

Using DAOPHOT to locate KOI667.01

Jessica Ann Larson

A senior thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of
Bachelor of Science

Dr. Denise Stephens, Advisor

Department of Physics and Astronomy

Brigham Young University

August 2013

Copyright © 2013 Jessica Ann Larson

All Rights Reserved

ABSTRACT

Using DAOPHOT to locate KOI667.01

Jessica Ann Larson
Department of Physics and Astronomy
Bachelor of Science

The exoplanet research group at BYU has been observing and reducing data taken of objects on the Kepler planetary candidate list. We observed KOI667.01 at West Mountain Observatory (WMO) and realized that it was a false positive. Because the field we were interested in was very crowded we could not use aperture photometry to analyze the data. We used DAOPHOT in order to reduce the data and compute accurate light curves. We were able to determine which object within our crowded field was the eclipsing binary.

Keywords: daophot, eclipsing binary, false positive, West Mountain Observatory, Kepler

ACKNOWLEDGMENTS

I would like to thank BYU for the opportunity to study astronomy and for facilitating and funding my research. It has been a wonderful opportunity to work with this university and I am very grateful. I would like to thank Dr. Stephens for giving me the opportunity to join her research group and learn more about the study and identification of exo-planets. I would like to thank her for spending time teaching me how to handle and interpret data as well as prepare my poster for the AAS conference and this paper. I would like to thank Joseph Rawlins for all of the foundation work that he did before I joined that group that made it possible to study KOI667.01. I would also like to thank WMO for their cooperation in taking data for my research. Last of all I would like to thank Thayne McCombs and Andy Hernandez for giving me instructions and passing along information about DAOPHOT.

Contents

Table of Contents	vii
1 Introduction	1
2 Observations	7
3 Data Reduction	9
3.1 Aligning Images	9
3.2 DAOPHOT	11
3.2.1 Finding Paramaters	11
3.3 The Process of Using DAOPHOT	13
3.3.1 Input Paramaters	13
3.3.2 Running Tasks in DAOPHOT	17
3.4 Comparison Stars	21
4 Results	23
4.1 Future Work	24
Bibliography	31
Index	33

Chapter 1

Introduction

The Kepler mission was selected in 2001 to be the 10th NASA Discovery Mission. Its purpose is to find exoplanets, especially near the habitable zone, around nearby stars. The mission included the cooperation of many different scientists who helped build the telescope, process the data, create the flight plan, and program the computers. The Kepler mission was designed to conduct a survey of the sky in an area near the constellation Cygnus with center coordinates RA=19h 22m 40s and Dec=+44°30' 00". Figure 1.1 shows an image of the full area. Kepler's goal was to find stars with magnitudes that have a periodic change of about 1/10,000 in brightness and transits that last from 1 to 16 hours. These conditions suggest the existence of a planet orbiting the star. The Kepler Space mission began collecting data in May 2009 and continued to collect data until May 2013 producing a large list of planetary candidates from its survey of more than 100,000 stars (NASA Accessed August 6, 2013).

The objects on the planetary candidate list must have additional observations taken in order to confirm that they are planets, or figure out what they actually are. The Kepler data is publicly available in order for scientists to follow up these objects more quickly. To date, there are 134 confirmed planets from the Kepler planetary candidate list. Follow up research is done by ground based telescopes that search for possible false positives such as eclipsing binaries or white dwarfs.

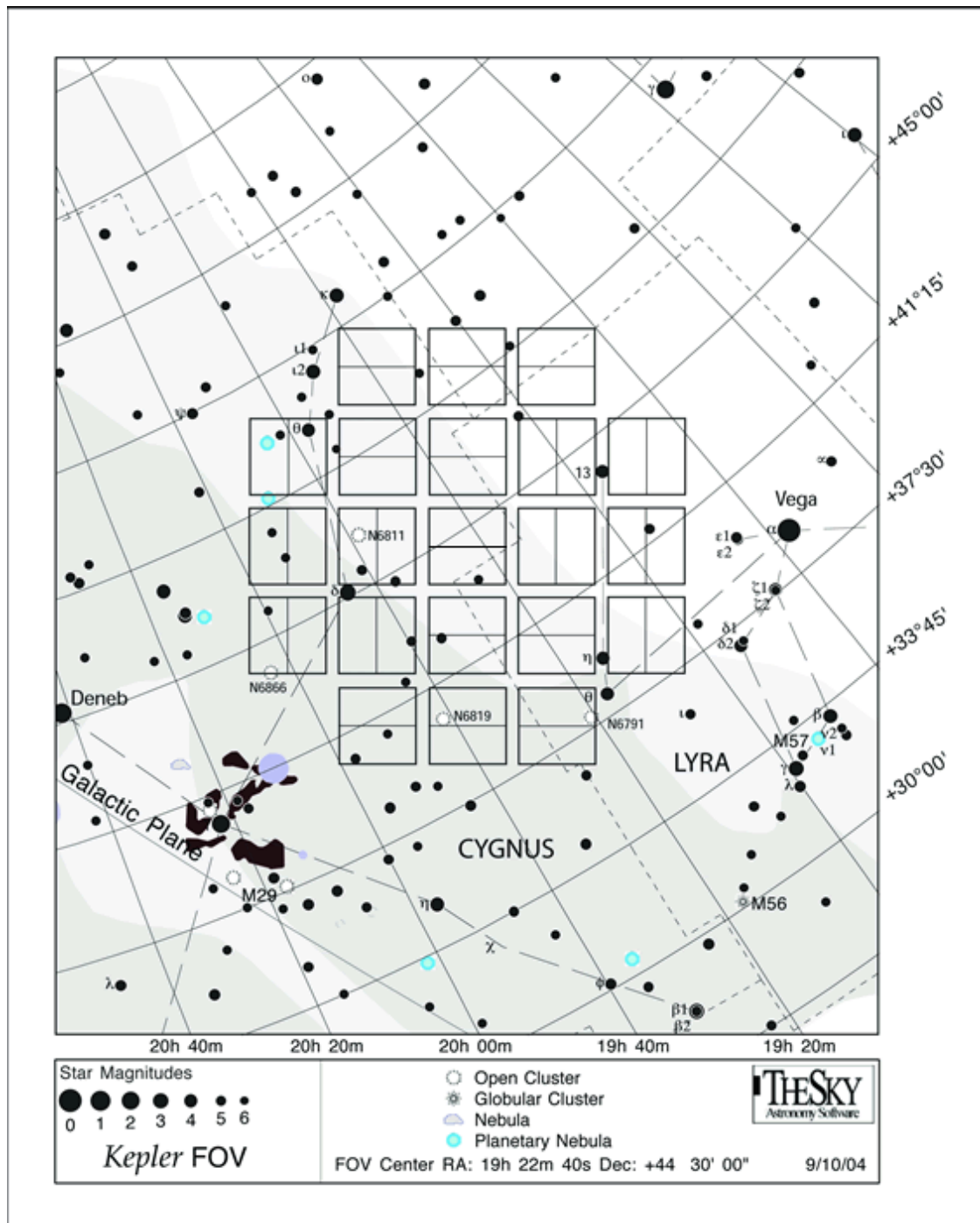


Figure 1.1 This shows the full area that Kepler took data on. Taken from NASA (Accessed August 6, 2013)

By taking clear pictures of the objects and constructing their light curves researchers are able to analyze individual objects to determine the best model representation. If an object appears to be something other than a planet it is marked as a false positive on the planetary candidate list in order to keep the public information updated.(NASA Accessed August 6, 2013)

At Brigham Young University (BYU), we have an astronomy research group doing follow up on hot Jupiter planetary candidates in Kepler. Our group is made up of an advisor and four undergraduate students. We used the Kepler planetary candidates to create a list of about 10 stars to take and analyze data on. The candidates selected were of interest because the information gained from the Kepler research gave parameters that appeared to be consistent with hot Jupiter type planets. Hot Jupiter planets were ideal for our project because they have large drops in magnitude, and therefore it is possible to detect their transits with WMO. The time for each object's transit was calculated as well as its position in order to take more data on it. One of the objects chosen to follow up was KOI 667.01.

