Instructions for

**Photometry and Analysis of Baker Observatory Celestron 14"/Apogee Alta U77
CCD Images Using IRAF**

Robert S. Patterson
November 21, 2011

Table of Contents

Part 2.
CCD Aperture Photometry

Other Related Manuals

"A User's Guide to the IRAF Apphot Package"
by Lindsey Davis, NOAO

"A User's Guide to Stellar CCD Photometry with IRAF"
by Philip Massey and Lindsey Davis, NOAO

"apphot" Help Pages

I. Aperture Photometry Using IRAF

1. The following packages are needed for doing aperture photometry: noao, digiphot, apphot, plot, tv. If they have not already been loaded automatically by your 'login.cl', load them by entering the package name.

2. There are several useful graphics plotting tasks in IRAF for examining your images:

   implot - to plot individual rows or columns of an image
   surface - to plot a perspective drawing of an entire image
   contour - to plot contour lines of equal intensity of an
       entire image

   Try these tasks on an image or two. Use the help pages if you have any questions.

3. Images may be displayed using this command:

      cl> display 80008 1

   You can use the task 'imexamine' to measure the FWHM of the stellar images, to plot lines, columns, histograms, contour plots, surface plots, etc. See the help pages in this folder.

4. Background information about the "apphot" package is given in "A User's Guide to the IRAF Apphot Package" by Lindsey E. Davis. Explanations and examples of the steps needed to do aperture photometry are given in "A User's Guide to Stellar CCD Photometry with IRAF" by Philip Massey and Lindsey E. Davis, NOAO. Refer to both of these guides as necessary. However, for differential aperture photometry you only need to use the steps indicated below.

5. Proceed by editing (epar) the parameter sets (daofind, phot, centerpars, datapars, fitskypars, findpars, and photpars) for the 'apphot' tasks. The parameter settings that <u>must</u> be used are located in Part IV "Recommended 'apphot' Parameter Sets". (A few stars may require "special parameters".) Note that the task parameters can be changed by just typing the set name, such as, 'datapars', except for 'phot' itself, for which it is necessary to type 'epar phot'.

6. There are now two different ways to proceed. The first is essentially automatic, and is known as "batch mode". It works well as long as the above parameter sets have been correctly edited. The second is interactive and takes much longer. <u>It should be sufficient to use the "batch mode" method for differential aperture photometry of relatively bright stars.</u>

BATCH MODE:

A. DAOFIND

1. First run the task 'daofind' to automatically find the brightest stars in each image. Use the "EditCoo" binder to make sure that you find coordinates for all the numbered stars in a field. The task will prompt you for several parameters (e.g., fwhm and threshold) whose default values should work well unless you are seeking really dim stars or the seeing is really bad or the tracking is really, really bad.

cl> daofind *.imh

2. Information about the stars found is printed out for each image as the task runs. If too many or too few "stars" are being found, something is wrong. You can abort the task by typing <Ctrl-C> and then "flpr". Erase the *.coo.N files and find out what's wrong before trying again.

3. The 'daofind' task puts the coordinates of the stars found into files that end in "coo.1", (e.g. 90008.imh.coo.1). Re-run 'daofind' as needed on all images or just some images until you have found the coordinates for the numbered stars in the "EditCoo" charts for each program star. Re-running the tasks generates "coo.2, coo.3, etc." files. Make sure that the final "coo" files end in ".1" only. Next, delete all ".coo" files for which saturation occurred in the image (see the uYMMDD.sat file), and rename saturated files: e.g., imrename 40007.imh x40007.imh, if this hasn't already been done.

B. EDITCOO [WRITTEN BY JUSTACE R. CLUTTER (2002)]

1. Next, you must prepare the "coo.1" files for photometry by using the 'editcoo' task. This ensures that the proper number of stars is ordered correctly in the "coo.1" files. The order is: program star, comparison star, and check star (if any). A typical line in a "coo.1" file looks like this:

```
123.456  543.012  -5.123  0.725  -0.025  0.055  1
x center  y center  mag_est  sharp  round            id
```

2. Proceed by displaying the first image:

cl> displ 90001.imh and then running the task 'editcoo'
cl> editcoo 90001.imh.coo.1

3. Use the mouse to place the cursor on the center of the program star on the display. The numerical values for the x and y coordinates will show on the image display window. Find the corresponding line in the "coo.1" file and place the cursor on that line by using keypad arrows.

4. Cut this line with the key combination **<Ctrl-K>**.

5. Using the arrow keys, re-position the cursor at the beginning of the line under the last comment line (the ones that start with #).

