

The CFHT + WFC + UH8K Optical System

December 22, 1998

1 Distortion Map

In the UH8K camera the chips are laid out in two rows each of four 2K by 4K chips laid side by side. There are gaps of about 60 pixels between the sides of the chips and about 20 pixels between the two rows. However, the chips are not precisely laid out on the grid but are slightly rotated and shifted with respect to an ideal tiling grid. In addition to the somewhat irregular chip layout, CFHT suffers from a field distortion introduced by the telescope wide-field corrector; a pin-cushion type distortion with shift amplitude of about 40 pixels at the corner of our roughly 1/2 degree square field.

We have solved for the projection from chip coordinates onto the sky under the assumption that for each 2K×4K image there is some smooth mapping from pixel coordinates to the rectilinear sky coordinates, but we do not assume that there is any static relationship between the mapping from exposure to exposure — i.e we are assuming that the telescope/detector system remains rigid throughout an exposure, but can deform in a smooth but otherwise arbitrary manner between exposures. As we shall see, this gives a solution of much greater accuracy. With this model one can also combine data taken with different instruments.

We have solved for the parameters of this model — the ‘astrometric solution’ — by minimising the residuals in the predicted sky positions of a set of reference stars. In principle, given certain conditions on the geometry of the telescope pointings, one can perform an ‘internal’ solution using only the mosaic data in hand and without reference to any external astrometric system (one should then be able to tie this into any other chosen astrometric system by inspection of the coordinates of any pair of stars). However, with the rather small offsets between exposures employed here, this turns

out to be quite unstable; it is fairly easy to find solutions which map the star positions onto each other to great accuracy, but the solutions tend to have unacceptably large large-scale field distortions. What we have done instead is to incorporate the external astrometric information from the start.

The projection we have chosen is the ‘orthographic’ projection which is the particular stereographic projection of the sky illustrated in figure 1. This projection is shape (but not size) preserving, which is convenient for weak lensing studies; though had we chosen any of the other standard projections, the induced shape distortions would have been on the order of $\theta^2 \sim 10^{-5}$ and be negligibly small for the field size here, and similarly, the scale change across the field for the projection we have adopted is on the same order.

To solve for the image mapping parameters with the ms0302 data we used bright (but non-saturated) stars. We typically detected 100 or so stellar objects per image, or on the order of 1000 stars for a complete mosaic, and of these about one half were reliably measured in the astrometric reference catalogue (see below). Stellar centroids were measured fractional pixel precision using the interpolation scheme described in appendix 4. We solved simultaneously for a set of low order spatial polynomials, one per $2K \times 4K$ image, which map pixel coordinates onto the sky. The solution was obtained by a sequence of refined least squares minimisations. The relatively accurate CCD mosaic star positions ensure that the CFHT images map onto each other with high precision, while the external catalogue serves to damp down the kinds of artificial distortion one tends to obtain with a purely internal solution.

The external astrometric data base to which we have chosen to refer our coordinate system is the USNOA catalogue (ref??). Unfortunately there are rather few stars which are both present in the USNOA catalogue and are non-saturated in the CFH images. This seems to be because to be included in the USNA catalogue an object had to be detected in both red and blue passbands, so many stars which were actually detected in the red drop out of the final catalogue. To circumvent this problem we have extracted the digitized sky survey (ref??) red image covering our field and ran our own object detection algorithm which detects most of the USNOA objects, which are predominantly stars, and which were originally derived from the same Palomar plates, as well as a substantial number of stars which did not survive the USNOA selection criteria. In the final astrometric solution we use only fairly bright, but non-saturated, stars from the CFH images, and the overlap with the the augmented reference catalogue is such that we typically find