When the data for KOI 667.01 came in we could see that the object was in a crowded field. We realized that this candidate might not be a planet at all but a false positive.Santerne et al. (2012) came to the same conclusion by looking at the Digitized Sky Survey and realizing that the object was in a crowded field. Since the region was so crowded they could determine that the Point Spread Functions (PSF) of different stars were being blended together.An image of the field is provided in Figure 1.2 with a circle around the crowded region where KOI 667.01 exists. This group analyzed 47 Kepler candidates with transit depths greater than 0.4%. The percentage of false positives that they found for this group of candidates was $34.8\% \pm 6.5\%$.

A false positive is a system that mimics the light curve of a transiting planet, but is actually something entirely different. Eclipsing binaries may appear to be transiting planets, especially when there is light from a nearby star. This could be caused by another star within the system making it a triple star system, or there could be a third star that appears to be close to the line of

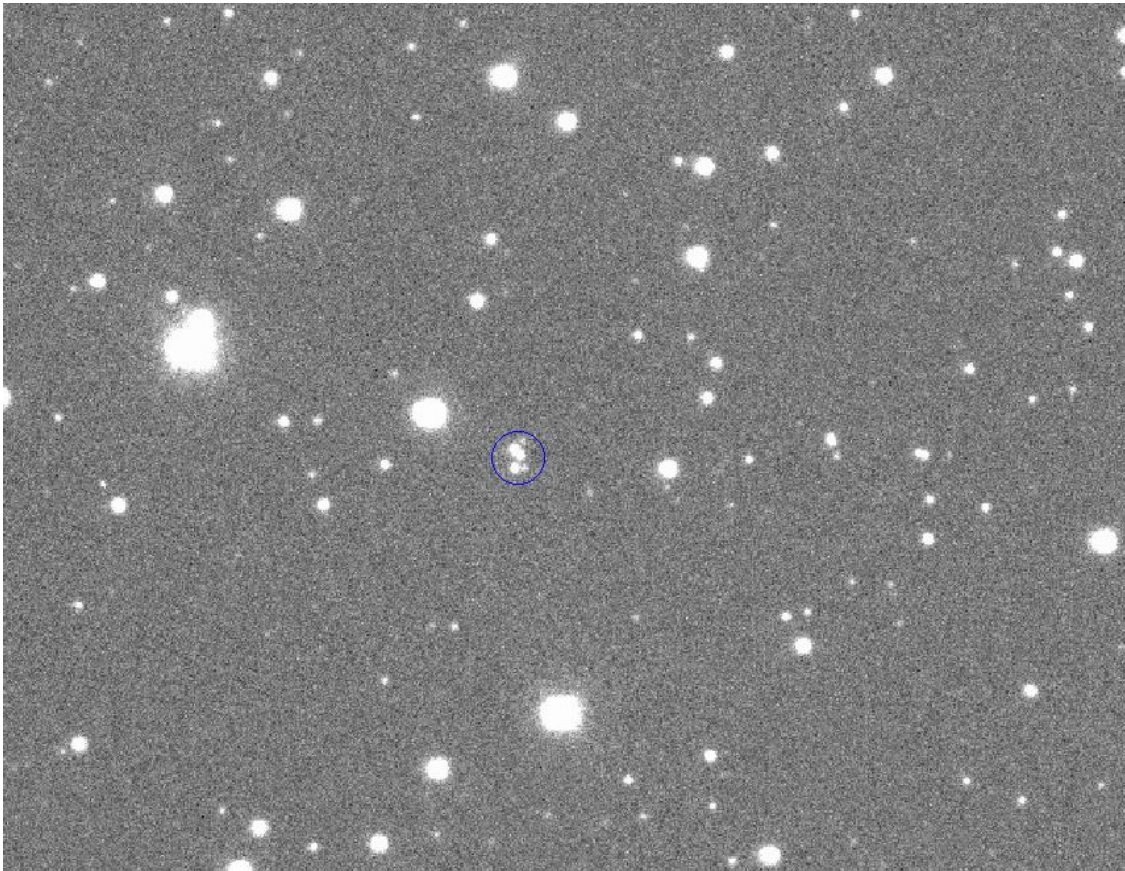


Figure 1.2 A frame of the crowded field taken at WMO.

sight. Either of these scenarios would produce a light curve that would appear to have a smaller percentage of a drop in magnitude. This is caused by the blended light from the third star contaminating the data and creating a shallower light curve (Seager & Mallén-Ornelas 2003). In order to find false positive systems it is important to continue to study objects and analyze their data carefully (Brown 2003).

There are several different ways that data can show the existence of an eclipsing binary rather than a transiting planet. When data is taken with different filters and the light curves show different transit depths based on the color this indicates an eclipsing binary. Another method is to use multiepoch spectroscopy to study the radial velocities which can indicate a stellar companion. Observations with higher angular resolutions should also be used to find false positives within planetary candidate lists (O'Donovan et al. 2006).

When there is a blended star system it is very difficult to be able to differentiate between the objects during the reduction process. We had to use DAOPHOT in order to see the individual objects and their light curves. DAOPHOT was developed by Stetson (1987) at the Dominion Astrophysical Observatory and was made for the purpose of getting better results in crowded star fields. One reason why DAOPHOT is more accurate is because the program does profile fitting iteratively. Using this program made it possible to obtain the information for individual stars even though they were very close together in our field.

Chapter 2

Observations

We took data with the 0.9 m telescope at BYU's West Mountain observatory near Provo Utah. The observatory is located atop a 6,850 foot mountain about an hour from the university campus. The 0.9m optical telescope CCD is a Finger Lakes PL-09000 model that is made up of $12\mu m$ pixels in a 3056×3056 array. The plate scale is $0.49''/\text{pixel}$ and the field of view is 25.2×25.2 (BYU Accessed August 6, 2013). KOI 667.01 was observed in the I and V filters on September 8th 2012 from about 11:30 PM to 3:00 AM local time (5:30 to 9:00 UT)

Chapter 3

Data Reduction

3.1 Aligning Images

Before using DAOPHOT, we aligned the images to make it easier to follow the stars throughout the night. To align the images, a program called MaxIm DL5 was used. To align with MaxIm DL5, you open the first few frames that need to be aligned. MaxIm DL5 does not have enough memory to handle a large amount of images, so you only align about five frames at a time. On the toolbar go to Process and then on the dropdown menu select Align. A box will open up in which the button Add All needs to be selected in order to choose all of the frames to be aligned and then press OK. Another box will open up that has a couple of options that need to be checked: make sure that **Align Mode** is on Auto – Star Matching and that **Reference** has the image name of the frame that will be used as the reference next to it. A screen shot of this box is shown in Figure 3.1. The default for the reference frame is the first frame opened up in MaxIm DL5. Once these parameters have been specified, hit OK and this should align the images. To make sure that the images are aligned, select View from the toolbar, then select Animate, Add All and OK. A box will open up and the Play button will rotate through all of the images that have been aligned. It is

