Basic Photometry

1 Pre-Processing

1.1 Image Formats

The first step in pre-processing is to ensure that the version of IRAF which you are using is capable of handling the image format produced by the FLI IMG1024-S CCD. It is usual to save the files in FITS (Flexible Image Transport System) format, but the Scarborough Telescope's CCD control program may not add the normal '.fits' extension to the filenames, so IRAF will have difficulty recognizing the files as FITS format. There are two ways to fix this problem:

1. Make IRAF format images from the fits images. Do this using the **rfits** command. This will produce versions of the input images in the native IRAF format. This format represents images as two files: a '.imh' file which contains the header data, and a '.pix' file which contains the raw pixels values in binary format. The image is always accessed and manipulated via the '.imh' file.

   *o*r

2. Rename the files so that the extension '.fits' appears at the end of each filename. This can be done by simply adding this extension to all existing filenames. For example, suppose you had a FITS format image named 'qso1.st8'. At a shell prompt (i.e., not in IRAF) simply type 'mv qso1.st8 qso1.st8.fits' to rename the image into something IRAF can handle without difficulty.

This second method is probably easiest, since you can keep everything in FITS instead of IMH format.

1.2 Constructing a Master Dark

The separate dark frames from the CCD can be combined directly to produce a master dark. Be sure to only combine the appropriate darks (i.e. same integration time and CCD temperature) for any single master dark. The most flexible image combination algorithms found in IRAF are in the **imcombine** task. Generally you will want to combine the darks with some form of positive outlier rejection, so as to exclude cosmic ray hits. A simple *median* combine often works well, as does an *average* combine in conjunction with the *crreject* outlier rejection algorithm. If you have only a few darks, the latter is probably a better choice. When combining the darks, be sure that the individual frames have a similar 'average' value (this value is robustly measured by the mode or median of the image, rather than the mean). As the darks are taken in a stable CCD configuration, at a fixed temperature, the average values should be basically the same[1] but you may need to slightly re-scale the images (as explained in the **imcombine** help file) to ensure the outlier rejection works properly.

1.3 Constructing a Master Flat

Next thing to do is combine the flats for each filter to produce master flat(s). Again, use the **imcombine** task[2]. Generally, you will not need to use any outlier rejection (just use an **average** combine with no rejection), as the flats are typically short enough that no cosmic ray hits are likely. You should however check the final image to ensure that no undesirable artifacts have slipped through. If you do need to turn on some sort of rejection algorithm, ensure that you also allow for re-scaling of the individual flat frames to ensure that the rejection works as desired.

---

[1] If the difference from image to image is greater than a few percent, then something is wrong!

[2] Note that the flats may have been dark-corrected at the telescope, so you might not need to do anything to them before combining the individual frames.

One more thing may need to be done to the master flat(s) before you can use them. The task used to apply the flats will check to see if they've been processed (which they haven't, at least as far as IRAF is concerned, if the flats were already dark-corrected at the telescope). So, you may need to make a fake header entry in the master flats to fool IRAF. The required fake header entry is named 'DARKCOR' and can have any value, so long as it is present. Make this modification using the **hedit** or **hfix** task, and note that you must explicitly instruct **hedit** if you want to add a header entry rather than modify an existing header entry.

1.4 Pre-Processing the Science Frames

The next step is to use the master dark(s) and flat(s) to dark current correct and flat field the science frames. Do this using the task **noao.imred.ccded.ccdproc**.

The parameters for **ccdproc** should be set with IMAGES equal to the names of the science frames you want to process, and DARK and FLAT set to the relevant master dark and master flat. Only DARKCOR and FLATCOR should be activated, and the CCDTYPE should be set to a blank. Be sure to use the correct master flat and dark if you are pre-processing a dataset with more than one filter or integration time.

2 Photometry

The basic goal of photometry is to deduce the flux of objects in an image. Generally, this is done in the measurement units of the image (typically ADU, expressed for convenience as an instrumental magnitude) and later converted to some useful physical unit (such as a calibrated apparent magnitude, which is related to flux in absolute physical units), at a later stage of the analysis. This section focuses on the first step, generating a measure of the instrumental magnitude.

