.par ${im}c.fts ${im}c.coo
   end

   ${BIN_DIR}/ushort2float ${refim}c.fts ${refim}r.fts -min 1.0

   echo xymatch:
   foreach im ( $images2 )
      echo $im
      set failed = 0
      set nstars = ` wc ${im}c.coo `

      if( $nstars[1] < 10  ||  $status != 0 ) then
         set failed = 1
         set pname = "sfind or wc"
         goto "FAKE"
      endif

      set corr = `${BIN_DIR}/xymatch ${WORK_DIR}/xymatch.par ${refim}c.coo ${im}c.coo ${im}c.match`

      if( $status != 0 ) then
         set failed = 1
         set pname = "xymatch"
         goto "FAKE"
      endif

      echo shifts: $corr

      #################################
      ##### correct if large shift #####
      #################################
      if( $corr[1]>$edge || $corr[2]>$edge || \
          $corr[1]<-$edge || $corr[2]<-$edge ) then
         set arg = ` fgrep ${im} ${WORK_DIR}/tplshifts.txt `
```

```
            @ n11 = $xl - $arg[2] + $corr[1]
            @ n12 = $xu - $arg[2] + $corr[1]
            @ n21 = $yl - $arg[3] + $corr[2]
            @ n22 = $yu - $arg[3] + $corr[2]

            echo nn: $n11 $n12 $n21 $n22

            ${BIN_DIR}/super_cutfits ${DATA_DIR}/${im}.fts ${im}c.fts $n11 $n12 $n21 $n22
            ${BIN_DIR}/sfind ${WORK_DIR}/sfind.par ${im}c.fts ${im}c.coo

            set nstars = ` wc ${im}c.coo `

            if( $nstars[1] < 10  ||  $status != 0 ) then
               set failed = 1
               set pname = "sfind or wc"
               goto "FAKE"
            endif

            echo correction:
            ${BIN_DIR}/xymatch ${WORK_DIR}/xymatch.par ${refim}c.coo ${im}c.coo ${im}c.match

            if( $status != 0 ) then
               set failed = 1
               set pname = "xymatch"
               goto "FAKE"
            endif
         endif
         ##################################
         ##################################

         ${BIN_DIR}/xygrid ${WORK_DIR}/xygrid.par ${im}c.match ${im}c.coeff

         if( $status != 0 ) then
            set failed = 1
            set pname = "xygrid"
            goto "FAKE"
         endif

         ${BIN_DIR}/resample2 ${WORK_DIR}/resample2.par ${im}c.coeff ${im}c.fts ${im}r.fts

         if( $status != 0 ) then
            set failed = 1
            set pname = "resample2"
            goto "FAKE"
         endif

FAKE:
         @ n11 = -$nxw
         @ n12 = -1
         @ n21 = -$nyw
         @ n22 = -1

         if( $failed != 0 ) then
            echo  $pname failed, preparing fake image ${im}r.fts
            ${BIN_DIR}/super_cutfits ${refim}r.fts ${im}r.fts $n11 $n12 $n21 $n22
         endif

      end

      ######
      ${BIN_DIR}/mstack ${WORK_DIR}/mstack.par ${WORK_DIR}/stackimages.txt ref_${prefix}.fts
      ######
```

```
      @ xl = 1   + $marg
      @ xu = $nx + $marg

      @ yl = 1   + $marg
      @ yu = $ny + $marg

      ${BIN_DIR}/super_cutfits ref_${prefix}.fts trim_${prefix}.fts $xl $xu $yl $yu

      echo trim_${prefix}.fts >> ${WORK_DIR}/trimfiles.txt

      ${BIN_DIR}/getpsf ${WORK_DIR}/getpsf.par ref_${prefix}.fts psf_${prefix}.bin

      if( $status != 0 ) then
         echo: getpsf failed for ref_${prefix}.fts
      endif

      if( $status == 0 ) then
         echo  "completed section [${ix},${iy}]"
      endif

      cp trim_*.fts  ..
      cd ..

      @ ix += 1
   end

   @ ix  = 1
   @ iy += 1
end

######
${BIN_DIR}/template ${WORK_DIR}/trimfiles.txt ${WORK_DIR}/tpl_${FIELD}.fts $xload $yload
######

cd ${WORK_DIR}

\rm trim_*.fts
\rm *cr.fts

cd ..

exit 0
```

C Pipeline script

Written by Przemyslaw Woźniak, modified by Eduard Westra.