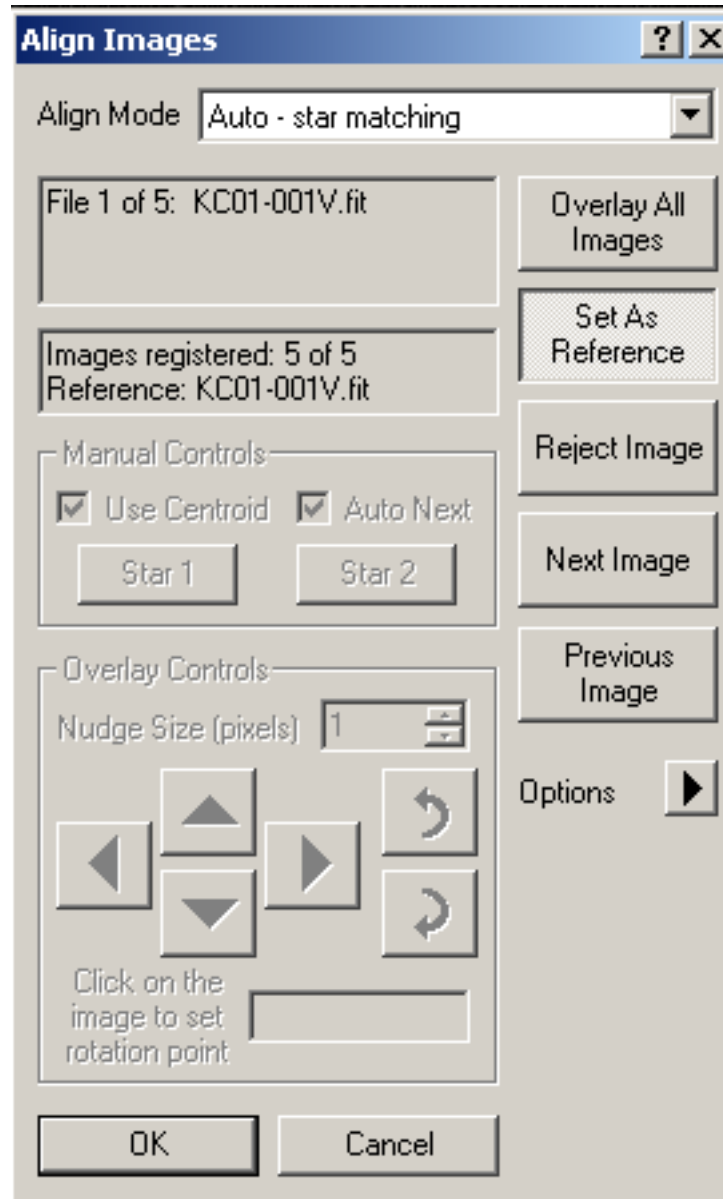


Figure 3.1 This is a box that needs to be checked when aligning images.

normal to see random cosmic rays show up in different places on each image but all of the stars should be aligned. To start the next batch close all of the images (make sure you save them) except the first. Keep the first image open to continue to use it as a reference frame. This process can be repeated for as many frames as necessary.

3.2 DAOPHOT

3.2.1 Finding Parameters

On inspection, the field of view for KOI 667.01 was too crowded to use aperture photometry, so we used DAOPHOT to reduce the data. Aperture photometry takes a circle and counts the flux within that circle and then takes the counts within an annulus and subtracts that as the background value. This is ineffective when objects are too close together because the aperture then contains light from other nearby stars. DAOPHOT is able to distinguish between two objects that are close together because it uses PSF fitting to find the maximum amount of light being detected in a pixel and centers on that. It continues with a process of finding the PSF fit and subtracting the background noise from it until it has an accurate fit. However, DAOPHOT has several parameters that must be set and carefully monitored throughout the data set to ensure the best possible results. This process can be somewhat time-consuming. Following is a list of parameters that need to be checked and explanations on how to set them.

READ NOISE and GAIN: These parameters are determined by the specific CCD that is being used. They only need to be determined once and then changed in the parameter settings. The READ NOISE and GAIN for the WMO 0.9 m have been determined to be 9.5 and 1.28. If the READ NOISE and GAIN have not been determined for a CCD they can be determined by using two unprocessed flat frames and two bias frames with the *findgain* task within IRAF. The units for these parameters should be electrons per Analog to Digital Units (ADU) (Maxwell Brigham Young University, Provo, UT. 2004).

The READ NOISE and GAIN are used in several different DAOPHOT tasks. They affect how the magnitude errors are computed in *phot* and *nstar*, they are used in the computation of optimal weights for the non-linear least-squares fitting code within *nstar* and used to compute the predicted

signal-to-noise ratios in *group*, and affect the sharpness in *nstar* (Davis 1994).

FWHM: An average FWHM needs to be found for every frame. In order to do this, use *imexam*; the user must be in the ds9 frame in order for this prompt to work. Move the cursor over a star in the field and press “a.” Within the information that is given back about that star is the FWHM under the heading ENCLOSED. Find the FWHM of about 20 stars in the frame, using stars located all around the frame. It is best to try and pick stars that are close to the same size as the object that is being studied. This will give you a good average for the frame. Throughout the night, the FWHM may vary. It will most likely gradually increase or decrease, because of the changing seeing and airmass. So it is important to change this parameter as the night goes on.

FITTING RADIUS and SKY ANNULUS: These are both determined by using the average FWHM that was just established. The FITTING RADIUS is found by multiplying the FWHM by 1.4 and the SKY ANNULUS by multiplying the FWHM by 3 or 4. For our data we multiplied the FWHM by 3.5.

STANDARD DEVIATION: There needs to be an average STANDARD DEVIATION of the background found for each night. This is found in a similar way to the FWHM. The difference is that the cursor should be placed over an empty region in the frame and the key “m” is pressed. This will bring up several parameters: one of them being the STDDEV. Take an average of around 30 points on the frame. This will also change slowly throughout the night, so it needs to be found for each frame. It is important to find the standard deviation by looking in the least crowded areas of the frame.

GOOD DATUM and PSF RADIUS: These are both found by looking at the radial graphs of several stars in the frame. When looking for the PSF RADIUS, change the parameters in *rimexam*

so that y_1 is set at zero and y_2 is set at 1000. Then look at the graph of several of the brightest stars by using *imexamin* and pressing the letter “r” while hovering over a star. When the graph appears, observe where the graph levels off on the x-axis. This is where the light from the star is dropping off into the background and the PSF radius will need to be set to at least that amount. Figure 3.2 gives a good example of a radial plot for the PSF RADIUS.

For the GOOD DATUM, change the y-axis parameters so that y_2 is INDEFF. Then look at the graphs of several bright stars and see what the highest y value is. If the graph levels off at the top the star is saturated and the Good Datum parameter needs to be lower than this number (Maxwell Brigham Young University, Provo, UT. 2004). Check several different stars and pick the lowest number to use for Good Datum. An example of a graph analyzed to find GOOD DATUM can be found in Figure 3.3. Make sure that there is some flexibility in the number that you choose in order to account for the detector bias level. If the graph levels off at 30,000 choose about 25,000 for the parameter of good datum. The GOOD DATUM parameter is a property of the CCD and so it only needs to be calculated once. An accurate measurement for good datum is used to identify bad data in *daofind*, *phot*, *psf*, *group* and *nstar*. It is also used to identify saturated stars in *phot* and *psf* (Davis 1994).