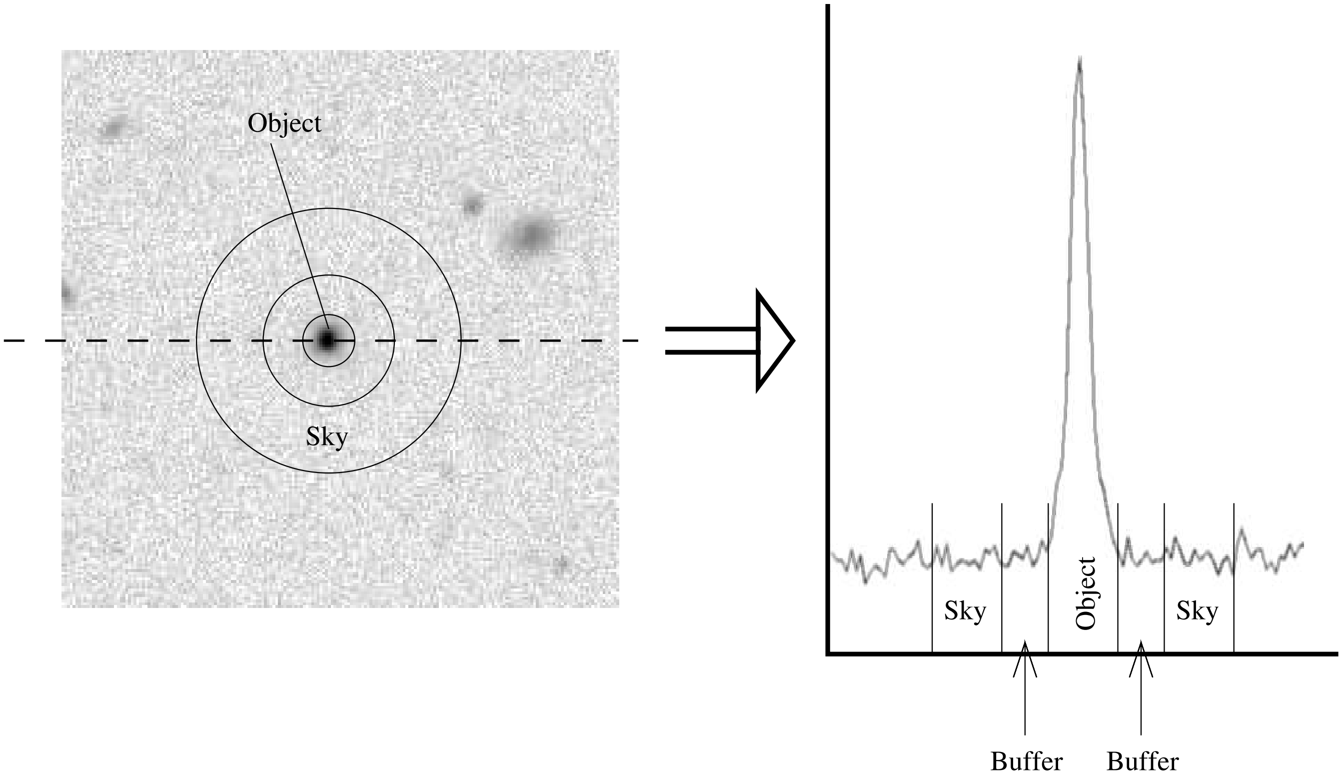

Figure 1: The basics of aperture photometry. Two apertures are defined, designated the 'object' and 'sky' apertures. The sky aperture is used to estimate and correct for the contribution of the background to the total counts in the object aperture. There is generally a buffer between the two apertures, primarily to ensure that the sky aperture is far enough away from the object to be representative.