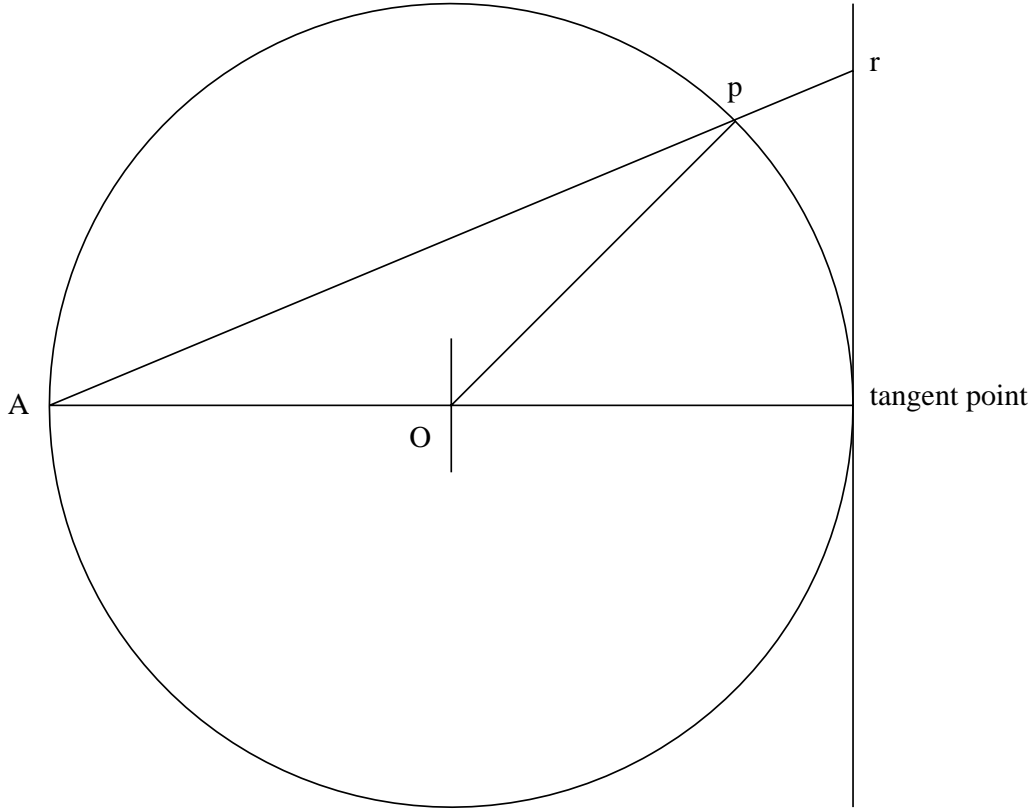


Figure 1: Illustration of the orthographic projection we have adopted where we have rotated the coordinate system so that the field center (α_0, δ_0) , or tangent point, lies along the x -axis. A point $p = (\alpha, \delta)$ on sky (represented here by the unit sphere) is projected onto the orthographic coordinate plane by projecting the line from the antipode of the tangent point through p into the tangent plane at point r . To fully define the r coordinate system it is necessary to specify the celestial coordinates of the field center (α_0, δ_0) , a rotation angle giving the orientation of the axes of r -coordinate system relative to the latitude and longitude unit vectors at the field center, and a scale factor.

about 50 stars in common per $2K \times 4K$ CFH image.

We modeled the mapping from pixel coordinates x_{pi} (this being the position of the p 'th star on the i 'th image) to $r = r_{\text{USNOA}}$ coordinates as a similar low order spatial polynomial:

$$r_p = x_{pi} + \sum_{l,m} a_{ilm} f_{lm}(x_{pi}) + e_{pi} \quad (1)$$

where e_{pi} is the observational error, which we assume to have an isotropical 2-D gaussian pdf with scale length σ_{pi} , whereas for the reference catalogue (the DSS catalogue, corrected as described above, and to which we ascribe the index $i = 0$)

$$r_p = x_{p0} + e_{p0} \quad (2)$$

or equivalently we can say that (1) applies for all i with the understanding that $a_{0lm} = 0$. This mapping is illustrated in figure 2.