3.3 The Process of Using DAOPHOT

3.3.1 Input Parameters

Before using DAOPHOT on a frame, the input parameters need to be changed. Here is a list of where to change each parameter.

Imheader: check here to make sure that $LTV2 = 0$ (for WMO data), GAIN and READNOIS are the correct inputs for the CCD being used.

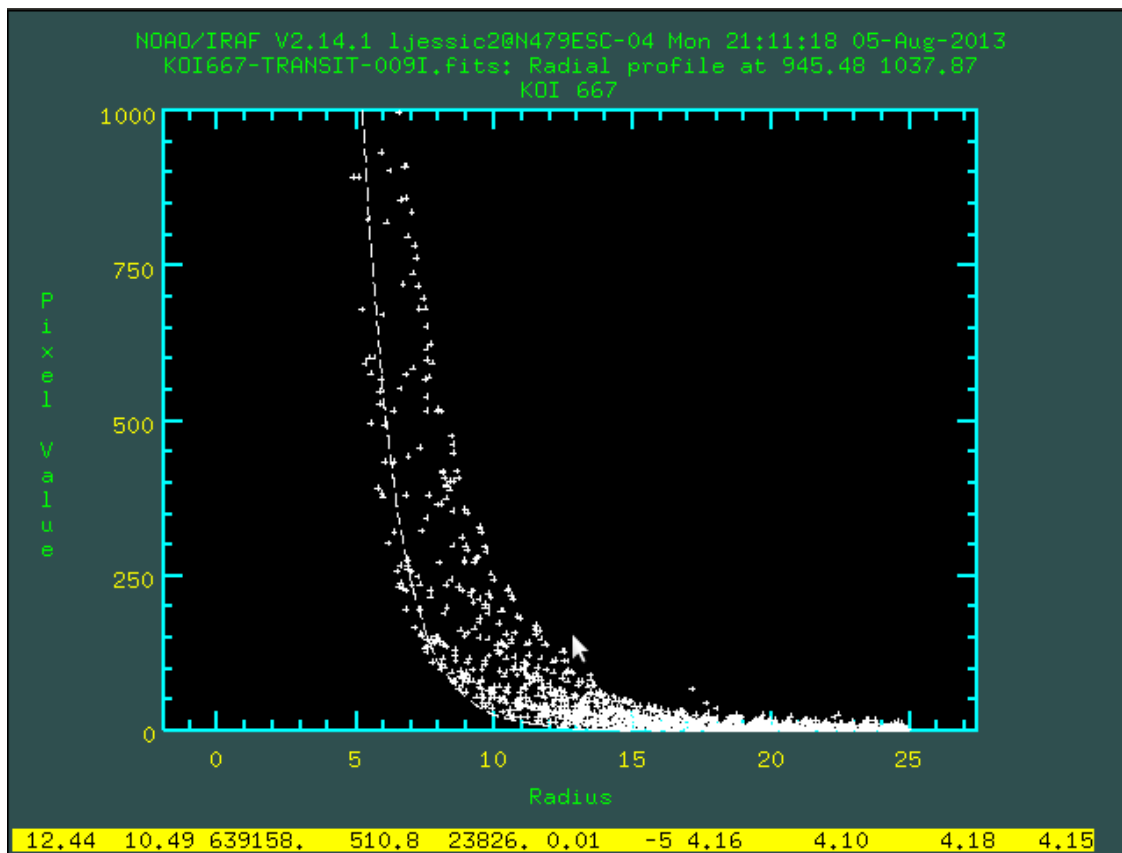


Figure 3.2 A screenshot of the radial plot that is used to find the PSF Radius. From this plot the PSF Radius should be about 13

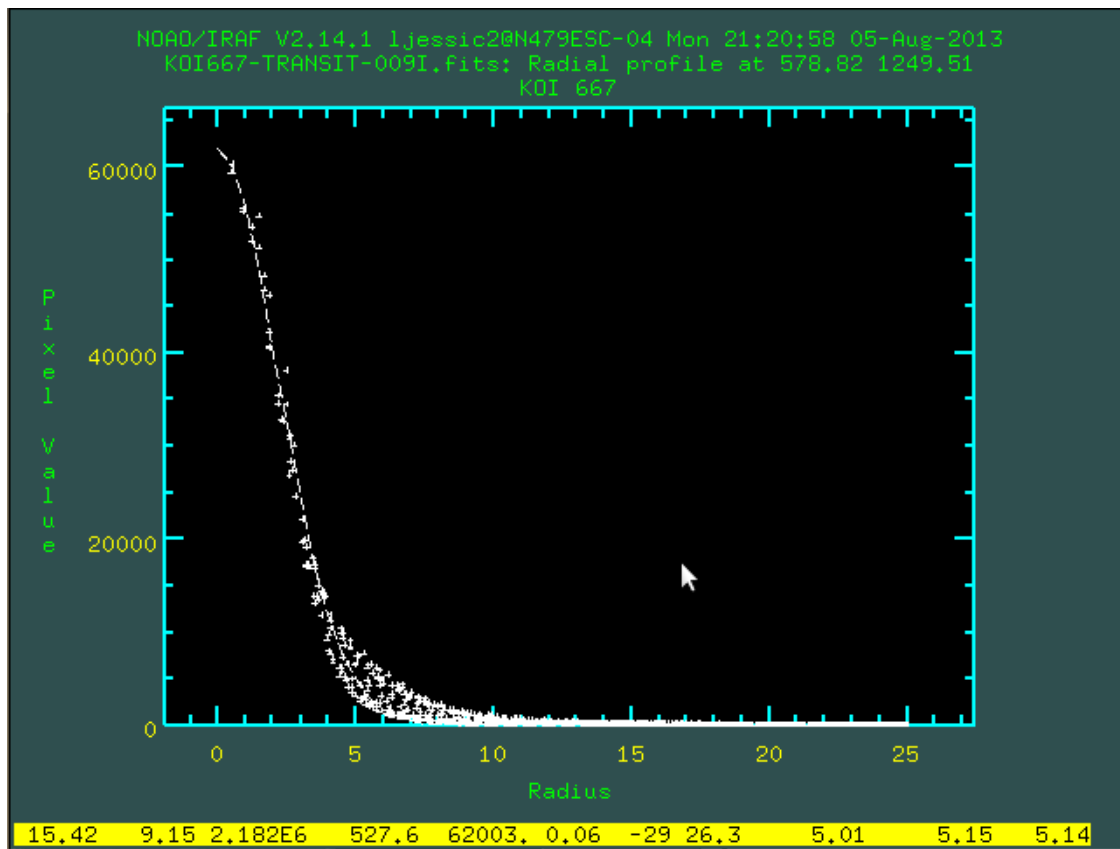


Figure 3.3 This is a screenshot of the radial plot to find Good Datum. From this plot Good Datum should be about 50,000.

Datapars: Put in the FWHM that was calculated for **fwhmpsf**, the STANDARD DEVIATION for **sigma**, GOOD DATUM in for **datamax**, make sure gain = GAIN and make sure the observation time is in hjd.