The basic approach we will follow is referred to as aperture photometry. Other more specialized techniques such as PSF-fitting photometry will not be covered in this course. The basic process of aperture photometry is illustrated in Figure 1. Two apertures are defined. The first, the 'object' aperture, is centred on the object of interest and is a simple circular aperture large enough to encompass the bulk of the flux from the object. The second, the 'sky' aperture, is also centred on the object of interest, but is a ring (i.e. annulus) located at some distance from the object, far enough away to be relatively uncontaminated by flux from the object, but close enough and large enough to be representative of the typical sky value at the position of the object. The sky aperture is used to estimate the sky in the object aperture. Appropriate modification of the total counts in the object aperture hence results in an estimate of the counts which are due to the object alone. The appropriate sizes for both apertures have been a matter of some discussion in the literature, and the answers vary somewhat depending on what uncertainties are dominant in the image (e.g, are your errors mostly due to random noise?, or perhaps residual flat-fielding effects are more important...). For example, assuming a purely Poissonian noise, and a Gaussian Point-Spread Function (PSF) it is possible to compute the optimal object aperture size relative to the size of the PSF.

The IRAF package we will be using to do aperture photometry is **noao.digiphot.apphot**. It consists of a single master executable task: **phot**, as well as a number of parameter sets (all ending with an '@' symbol). The parameter sets each control the operation of a portion of the photometry process. These sub-components may also be run separately, though I recommend avoiding this in general. The sections below describe the process and justification for setting up the various parameters.

2.1 Object Centroiding

The **phot** task uses an input list of coordinates as locations around which aperture photometry is performed. The input list can be generated by other tasks; for example, you can use the logging functions in **imexamine**, or use the 'regions' generation in DS9, to generate such a list interactively. The process of generating a list of object locations is generally referred to as 'object finding' and requires sophisticated algorithms and processes when applied to typical large format images from CCDs on large telescopes. However, for our purposes, it will be sufficient to find objects by visual inspection.

Imaging data often comes in sets, where one has a number of images all of the same field, with slight offsets. The **phot** task is designed to handle just this sort of data. In this case, one defines a single list of object locations (typically by running some sort of object finding on a stack of all the sub-images) and then tweaks the positions of objects to represent the shifts on individual images. Re-centring is necessary, as centring errors can cause errors in the derived counts for objects, particularly if the apertures are relatively small, or the field rather crowded by many objects. This re-centring is controlled by the *centerpars* parameter set.

Several parameters in the *centerpars* parameter set should be modified to fit your data. First, CBOX should be set to several times the maximum shift, in pixels, that you expect between your input positions and the positions on any given image. MAXSHIFT should be exactly this maximum shift; objects whose re-centred positions are further than MAXSHIFT from the input positions are flagged in the final output. CTHRESHOLD controls the threshold above which pixels are considered when applying the centring; the default value of 0 (i,e, the sky level) is generally acceptable, though you may try modifying this value if the centring algorithm is poorly behaved. MINSNRATIO controls the level at which the centring will be performed; if the signal-to-noise (S/N) threshold interior to the centring box is below this level is then an error flag is raised in the output. The default value is again generally fine, but can be tweaked in some cases. The final parameter of concern is CALGORITHM, which controls the selection of the algorithm used to find the centres. Either the 'centroid' algorithm, which finds centres using moments, or the 'gaussian' algorithm, which fits a 2-D Gaussian PSF to the image, generally work fine. All other parameters in this parameter set can generally be left alone.

2.2 Finding the Sky Value

A critical part of aperture photometry is defining the sky values. For faint objects, the flux in even the object aperture may be dominated by background counts and so a correct computation of the background is critical. The parameter set used for controlling the sky-finding algorithm is *fitskypars*.