```
#### Begin Pipeline ####
#!/bin/tcsh -f

echo set_basic_variables
source set_basic_variables 1              ### Make log-file + first entry         ###
cd ${PROG_DIR}

echo calculation
source calculation
cd ${PROG_DIR}

echo prepare_txt
source prepare_txt                        ### Make global txt files               ###
cd ${PROG_DIR}

echo global_variables
source global_variables                   ### Setup unchangeable global variables ###
cd ${PROG_DIR}

echo manipulations
source manipulations                      ### Certain manipulation necessary      ###
cd ${PROG_DIR}

echo section_cross
source section_cross                      ### Running the cross-binary            ###
cd ${PROG_DIR}

############################
###### BEGIN BIG LOOP ######
############################
set ihalf = 1

while ( $ihalf <= 2 )
    if ( ${ihalf} == 1 ) then
      set part = "first"
    else
      set part = "second"
    endif

    cd ${WORK_DIR}

    set ix = 1
    set iy = 1

    set images1 = ‘ cat images_half_${ihalf}.txt ‘
    source ${PROG_DIR}/preparation_big_loop           ### Making 2 files and untarring difs   ###
    while( $iy <= $yload )
        while( $ix <= $xload )
            source ${PROG_DIR}/create_image2subtract  ### Prepare file which are being subt.  ###
            source ${PROG_DIR}/subtraction_photometry ### Subtract and do photometry          ###

            @ ix += 1
            set date = ‘ date ‘
            echo $date >> ${LOG_DIR}/${LOG_FILE}
            echo "Finished loop of ${prefix} of ${part}" >> ${LOG_DIR}/${LOG_FILE}
            echo "" >> ${LOG_DIR}/${LOG_FILE}
        end
        @ ix  = 1
```

```
        @ iy += 1
    end
    set date = ` date `
    echo $date >> ${LOG_DIR}/${LOG_FILE}
    echo "Finished ${part} of big loop" >> ${LOG_DIR}/${LOG_FILE}
    echo "" >> ${LOG_DIR}/${LOG_FILE}
    @ ihalf += 1
end
##########################
###### END BIG LOOP ######
##########################

cd ${WORK_DIR}

set date = ` date `
echo $date >> ${LOG_DIR}/${LOG_FILE}
echo "Finished" >> ${LOG_DIR}/${LOG_FILE}

cd ${PROGRAM_DIR}

echo "Do not forget to run getvariables and restant_subtraction_photometry"

exit 0
#### End Pipeline ####

#### Begin set_basic_variables ####
#!/bin/tcsh -f

set date = ` date `
###########################################
### SETTING UP THE MOST BASIC VARIABLES ###
###########################################
set PROGRAM_DIR = "/ari2/users/westra/final_sec1" ### WITHOUT END SLASH ###
set PROG_DIR = "${PROGRAM_DIR}"
set DATA_DIR = "${PROGRAM_DIR}/DataDir"
set WORK_DIR = "${PROGRAM_DIR}/WorkDir"
set BIN_DIR  = "nice +10 ${PROGRAM_DIR}/bin"
set TAR_FILE = "/ari2/users/westra/runs/tpldifs_ccd3_sec1_10062002.tar"
set LOG_DIR  = "${PROGRAM_DIR}/WorkDir/Log"

set FIELD = "47Tuc_ccd3_sec1"
set refim = "lccd0817_ccd3_sec1"

set LOG_FILE = "log_${date[3]}_${date[2]}_${date[6]}_${date[4]}"
#################
### END SETUP ###
#################

if ( $1 ) then
    echo >! ${LOG_DIR}/${LOG_FILE}
    echo $date >> ${LOG_DIR}/${LOG_FILE}
    echo "Started the Pipeline" >> ${LOG_DIR}/${LOG_FILE}
    echo "" >> ${LOG_DIR}/${LOG_FILE}
else
    echo $date >> ${LOG_DIR}/${LOG_FILE}
    echo "Begin of ${2}" >> ${LOG_DIR}/${LOG_FILE}
    echo "" >> ${LOG_DIR}/${LOG_FILE}
endif

exit 0
#### End set_basic_variables ####
```

```
#### Begin calculation ####
#!/bin/tcsh -f

source set_basic_variables 0 calculation

cd ${WORK_DIR}

###########################################
### CALCULATION OF AMOUNT OF IMAGES ETC ###
###########################################
set images = ` cat images.txt `

set ihalf = 1
while ( $ihalf <= 2 )
    echo >! images_half_${ihalf}.txt
    echo >! mjd_half_${ihalf}.txt
    if !( -e half_${ihalf} ) then
        mkdir half_${ihalf}
    endif