In the datapars parameters **fwhmpsf**, **sigma** and **datamax** are particularly important. They are used in order to optimize the size for the *daofind* algorithm, set a specific threshold for faint objects in the frames used with *daofind* algorithms, choose a FWHM of the *psf* for the *phot* task centering algorithms, and choose a first guess at an accurate FWHM of the PSF to be used in the *psf* task (Davis 1994).

Daopars: Put in PSF RADIUS for **psfrad**, FITTING RADIUS for **fitrad**, SKY ANNULUS for **sannulu** and make sure **groupsk** = no. These parameters are used in the *pstselect*, *psf*, *group* and *nstar* tasks. They are used to help compute the PSF, different groups of stars and PSF fitting parameters. The PSF RADIUS refers to the radius that will be used to define the PSF. The FITTING RADIUS is used to fit the PSF to each star in the field (Davis 1994).

Fitskypars: put in SKY ANNULUS for **annulus**. This set of parameters controls the sky fitting algorithm used by the *phot* task. Information from the *phot* task is then used to create sky values for individual stars and these are then used in the *psf* task in order to compute the PSF. These sky values are also used to find an average sky value which is then used in the *nstar* task. The parameters likewise are used to predict the signal-to-noise ratios in the *group* task (Davis 1994).

Photpars: change **apertur** to the FWHM. The parameters in photpars affect the *phot* task by forming the aperture photometry algorithm.

Centerpars: In centerpars it is important to pick the **cbox** and **maxshift** parameters carefully. Since our object was in such a crowded field we used **cbox** = 1 and **maxshift** = 2. These parameters

help determine the centering algorithm for the task *phot* which is an aperture photometry task. **maxshift** is the number of pixels that the center is allowed to move while **cbox** specifies the size of the box that will be used to search for the brightest peak.

3.3.2 Running Tasks in DAOPHOT

Run *daofind*: this double checks some key parameters, and finds the objects in the image that are good to use. For each star the task looks for peak intensities that are within a specified range. The computer then tries to determine approximations for centers, magnitudes, and other shape characteristics. It is important to have an accurate FWHM for this task because if the FWHM is incorrect some stars in the field might be missed or objects that are not stars may be included. Another important parameter is the STANDARD DEVIATION parameter that helps give a detection threshold and if the threshold is incorrect there may be too few or too many objects identified in the field. It makes a file with the number of objects in it along with their coordinates from the frame under the heading **ID**. Other information that is included in this file are **xcenter** and **ycenter**, which give the coordinates of each object; **mag**, which gives an estimated magnitude; **sharpness**, which is the ratio of the amplitude of the delta function to the amplitude of the Gaussian for a specific object; and **roundness**, which is created by taking the x and y pixel distribution and fitting a gaussian function to it. This is the first thing that should be run since it is used for later tasks. It is only necessary to do this on the first frame and then use that same file for all of the other frames.

Since we wanted to take the information that we found with DAOPHOT and run it in VARSTAR we didn't use *daofind* because it would have given us over 1000 objects and would have made using VARSTAR cumbersome. Instead, we manually chose 124 objects from the frame to use as comparison objects.

Run *phot*: This does aperture photometry on the list of stars found in *daofind* (or the objects picked manually). It examines all of the stars from the list and produces some initial computations of their centers, sky values, and initial magnitudes. The centers and magnitudes that are found here will be used later in the *psf*, and *nstar* tasks as starting values for their least-squares PSF computation and fitting routines. In the *group* task the centers and magnitudes are used to compute the signal-to-noise values. The *psf* task also uses the sky values in order to determine the PSF model. *nstar* uses the sky values to create group sky values. Last of all *group* uses the sky values to estimate the signal-to-noise ratios (Davis 1994).

There are several important parameters that *phot* uses to compute information. The **algorithm** is the algorithm *phot* uses to find the centers of each object. *phot* uses the **algorithm** to compute the sky values. If the region being studied is crowded, the **algorithm** should be on the default setting “mode”. If contamination is caused by nebulosity or large galaxies then “median”, “centroid”, or “crosscor” should be used for **algorithm**. **Annulus** and **dannulus** are parameters that specify where the sky annulus is with regards to the star. It is important to have a position far enough away to avoid contamination from the star, while still being close enough to show what the intensity distribution is like under the star. The **apertures** parameter gives a radius of photometry aperture that will be used for the rest of the tasks. The last parameters worth mentioning are **datamin** and **datamax** which are used to find bad data within the photometry and remove it before sky fitting takes place. These stars will have a magnitude of INDEF in the output file (Davis 1994).

The *phot* output will list information for each star. The **xcenter** and **ycenter** give coordinates for each star that can be compared to the previous coordinate list. **msky**, **stdev** and **nsky** are lists of estimated sky values, standard deviations and the number of pixels that were used to compute the information. **sum** and **area** contain information the total counts in the photometry aperture and area of the aperture with units in pixels squared. Under the titles **mag** and **merr** are found the magnitude and magnitude error. **Merr** is the error in magnitude found from Poisson noise from

the star, read noise and sky noise.

When *phot* is run there may be some errors in the output. Just because there is an error here doesn't mean that there will be an error on that object when *nstar* is run on it (Davis 1994). But any objects that do have errors for their magnitudes should be checked to make sure that they are marked as close to the center as possible. We also had to check our objects since they were so close together. Because two of the objects were very faint, we had to be aware of their markers being shifted on top of each other. When this happened the magnitude of the faint star would appear to be the same as the brighter star. This is why it is important to change the **maxshift** and **cbox** in *centerpars* so that when stars are close together the center points will stay on individual objects.

Run *pstselect*: It will ask you to pick the number of objects that will be used to create a set of candidate psf stars. These will be the best psf stars found in the set. It is usually good to pick a small amount (10-30) we used a set of 25. It will pick bright, evenly distributed, isolated stars. They will be used to create templates for psf models. **Psfrad** and **fitrad** are both parameters used in *pstselect* for the purpose of eliminating stars that are in crowded areas of the frame. If **psfrad** is too small there are problems with fitted stars not being subtracted fully from the image. If **psfrad** is too big there arises a problem of the task taking a lot longer because of data extractions and subtractions that are avoidable. The *pstselect* output contains information copied directly from the photometry file and so it should not have been altered in any way (Davis 1994).

Run *psf*: This will build a point spread function model from the input *phot* file. It can be done manually (interactive = yes) or the program can complete and write the PSF without any input (interactive = no). We just let the program create the PSF. It is recommended that beginners let the program compute it, but with more experience doing it manually could produce better results. This PSF model is used in *nstar* to do PSF fitting photometry, and it is used by *group* to find the magnitudes of stars that previously had magnitudes of INDEF. The two most important parameters

for this task are *psfrad* and the fitting radius. The *psfrad* is used to select a region over which the PSF model will be defined. The fitting radius is used in picking the stars and the areas around them that are used in creating the analytic component of the PSF model. The output from *psf* is a star list *test.psf.#* with the information of the stars that ended up being used to create the PSF model. It also makes a PSF star group photometry file *test.psg.#* that contains information about each of the stars used in the PSF model as well as their neighboring stars. Within the header produced the information of all of the parameters used to create the model are stored as well as the best parameters of the selected analytic function, a record of all the stars used to compute the PSF and look up tables (Davis 1994).

Run *group*: This will take the *phot* file and create groups of objects based on their proximity and magnitude level. It is used in conjunction with *nstar*.