6. Paste the line that you cut at this position with the key combination **<Ctrl-U>**.

7. Repeat this procedure for the comparison star and then the check star.

8. Delete all other data lines in the file using the cut key combination **<Ctrl-K>**.

9. Save the file: **<Ctrl-O>**, **<Enter>** and then exit 'editcoo' with **<Ctrl-X>**.

10. Repeat the procedure for all of the images that require photometry.

C. PHOT

1. The aperture photometry can now be done (rather quickly!). First, **make sure** that the "apphot" parameter sets, including 'datapars' are correct. **NOTE: Some stars will require re-running with SPECIAL parameter sets! See last section of Image Reduction & Analysis binder).** When ready, proceed by typing:

cl> phot *.imh

2. You will see a short version of the results listed on the screen. The complete results are stored in files e.g. (90008.mag.imh.1). These can be displayed on the screen by "type 90008.mag.imh.1" or "page 90008.imh.mag.1", but a much cleaner way is to use the 'txdump' task.

3. Send the following to a file e.g., u11028.qwk for printout using the redirection symbol ">":

cl> txdump *mag.1 xairmass,ifilter,otime,mag,merr,sum,msky,nsky yes > u11028.qwk

The above is one continuous line. No spaces are allowed between items in the list. Note that, in this case, u1 stands for "Apogee u77, year 2011".

4. Next, repeat with these changes:

cl> txdump *mag.1 xairmass,ifilter,otime,mag,merr,sum,msky,nsky yes head+ > u11028.mag

The above is one continuous line. No spaces allowed between items in the list.

D. Database Option 1: Night_Sum [Written by Justace Clutter, 2002]

1. Enter "Night_Sum --help" to look at the usage page for this task.

2. The purpose of the first run of Night_Sum is to place the photometry results in a temporary database "/u2/RSP/u77test" in order to check the validity of the delta magnitudes. At this time a printout is also produced. Use this command line example:

cl> Night_Sum –m u10311.mag -i --TeX --base /u2/RSP/u77test   (there is no = after "base" when running Night_Sum while in IRAF). Be very careful to distinguish between the two databases "/u2/RSP/u77test" and "/u2/RSP/u77"!!!

This task will also generate a printout for studying the instrumental and differential magnitudes for a given night.

3. Next, use the task "plotdelNite" to make light curves for individual stars on a given night. Run, for example:

   cl> !plotdelNite u77test ysg 331777 0.5 2455371.620   2455371.626 (! for Linux)

     where the HJDs are chosen to be just before and just after the observation times for a given star. Print the resulting three-panel set of light curves. Submit the light curves and the Night-Sum_Run printout for approval.

4. If the Night_Sum_Run is returned for corrections, you will have to go back to the .mag.1 files and delete the bad .mag.1 files. Then run txdump again to generate a new u10311.mag file. This corrected file will be inserted into the real u77 database.

5. The purpose of the second run of Night_Sum is to place the (corrected) photometry results in a permanent database "/u2/RSP/u77", for example:

cl> Night_Sum –m u10311.mag -i --TeX --base /u2/RSP/u77 (No need to use ---TeX again if there were no corrections)**.**

6. Put the (corrected) Night_Sum_Run printout and light curves in the Night-Sum binder.

E. Database Option 2: PhotDB [written by Alex Hutchins, 2008]

1. This task produces differential instrumental magnitudes and error estimates from the ".mag" file made in C.4., and is used to insert the results into the u77 database. 'PhotDB' requires that 'editcoo' has already been run on each image first.

2. 'PhotDB' is run twice. The first time, the output must be sent to a PROOF database. This output requires **approval** before running 'PhotDB' a second time in order to place the results into the u77 database.

3. Assuming you are in the directory where the mag file to be proofed is located:

cl> python PhotDB.py <mag file name> /u2/RSP/u77PROOF

4. After receiving approval, re-run 'PhotDB' like this:

cl> python PhotDB.py <mag file name>

5. Printouts are to be placed in the appropriate binders.

F. Data Analysis

Use the "plotdelV" script [written by Shawn Poindexter, 2003] to make light curves from the various delta magnitude files. Also use Microsoft Excel to make light curves and light curves with error bars. See the instructions for these procedures.

IMPORTANT

If you modify any files in /u2/RSP/u77 on the pc-astro server, you must change permissions as follows:

!chmod 660 <filename>
!chgrp rsp <filename>

where the ! is used while in IRAF to drop down to the Linux Operating System.