    @ ihalf += 1
end
############################
### FINISHED CALCULATION ###
############################

exit 0
#### End calculation ####

### Begin prepare_txt ####
#!/bin/tcsh -f

cd $WORK_DIR

################################
### PREPARE GLOBAL TXT FILES ###
################################

### images is set in calulation ###
set noofimages = 0
foreach im ( $images )
   @ noofimages += 1
end

if ( -e images_half_1.txt ) then
  rm images_half_1.txt
endif
if ( -e images_half_2.txt ) then
  rm images_half_2.txt
endif

if ( -e failed_1_1_1 ) then
  rm failed_?_?_?
endif

set i = 1
while ( ${i} <= ${noofimages} )
    @ rest = ${i} % 2
    if ( $rest == 1 ) then
        echo $images[${i}] >> images_half_1.txt
    else
        echo $images[${i}] >> images_half_2.txt
```

```
    endif
    @ i += 1
end

set i = 1
while ( ${i} <= 2 )
    awk '{printf"%scr.fts\n",$1}' images_half_${i}.txt >! crossimages_half_${i}.txt
    set half_image = ` cat images_half_${i}.txt `
    foreach im ( $half_image )
        cat mjd.txt | grep $im >> mjd_half_${i}.txt
    end
    @ i += 1
end

######################################
### END PREPARING GLOBAL TXT FILES ###
######################################

source ${PROG_DIR}/make_log "Created txt-files for ${noofimages}"

exit 0
### End prepare_txt ####

### Begin global_variables ####
#!/bin/tcsh -f

source ${PROG_DIR}/make_log "Setup of global variables"

###################################
### SETTING UP GLOBAL VARIABLES ###
###################################
cd ${WORK_DIR}

set nx  =  512
set ny  =  512
set nx0 =  2048
set ny0 =  1045

set edge  =  5
set marg  =  0
set kerhw =  7

@ xload = $nx0 / $nx
@ yload = $ny0 / $ny

@ nxw = $nx + 2 * $marg
@ nyw = $ny + 2 * $marg

set images1 = ` cat images.txt `
###########################
### END GLOBAL VARIABLES ###
###########################

exit 0
### End global_variables ####
```

```
### Begin manipulations ####
#!/bin/tcsh -f

cd ${WORK_DIR}

#####################################
### MANIPULATIONS OUTSIDE BIG LOOP ###
#####################################
foreach im ( $images1 )
    if ( $im != $refim ) then
        echo $im
        ${BIN_DIR}/super_cutfits ${DATA_DIR}/${im}.fts ${im}cr.fts 512 1535 1 1024
    endif
end
echo $refim
${BIN_DIR}/super_cutfits ${DATA_DIR}/${refim}.fts ${refim}cr.fts 512 1535 1 1024
#########################
### END MANIPULATIONS ###
#########################

exit 0
### End manipulations ####

### Begin section_cross ####
#!/bin/tcsh -f

##################################
### SEPERATE SECTION FOR CROSS ###
##################################
cd ${WORK_DIR}

if ( -e shifts.txt ) then
    \rm shifts.txt
endif

set ihalf = 1
while ($ihalf <= 2)
    ${BIN_DIR}/cross ${WORK_DIR}/cross.par ${refim}cr.fts -f ${WORK_DIR}/crossimages_half_${ihalf}.txt >!
    ${WORK_DIR}/shifts_half_${ihalf}.txt
    @ ihalf += 1
end

if ( -e shifts.txt ) then
  rm shifts.txt
endif

cat ${WORK_DIR}/shifts_half_?.txt >! shifts.txt

rm l*cr.fts

#################
### END CROSS ###
#################

exit 0
### End section_cross ####
```

```
### Begin create_image2subtract ####
#!/bin/tcsh -f

set prefix = "${ix}_${iy}"

cd ${WORK_DIR}/half_${ihalf}/dif_${prefix}
set dir = ` pwd `
echo $dir

@ xl = 1 + $nx * ($ix - 1) - $marg
@ xu =     $nx *  $ix      + $marg

@ yl = 1 + $ny * ($iy - 1) - $marg
@ yu =     $ny *  $iy      + $marg

${BIN_DIR}/super_cutfits ${DATA_DIR}/${refim}.fts ${refim}c.fts $xl $xu $yl $yu
${BIN_DIR}/sfind ${WORK_DIR}/sfind.par ${refim}c.fts ${refim}c.coo

foreach im ( $images1 )
    set arg = ` fgrep ${im} ${WORK_DIR}/shifts.txt `

    @ n11 = $xl - $arg[2]
    @ n12 = $xu - $arg[2]
    @ n21 = $yl - $arg[3]
    @ n22 = $yu - $arg[3]

    echo ${im}: $n11 $n12 $n21 $n22

    ${BIN_DIR}/super_cutfits ${DATA_DIR}/${im}.fts ${im}c.fts $n11 $n12 $n21 $n22
    ${BIN_DIR}/sfind ${WORK_DIR}/sfind.par ${im}c.fts ${im}c.coo
end