Run *seepsf*: This task uses the look up tables to create an image of the PSF model. The image created will be an ideal star with magnitude equal to the magnitude of the *psf*. This image can be viewed to analyze and detect any major problems with the PSF. If the image does not look like a star, then something has gone wrong, and it would be best to go back and check the parameters found before beginning *nstar*. While this task can be helpful it is optional.

Run *nstar*: This will fit the PSF to the groups of stars created from *group* simultaneously. This task is useful in having more control over the groups being used and in fitting stars that are in groups that are spread out across the frame. Another task that can be used instead of *nstar* is *allstar* and it may produce better results.

nstar produces the files *image.nrj* and *image.nst*. The *.nrj* contains stars that were rejected while the *.nst* file contains the photometry database. These files contain most of the usual information as well as **niter** which records the number of iterations that were used to fit each star, **chi** which

shows the goodness of fit, **sharp** that shows the stars shape, **pier** which is an integer error code and **peror** which is an error string. The **chi** should be 1, and if it isn't then the object may be made up of more than one star, the parameters are wrong or the PSF model is wrong. The output **sharp** is the difference between the object's width and that of the PSF model. This value should be approximately zero (Davis 1994).

Pull up the .nst file, and make sure it contains the objects that you are interested in. Since we were using VARSTAR, we also had to make sure that the .nst files contained all the same objects. In some of the files, objects would be missing because their magnitudes were indefinite and so their information was not in the .nst file. When this happened we went back and redid that frame until all of the objects were in the .nst file.

After creating .nst files for all of the frames we had to create a file to run in VARSTAR. So we txdump'ed the star's id, magnitude, otime, xairmas and filter from the .nst files into a new file. This information needed to be fixed so that the stars were represented in the same order for each frame, which wasn't the case in the .nst files because they listed the objects in order of magnitude instead of by ID number. To fix this we put in the command

```
sort -n -s -k3,3 -n -k1,1 s.lst > se.lst
```

where s.lst is the name of the file that needs to be sorted and se.lst is the name of the new sorted file.

3.4 Comparison Stars

After running the file through VARSTAR and analyzing those results to figure out which star was the eclipsing binary, we decided to use comparison stars to find the absolute magnitudes for the eclipsing binary. To do this we found the frames taken at WMO of standard stars and reduced those frames in the exact same way that we reduced the frames for our data. Then we averaged

the magnitudes that were computed for those standard stars. By looking up their magnitudes on the Landolt Photometric Standard Stars list (University of Arizona Accessed August 6, 2013) We were able to find the average change between the instrumental magnitude and found magnitude. We applied this to our eclipsing binary to extract the absolute magnitudes.

Chapter 4

Results

By examining the graphs created from the VARSTAR output we were able to identify that the eclipsing binary was the star circled in Figure 4.1. The graphs of the five stars that we analyzed can be found in Figures 4.2 and 4.3. Figure 4.4 has each of the five stars labeled. There is a clear drop in Star 1 but not the other four indicating that the eclipsing binary was Star 1. It can also be seen from looking at the star positions in Figure 1.2 that the larger star below the eclipsing binary had been contaminating the data causing it to look like a transiting planet.

Once we knew which object the eclipsing binary was we analyzed that graph. These graphs can be seen in Figure 4.5. After finding the absolute magnitudes and averaging the data points to remove some of the noise we could see the light curves more clearly. The drop in magnitudes indicated that there was an object transiting this star. In the V filter the change in magnitude is about 0.4 while in the I filter the change in magnitude is about 0.2. The difference of the depth of the magnitudes in each filter illustrates that this is an eclipsing binary and not a planet. As a planet transits a star it will block about the same amount of light in all wavelengths because a planet will not be emitting much light of its own. So for a planet the drop in magnitudes for all filters should be the same. But when another star passes in front of a star, it will be emitting its own light and will cause different drops in the different wavelengths depending on what type of star it is. This

will cause the drops in magnitude to vary in the different filters.

From Figure 4.5 it is clear that we were able to see a part of a transit but did not get the entire transit. Starting at time 1:10 AM local time (7:10 UT) KOI 667.01 came too close to the horizon. This caused the airmass to become too large climbing from 1.06 at the beginning of the night, reaching 1.6 at about 1:10 AM and continuing to climb until it reached 2.06 at the end of the night. When the airmass became 1.6 this indicated an altitude of about 38° and by the end of the night was at about 30° . These large airmasses caused the point spread function of the objects to spread out. The point spread function from the lower objects bled into the point spread function of the eclipsing binary causing inaccurate magnitudes after 1:10 AM.

4.1 Future Work

There has been more data taken of KOI 667.01 that has not been reduced yet, but there are plans to study the new data in the fall. The contribution from this paper will help future students learn how to use DAOPHOT in their research. From the light curves produced so far we were able to determine that the object we were interested in was one of the fainter stars. This helped determine better integration times that were used when more data was taken for KOI 667.01 at the Dominion Astrophysical Observatory (DAO). It is hopeful that this new data will contain more information such as an entire transit. It is recommended that KOI 667.01 is followed up in order to obtain more accurate light curves and in order to model the stars within the system.