Our astrometric solution is that set of star positions r_p and parameters a_{ilm} which minimise the sum of the squared residuals:

$$\chi^2 = \sum_p \sum_i e_{pi}^2 / \sigma_{pi}^2 = \sum_p \sum_i (r_p - x_{pi} - \sum_{l,m} a_{ilm} f_{lm}(x_{pi}))^2 / \sigma_{pi}^2 \quad (3)$$

which is quadratic in the parameters a_{ilm} , f_p . Provided we have chosen suitable pointings this yields the well conditioned set of linear equations

$$\partial\chi^2/\partial a_{ilm} = 0; \quad \partial\chi^2/\partial r_p = 0 \quad (4)$$

which we solve by LU decomposition for the mode amplitudes a_{ilm} and star positions r_p .

As an objective check on how well this procedure worked, we withheld a random subset of 20% of the stars, and did not use these in the registration solution. After solving for the image mapping, we applied the solution to these stars and measured how well their r -coordinates agreed. Typical results are shown in figure 3. This exercise gave a rms displacement of about 7 milliarcsec, or about 1/30 of a pixel, for the rms separation (one component) corresponding to a 1-particle rms of about 5 milliarcsec. With this degree of accuracy, any artificial shape distortion due to inaccuracy of the image mapping is, at worst, on the order of $(\delta\theta/\theta_{\text{obj}})^2$, where $\delta\theta$ is typical error in the warping solution and θ_{obj} is the size of the objects, the and is entirely negligible.



Figure 2: Illustration of the mapping from CFHT image pixel coordinates and reference catalogue coordinates onto the sky. The orthographic coordinates from the reference catalogue map directly onto r -space with no distortion (but considerable measurement error). Each of the CFHT science images maps onto the plane in a distorted manner, shown grossly exaggerated in the figure, and which we describe by a low-order spatial polynomial as described in the text. Having solved simultaneously for the relationship between r -coordinates and x -coordinates for all of the images we warp each image to generate an r -space image in a somewhat oversized rectangular ‘bounding box’ as shown by the dotted box. Once these are generated we stack them and average to produce the final image.