${BIN_DIR}/ushort2float ${refim}c.fts ${refim}r.fts -min 1.0

echo xymatch:
foreach im ( $images2 )
    echo $im
    set failed = 0
    set nstars = ` wc ${im}c.coo `

    if( $nstars[1] < 10  ||  $status != 0 ) then
        set failed = 1
        set pname = "sfind or wc"
        goto "FAKE"
    endif

    set corr = `${BIN_DIR}/xymatch ${WORK_DIR}/xymatch.par ${refim}c.coo ${im}c.coo ${im}c.match`

    if( $status != 0 ) then
        set failed = 1
        set pname = "xymatch"
        goto "FAKE"
    endif

    echo shifts: $corr

    ##################################
    ##### correct if large shift #####
    ##################################

    if( $corr[1]>$edge || $corr[2]>$edge || $corr[1]<-$edge || $corr[2]<-$edge ) then
        set arg = ` fgrep ${im} ${WORK_DIR}/shifts.txt `
```

```
        @ n11 = $xl - $arg[2] + $corr[1]
        @ n12 = $xu - $arg[2] + $corr[1]
        @ n21 = $yl - $arg[3] + $corr[2]
        @ n22 = $yu - $arg[3] + $corr[2]

        echo nn: $n11 $n12 $n21 $n22
        ${BIN_DIR}/super_cutfits ${DATA_DIR}/${im}.fts ${im}c.fts $n11 $n12 $n21 $n22
        ${BIN_DIR}/sfind ${WORK_DIR}/sfind.par ${im}c.fts ${im}c.coo

        set nstars = ‘ wc ${im}c.coo ‘

        if( $nstars[1] < 10  ||  $status != 0 ) then
            set failed = 1
            set pname = "sfind or wc"
            goto "FAKE"
        endif

        echo correction:
        ${BIN_DIR}/xymatch ${WORK_DIR}/xymatch.par ${refim}c.coo ${im}c.coo ${im}c.match

        if( $status != 0 ) then
            set failed = 1
            set pname = "xymatch"
        goto "FAKE"
        endif
    endif

    ${BIN_DIR}/xygrid ${WORK_DIR}/xygrid.par ${im}c.match ${im}c.coeff

    if( $status != 0 ) then
        set failed = 1
        set pname = "xygrid"
        goto "FAKE"
    endif

    ${BIN_DIR}/resample2 ${WORK_DIR}/resample2.par ${im}c.coeff ${im}c.fts ${im}r.fts

    if( $status != 0 ) then
        set failed = 1
        set pname = "resample2"
        goto "FAKE"
    endif

FAKE:
    @ n11 = -$nxw
    @ n12 = -1
    @ n21 = -$nyw
    @ n22 = -1

    if( $failed != 0 ) then
        echo "${im}" >> ${WORK_DIR}/failed_${ihalf}_${prefix}.txt
        source ${PROGRAM_DIR}/make_log "$pname failed, preparing fake image ${im}r.fts"
        echo  $pname failed, preparing fake image ${im}r.fts
        ${BIN_DIR}/super_cutfits ${refim}r.fts ${im}r.fts $n11 $n12 $n21 $n22
    endif
end

cd ${WORK_DIR}/half_${ihalf}/dif_${prefix}
rm -f l*c.fts ### not necessary after here ###

exit 0
### End create_image2subtract ####
```

```
### Begin subtraction_photometry ####
#!/bin/tcsh -f

##############################################################
###  subtraction and photometry                            ###
##############################################################

set failed = 0

${BIN_DIR}/aga ${WORK_DIR}/aga.par blank_ushort.fts ref_${prefix}.fts
               ${WORK_DIR}/mrjimages_half_${ihalf}.txt

if( $status != 0 ) then
    set failed = 1
    set pname = "aga"
    goto "CRASH"
endif

@ xl -= 1
@ yl -= 1

echo "xl: ${xl} yl: ${yl}" >! xl_yl
echo >! ${FIELD}_${prefix}.db
echo >! ${FIELD}_${prefix}.cat

if( $status == 0 ) then
    echo  "completed section [${ix},${iy}]"
endif

CRASH:
if( $failed != 0 ) then
    echo  "CRASH for section [${ix},${iy}]; program: $pname "
endif

exit 0
### End subtraction_photometry ####

### Begin make_log ####
#!/bin/tcsh -f

set date = ` date `
echo $date >> ${LOG_DIR}/${LOG_FILE}
echo $1 >> ${LOG_DIR}/${LOG_FILE}
echo "" >> ${LOG_DIR}/${LOG_FILE}

exit 0
### End make_log ####
```