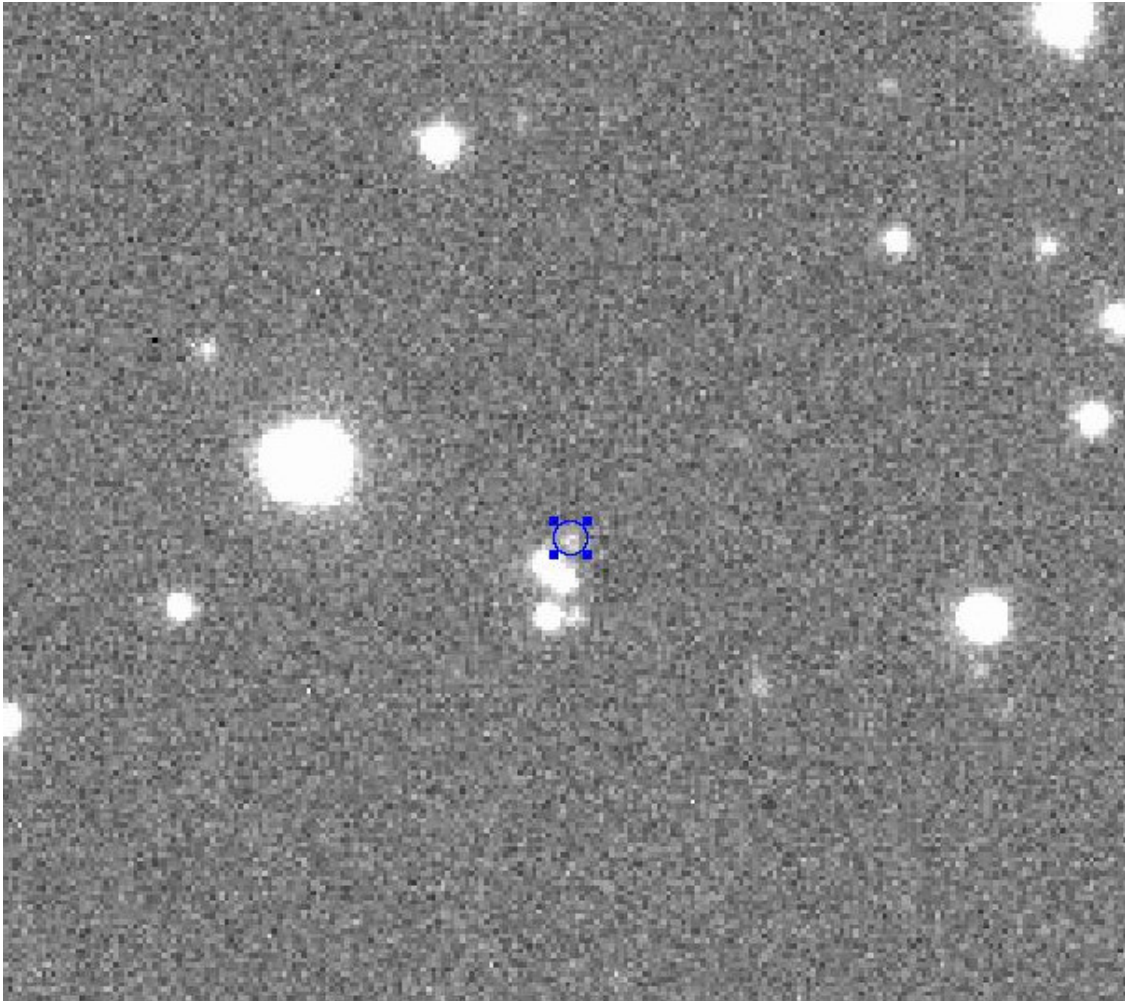


Figure 4.1 This is a picture of the eclipsing binary circled.

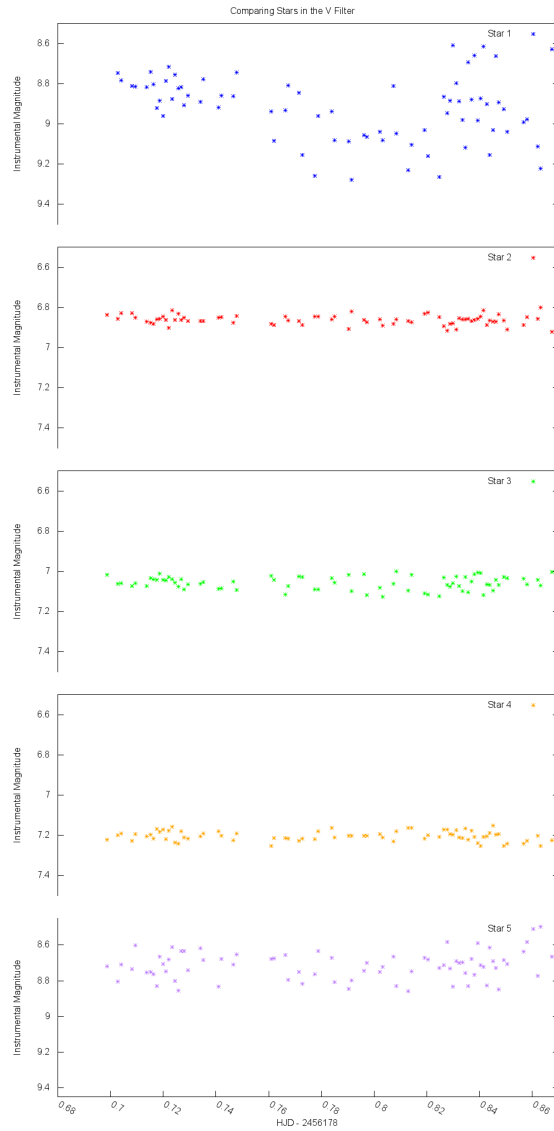


Figure 4.2 These are the graphs of the five stars in the V filter made from the VARSTAR output.

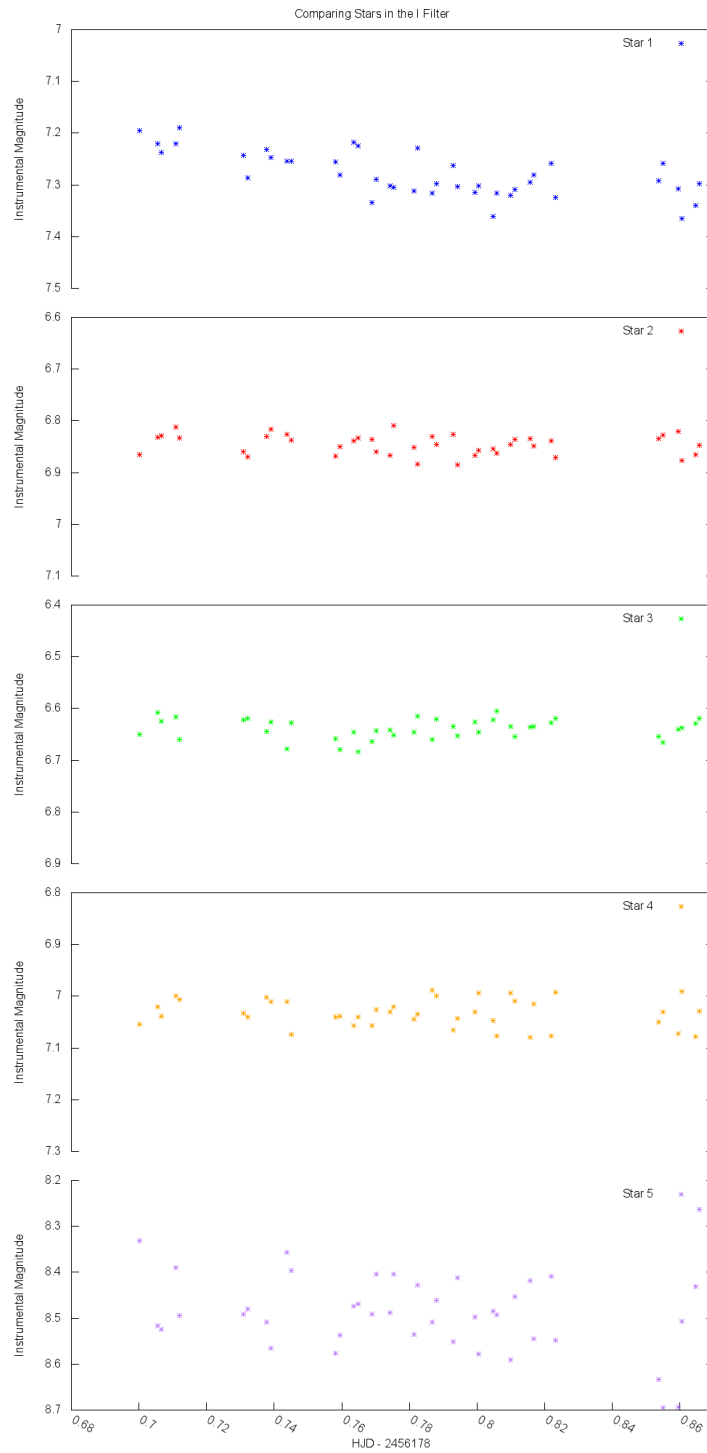


Figure 4.3 These are the graphs of the five stars in the I filter made from the VARSTAR output.