INTERACTIVE MODE:

      1. This method allows you to set various parameters as you go, based on interactive graphics plots. To do this you will need to make contour plots of the images:

      cl> contour 50008 floor=10 ceiling=210 >G 50008.ctr

  The purpose of using a ceiling is to leave blank cores in the star contours so that it is easier to position the cursor. The ">G" in the command line places the graphics characters in a file that you name "50008.ctr", for example, with the extension "ctr" meaning a contour file. Note that the task will not display the contour plot unless you leave off the ">G 50008.ctr".

      2. Repeat using 'contour' for all the images for which you wish to do aperture photometry.

      3. The 'phot' task will run if you use the graphics cursor to locate stars on a contour file. Use of 'apphot' typically requires the following steps:

      cl> set stdimcur=stdgraph
      cl> contour 50008 >G 50008.ctr (Unless you already did it)
      cl> phot 50008 coords="" interactive+

  Next, display the .ctr plot in the Tek graphics window:

      cl> :.read 50008.ctr<Return>

  {After the contour file is displayed, move the cursor to
   a star of choice and then press "i"}

  where 50008 is a sample image name, don't use the underline.

      Press "v" to enter the interactive parameters. When done press "q" to quit, then "w" to quit and save the parameters. You can repeat the cursor mode read as necessary (:.read 50008.ctr).

      4. Place mouse cursor at center of a star and tap the Space Bar key. Repeat for additional stars, then go to next image. Raw instrumental magnitudes are placed in a file 50008.mag.1 or something similar. Print these out:

      cl> txdump *mag.1 image,xairmass,ifilter,otime,mag,merr
      After you have successfully performed aperture photometry on the images for a given night using either procedure, then **ARCHIVE THE REDUCED OBJECT IMAGES**.

II. ARCHIVING REDUCED IMAGES

1. Use the 'wfits' task to create "fits" image files. Unlike IRAF ".imh" image files, the ".fits" files can be transported from computer to computer:

      cl> wfits N*.imh fits=wN (where w stands for wfits; N = 1, 2, …9, a, …x1, x2, …Zero, Dark, and FlatV.

      This command line will take all IRAF images in the current subdirectory and create a sequence of "fits" files in the same directory with root names w1, w2, …w9, wa, …wx1, wx2, wZero, wDark, and wFlatV.

2. Next, you need to make a listing of the archived fits files:

      cl> rfits w*.fits make- iraf=u6 > u11028.ark

      The "make-" ensures that new IRAF images will not be created. "iraf=u6", for example, will give the original root name in the list.

3. Print the u11028.ark file for storage in the ARCHIVE binder.

4. Next, the files to be archived must be organized and moved. Use 'mkdir' to create a directory e.g. "ark" that is one level above all the "night" directories, "u1, u2, …u9", where your "fits" files are located. Make separate sub-directories below "ark" for each night, e.g. /u11028, /u11113, /u11209.

5. In addition to the "fits" reduced image files, also move all of the "coo.1" files, "mag.1" files, the uYMMDD.lis, .log, .sat, .red, .qwk, .ark and .mag files for archiving, and Night_Sum_Run. (Don't forget to archive any "x-files", i.e. saturated images, and "y-files", i.e. non-photometry images that were re-named with the pre-pended "x, y").

6. The following are examples of how to move your files:
      cl> move wN*.fits ../../ark/u11028   (Assuming your are in a "work" directory)
      cl> move u11028.* ../../ark/u11028
      cl> move *.coo.1 ../../ark/u11028
      cl> move *.mag.1 ../../ark/u11028
      cl> move Night_Sum_Run ../../ark/u11028

7. Use "chgrp rsp *.* (and *) to allow anyone in the group to read these files.

8. Notify me when files are ready for archiving. After I have done so, I will give you the okay to delete and imdelete all the files in a given directory, e.g. u6.

(Optional) CD Writing Procedure

1. Open the CD "burning" application by clicking on the **CD Burner** icon on your desktop.

2. Create a new data project by accessing the **File** menu and then the **New Project** menu. The screen will now be divided into two sections; the top of the window is the filesystem on the server, and the bottom of the window is the virtual filesystem of the yet-to-be-created CD. **Note**: Be sure that an "audio" window is not open and blocking the "data" window.

3. Find the archive directory that you created in step 4. Go into the archive directory and then click and drag all the directories into the right pane of the bottom half of the window.
Note: If the directory is large, you will need to wait until the application becomes responsive again before you continue.