The parameters of primary interest are SALGORITHM, ANNULUS and DANNULUS. The latter two control the inner radius of the sky annulus, and its width, respectively. Note that DANNULUS is *not* the outer radius, it is rather the width of the annulus. The inner radius should be set so as to exclude any light due to the object. For point-like objects, a bare minimum is 3× the full width at half-maximum (FWHM) of the PSF, and larger may be more appropriate if the PSF has strong wings (or, of course, if the objects are extended). Signal-to-noise considerations dictate that the width of the annulus should be as large as reasonably possible, with competing considerations of contamination by other objects and large scale sky variations setting upper limits on the sky aperture size. SALGORITHM describes the process by which the pixels in the sky annulus are analyzed to make an estimate of the sky. A number of options are available. The simple *mean* should generally be avoided, as it is generally not robust if the distribution of pixel values is non-Gaussian (and it generally is, due to other objects and cosmic rays). A *median* or *modal* estimator is often a good choice, and a *centroid* or *gaussian* algorithm (which look in detail at the distribution of pixel values) will often produce reasonable results. The sky estimation routines also offer a number of rejection algorithms which attempt to recognize and exclude pixels which have values different than that expected from the overall distribution of sky pixels. A simple sigma-clipping is, for example, quite good at detecting and rejecting high pixel values due to cosmic rays and hot pixels. However, unless such pixels are a significant fraction of all the pixels in the sky annulus, such complex rejection schemes do no better than a simple *median* estimator.

2.3 Apertures et Cetera

A number of other parameters need to be set in order for **phot** to work properly. In the *datapars* parameter set, FWHMPSF should be set to the measured FWHM of the PSF of point sources in the images. This ensures that any algorithms which fit objects in the image (e.g. the 'gaussian' option for object finding) uses appropriately sized templates. EXPOSURE should be set to the name of the header item containing the integration time for the images.

In the *photpars* parameter set, you will also need to modify both the APERTURE and ZMAG parameters. The former is a list of object apertures. **phot** allows one to use multiple object apertures; it is generally a good idea to use quite a few (3 at minimum, and more than 10 in some cases) as an analysis of the resultant instrumental magnitudes and errors as a function of aperture radius can often reveal the 'best' aperture size to use for a particular set of images. Twice the FWHM of the PSF is a good starting size when no other information is available. When instrumental magnitudes are desired, ZMAG should be set to zero, so the the magnitude, m, reported by **phot** is simply

$$m \;=\; 2.5\ \log_{10}(C_{obj}) \tag{1}$$

where C_{obj} is the integrated number of counts (in ADU) computed for the object.

2.4 Running phot

Actually running **phot** is easy, once all the various parameter sets have been tweaked as described above. Within the parameters for **phot** itself, simply provide a list of images (IMAGE), the name of a file containing x,y pairs (1 per line) giving object coordinates (COORDS), turn the interactive and verify modes off (INTERACTIVE, VERIFY) and the verbose mode on (VERBOSE), and execute the task. A number of lines of photometry data, one per object per image, should then scroll by. The thing to watch for here is a large number of errors; if a significant fraction of the data are flagged with errors, then something is likely wrong in your setup, most probably in the centroiding. **phot** also produces a '.mag' file for each image. This contains the detailed data on computed magnitudes, magnitude uncertainties, photometry errors et cetera.

2.5 Interpreting the Results

The extraction of detailed data from the '.mag' files is made much easier by the **noao.digiphot.daophot.pdump** task. With this, you can extract various items from the output files, using various boolean logic operators. The help file of **pdump** is detailed and gives a number of examples of how to use it. Fundamentally, one is interested in extracting an instrumental magnitude, with an estimated uncertainty, for each object on each image. Most other data in the '.mag' files need only be considered if you are either trying to ascertain optimal parameters for the **phot** task, or are checking for unaccounted errors in the resultant data.

Instrumental magnitudes can only be used to compute differential magnitudes, in which objects on the same image are compared to one another. Even this simple process is only formally correct if the objects are of exactly the same colour, otherwise colour-term corrections should be applied if very accurate differential magnitudes are required. Most often, it is desirable to reference the instrumental magnitudes to some standard, physically motivated, measurement. The process of transforming instrumental magnitudes to zero-airmass colour-corrected calibrated apparent magnitudes (which measure exo-atmospheric fluxes in standard bandpasses) will be described later.