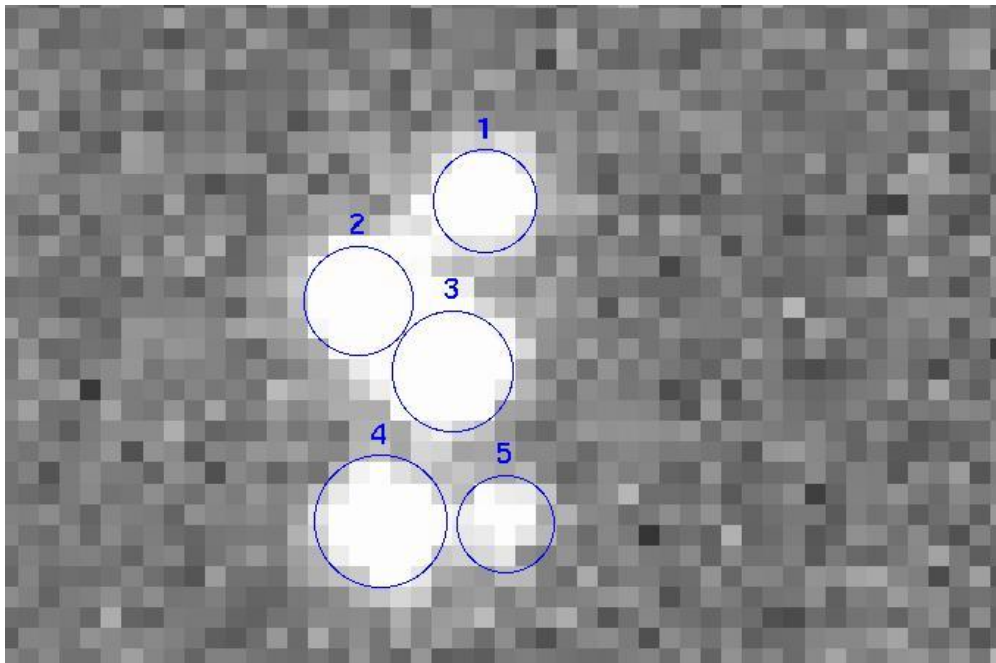


Figure 4.4 This is an image of the five stars labeled, who's light curves we analyzed.

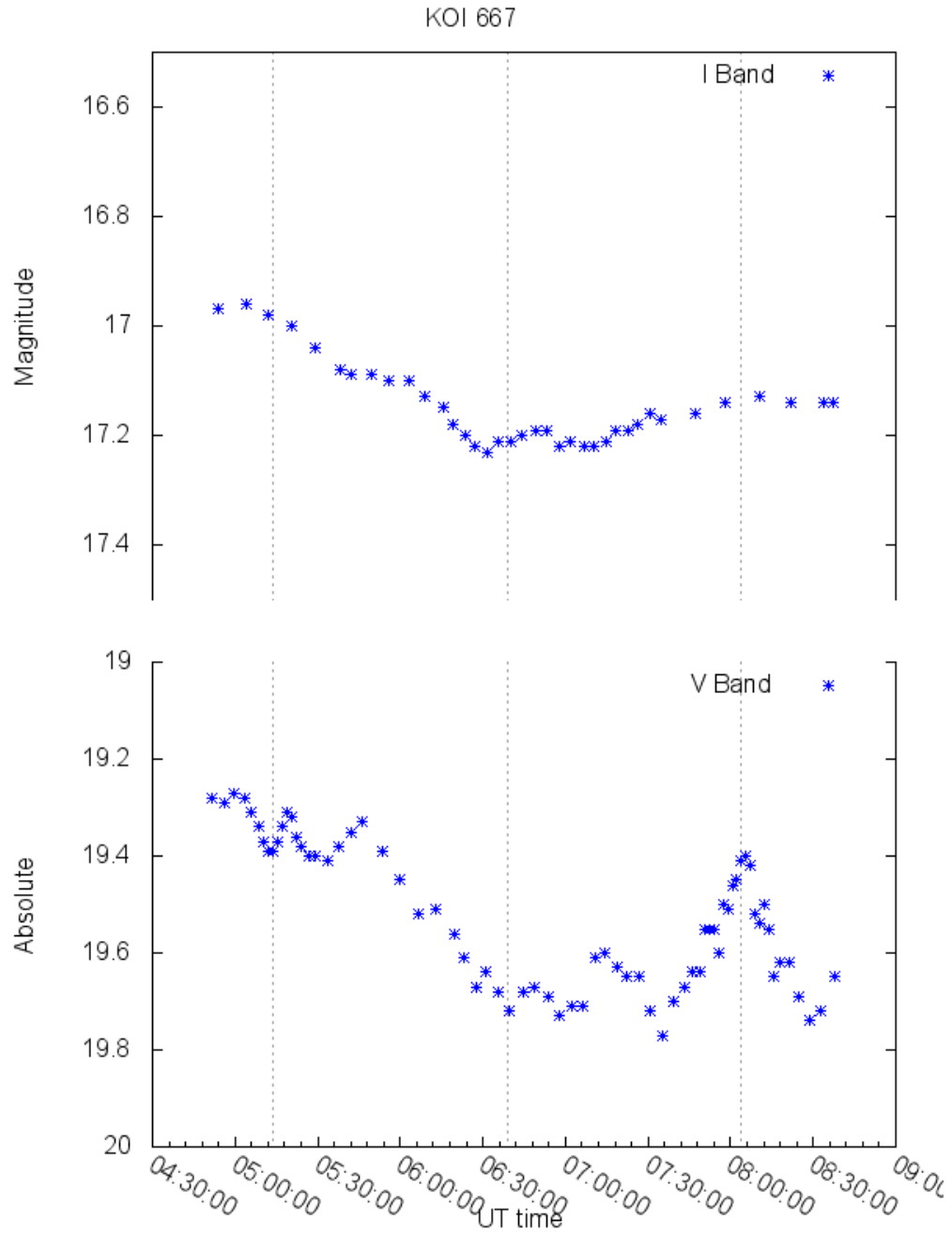


Figure 4.5 These are the light curves of star 1 in the I and V filters with the absolute magnitudes shown.

Bibliography

Brown, T. M. 2003, ApJ, 593, L125

BYU. Accessed August 6, 2013, <http://wmo.byu.edu/36inch/Instrumentation.aspx>

Davis, L. E. 1994, in IRAF Programming Group, National Optical Astronomy Organization, Tucson, AZ.

Maxwell, J. Brigham Young University, Provo, UT. 2004, Master's Thesis

NASA. Accessed August 6, 2013, <http://kepler.nasa.gov/>

O'Donovan, F. T., et al. 2006, ApJ, 651, L61

Santerne, A., et al. 2012, A&A, 545, A76

Seager, S., & Mallén-Ornelas, G. 2003, ApJ, 585, 1038

Stetson, P. B. 1987, PASP, 99, 191

University of Arizona. Accessed August 6, 2013, <http://james.as.arizona.edu/~psmith/tableA.html>

Index

centerpars, 16

daofind, 13, 16–18

daopars, 16

daophot, 5, 11, 13, 17

datapars, 16

fitskypars, 16

FITTING RADIUS, 12

FWHM, 12, 16, 17

GAIN, 11

GOOD DATUM, 12

group, 12, 13, 16, 18–20

imheader, 13

Maxim DL, 9

nstar, 11–13, 16, 18–20

phot, 11, 13, 16–20

photpars, 16

psf, 13, 16, 18, 19

PSF RADIUS, 12

pstselect, 16, 19

READ NOISE, 11

seepsf, 20

SKY ANNULUS, 12

STANDARD DEVIATION, 12