4. Check the data size by looking at the bar immediately below the bottom pane. This bar shows both textually and graphically the current capacity of your selected directories. **Note**: If the size is over 700 MB, you have too much data for one CD. You need to move some of the contents elsewhere and put it on a second CD.
Once you have moved some files out of the directory, you can re-check the size by repeating this step.

5. Click on the "burn" icon on the tool bar. This will open a dialog window with the CD burning options. **Note**: The defaults should be okay, but it doesn't hurt to double check. Ensure that the write speed is set to **8X**, and that the following options are checked under the **Advanced** tab: **Generate rockridge entries**, **Generate joliet entries**, and **No deep directory relocation**. At this point, you can also change the name of the CD by clicking on the **Settings** tab and changing the **Volume name** field.

6. Begin burning the CD by clicking on the **Write** icon in the upper right hand corner of the pop-up dialog window. **Note**: The burning operation will be finished when the output window states that the writing has finished and the **Close** button appears at the bottom of the **Burn** dialog window.

7. Click on **Close** and then click on **Quit** from the File menu.

8. Test the CD in the AIAL PC. After you confirm that all the files have been successfully archived, be sure to clean up: Make sure that all *.fits, *.lis, *.log, *.red, *.sat, *.qwk, *.coo, *.mag, *.imh, and fs* files are deleted from the pc-astro directory that you were working in.

Some Useful Graphics Help Pages

Recommended "apphot" Parameter Sets

PACKAGE = apphot
  TASK = daofind

```
image        = ""            Input image(s)
output       = default       Output coordinate file(s) (default: image.coo.?)
(starmap     = "")           Output density enhancement image(s)
(skymap      = "")           Output sky image(s)
(datapars    = "")           Data dependent parameters
(findpars    = "")           Object detection parameters
(boundary    = nearest)      Boundary extension (constant|nearest|reflect|wrap)
(constant    = 0.)           Constant for boundary extension
(interactive = no)           Interactive mode ?
(verify      = yes)          Verify critical daofind parameters ?
(update      = yes)          Update critical daofind parameters ?
(verbose     = yes)          Print daofind messages ?
 (graphics   = stdgraph)     Graphics device
(display     = stdimage)     Display device
(icommands   = "")           Image cursor: [x y wcs] key [cmd]
(gcommands = "")             Graphics cursor: [x y wcs] key [cmd]
(mode        = ql)
```

```
PACKAGE = apphot
TASK = phot

Iimage         = ""                 Input image(s)
coords         = "default"          Input coordinate list(s) (default: image.coo.?)
output         = "default"          Output photometry file(s) (default: image.mag.?
skyfile        = ""                 Input sky value file(s)
(plotfile      = "")                Output plot metacode file
(datapars      = "")                Data dependent parameters
(centerpars    = "")                Centering parameters
(fitskypars    = "")                Sky fitting parameters
(photpars      = "")                Photometry parameters
(interactive   = no)                Interactive mode ?
(radplots      = no)                Plot the radial profiles?
(verify        = yes)               Verify critical phot parameters ?
(update        = yes)               Update critical phot parameters ?
(verbose       = yes)               Print phot messages ?
(graphics      = "stdgraph")        Graphics device
(display       = "stdimage")        Display device
(icommands     = "")                Image cursor: [x y wcs] key [cmd]
(gcommands     = "")                Graphics cursor: [x y wcs] key [cmd]
(mode          = "ql")
```

```
PACKAGE = apphot
TASK = centerpars

(calgorithm    =centroid)    Centering algorithm
(cbox          = 5.)         Centering box width in scale units
(cthreshold    = 0.)         Centering threshold in sigma above background
(minsnratio    = 1.)         Minimum signal-to-noise ratio for centering alg
(cmaxiter      = 10)         Maximum iterations for centering algorithm
(maxshift      = 1.)         Maximum center shift in scale units
(clean         = no)         Symmetry clean before centering
(rclean        = 1.)         Cleaning radius in scale units
(rclip         = 2.)         Clipping radius in scale units
(kclean        = 3.)         K-sigma rejection criterion in skysigma
(mkcenter      = no)         Mark the computed center
(mode          = "ql")
```

```
PACKAGE = apphot
TASK = datapars

(scale         = 1.)         Image scale in units per pixel
(fwhmpsf       = 5.)         FWHM of the PSF in scale units
(emission      = yes)        Features are positive ?
(sigma         = 1.)         Standard deviation of background in counts
(datamin       = -10.)       Minimum good data value
(datamax       = 65000.)     Maximum good data value
(noise         = "poisson")  Noise model
(ccdread       = "")         CCD readout noise image header keyword
(gain          = "")         CCD gain image header keyword
(readnoise     = 12.4)       CCD readout noise in electrons
(epadu         = 1.1)        Gain in electrons per count
(exposure      = "exptime")  Exposure time image header keyword
(airmass       = "title")    Airmass image header keyword
(filter        = "subset")   Filter image header keyword
(obstime       = "hjd")      Time of observation image header keyword
(itime         = INDEF)      Exposure time
(xairmass      = INDEF)      Airmass
(ifilter       = "INDEF")    Filter
(otime         = "INDEF")    Time of observation
(mode          = "ql")
```

```
PACKAGE = apphot
TASK = findpars

(threshold     = 4000.)        Threshold in sigma for feature detection
(nsigma        = 1.5)          Width of convolution kernel in sigma
(ratio         = 1.)           Ratio of minor to major axis of Gaussian kernel
(theta         = 0.)           Position angle of major axis of Gaussian kernel
(sharplo       = 0.2)          Lower bound on sharpness for feature detectio
(sharphi       = 1.)           Upper bound on sharpness for  feature detection
(roundlo       = -1.)          Lower bound on roundness for feature detection
(roundhi       = 1.)           Upper bound on roundness for feature detection
(mkdetections = no)            Mark detections on the image display ?
(mode          = "ql")


PACKAGE = apphot
TASK = fitskypars

(salgorithm    = "centroid")  Sky fitting algorithm
(annulus       = 15.)          Inner radius of sky annulus in scale units
(dannulus      = 3.)           Width of sky annulus in scale units
(skyvalue      = 0.)           User sky value
(smaxiter      = 10)           Maximum number of sky fitting iterations
(sloclip       = 0.)           Lower clipping factor in percent
(shiclip       = 0.)           Upper clipping factor in percent
(snreject      = 50)           Maximum number of sky fitting rejection iterations
(sloreject     = 3.)           Lower K-sigma rejection limit in sky sigma
(shireject     = 3.)           Upper K-sigma rejection limit in sky sigma
(khist         = 3.)           Half width of histogram in sky sigma
(binsize       = 0.1)          Binsize of histogram in sky sigma
(smooth        = no)           Boxcar smooth the histogram
(rgrow         = 0.)           Region growing radius in scale units
(mksky         = no)           Mark sky annuli on the display
(mode          = "ql")



        PACKAGE = apphot
          TASK = photpars

(weighting     = "constant")  Photometric weighting scheme
(apertures     = "8")          List of aperture radii in scale units
(zmag          = 15.)          Zero point of magnitude scale
(mkapert       = no)           Draw apertures on the display
(mode          = "ql")
```

"apphot" Help Pages

Retro-active Determination of HJD

This is the process used to modify headers of archived images in order to be able to use the IRAF task 'setjd' in "astutil". This requires the use of the task 'hedit' in order to add certain keywords required by 'setjd'.

  1. Load "astutil" if this is not done at login.
  2. Edit the parameter set for 'setjd':

(observa	= "obspars")
(exposur	= "exptime")
(jd	= "jd")
(hjd	= "hjd")
(ljd	= "ljd")

  3. Edit the parameter set for the 'observatory' task:

obsid	= "baker"
(observa	= "obspars")
(name	= "MSU Baker Observatory")
(longitu	= 93.041747)
(latitud	= 37.398626)
(altitud	= 424)
(timezon	= 6)
(overrid	= baker)

Next, the task 'hedit' must be used to add the required header keywords and values. The general usage of the task is:

      hedit images fields value (add=no) (delete=no)

The "ut", "dec", and "ra" keywords must have assigned values in order for 'setjd' to work. Also, four "fields" must be added to each image header; "date-obs", "exptime", "title", and "subset". The value for "date-obs" is the same for each header, and so might be "subset". Thus, these two can easily be added as follows:

      hedit n*.imh date-obs 25-7-95 add+
      hedit n*.imh subset VV add+

The exposure times will be different for each image, but if one value occurs more than others it is easiest to make them all the same and then come back in with 'ccdhedit' and change individual values as needed:

```
hedit n*.imh exptime 20 add+
ccdhed n10007,n10008,n10009 exptime 30
```

Note that the exposure times <u>must be in seconds</u>, **not** deciseconds!
For the titles:

```
hedit n*.imh title 950725 add+
ccdhed n10010, n10011,n10012 title "200102"
```