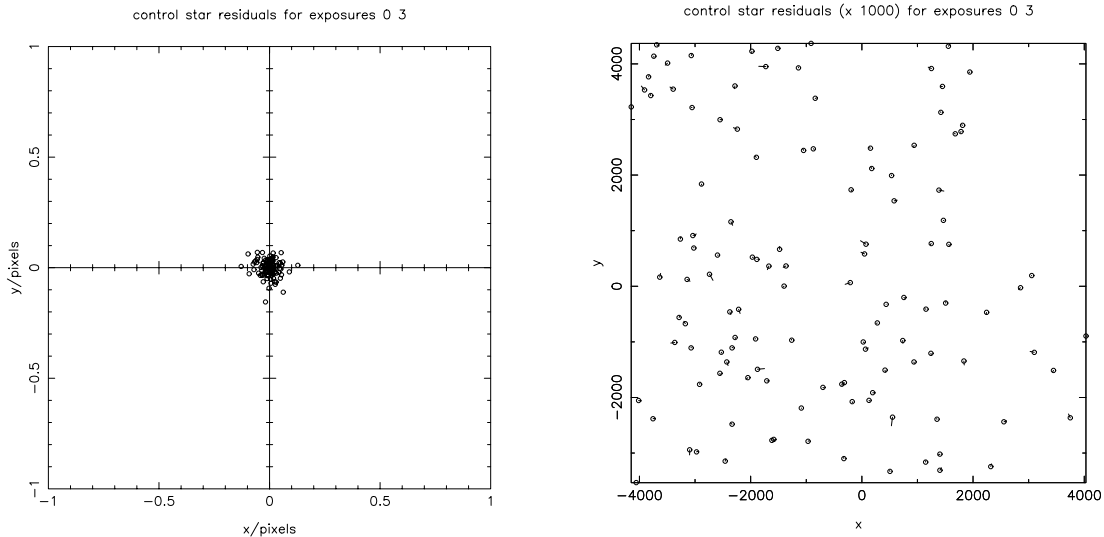


Figure 3: Residuals in the image mapping for the control sample. Left hand panel shows the differences between model r -values for a set of control stars — those stars not used in the solution — between a typical pair of exposures. The unit of length here is the pixel size in the source images or $0.207''$. The right hand shows the same residuals, now plotted as vectors with the base of the vector placed at the object position and with the length of the vector exaggerated by a factor 1000. These plots show that the mapping is very accurate, with residual pairwise separation with rms of 7 milliarcsec (per component), corresponding to a 1-particle positional uncertainty (smaller by a factor $1/\sqrt{2}$) of about 5 milliarcsec. The residuals appear to be approximately Gaussian distributed, and we see no obvious systematic variation of the residuals over the image; we find no tendency for unusually large residuals along the ‘overlap’ regions between the chips, for example.

The transformation coefficients a_{ilm} we have thereby obtained give the mapping from x_i -space to r -space (i.e. it gives $r = r(x_i)$ as an explicit function of x_i). For actually warping the images what is more useful is the inverse mapping $x_i = x_i(r)$, since we need to compute, for each pixel in a target image defined in r -space, what is the image of the pixel center in x_i -space, so we need x_i as an explicit function of r . (Actually one can perform the image warping using the forward transformation - see, for example, the `warpimage` man page - but it turns out to be relatively expensive in terms of computational effort). To obtain the inverse transformation we generated a regular grid of points which span the $2K \times 4K$ region of x_i -space occupied by the real pixels, applied the forward transformation to compute the model r -values and then fed these x_i, r value pairs to our least-squares program to solve now for the inverse mapping

$$x_i = r + \sum_{l,m} a'_{ilm} f_{lm}(r) \quad (5)$$

to obtain the coefficients a'_{ilm} .

We are now almost ready to actually apply this solution to generate a set of warped images. First however, we solved, again in a least squares manner, for a set of magnitude offsets (one set for the chips and one set for the exposures) which account for any variation in sensitivity between chips and for varying extinction between exposures. These corrections were very small; typically ~ 0.01 magnitudes with maximum correction of $\simeq 0.04$ magnitudes, so we can conclude that the observing conditions were accurately photometric.

2 Point Spread Function

The seeing in the final MS0302 images is very good, (FWHM $\simeq 0.6''$) which means that departures from the pure circular psf expected from atmospheric turbulence become very noticeable. These departures from circularity in the psf most likely stem from several sources: guiding errors, charge-transfer effects, and aberrations of the telescope optics. In the latter category there is a well known astigmatism, thought to be caused by an imperfection in the primary mirror figure. This is readily seen as a variation in the psf ellipticity as one moves the camera up and down through focus. Consequently, this effect couples to any tilting of the chips relative to the focal plane. The result of this is a variation of the psf which varies smoothly across any one

chip, but which jumps discontinuously as one passes to a neighbouring chip.

This is something of a nuisance, particularly in weak lensing studies, where we need to accurately model and correct for variations in the shape parameters due to seeing. In the past we have modelled the psf anisotropy as a smooth low-order polynomial. This is clearly inappropriate here since the psf shape parameters for the summed images will have a step-like variation pattern.

In weak lensing, we measure the weighted quadrupole moments of the objects $q_\alpha = M_{\alpha lm} \int d^2r r_l r_m w(r) f(r) / \int d^2r f$, from which we can form an estimate of the shear γ_α . Under the assumption that the non-circular psf can be modeled as the convolution of the circular psf from atmospheric seeing with a compact ‘anisotropising kernel’ $h(r)$, the response of the non-circularity on the quadrupole moments of galaxies is simply $q_\alpha \rightarrow q'_\alpha = q_\alpha + R_{\alpha\beta} p_\beta$ where the ‘response function’ $R_{\alpha\beta}$ is, like q_α a simple linear function of f , and p_β is, it turns out, just the (unweighted) quadrupole moment of the kernel h .

This means that the effect of the psf anisotropy is encoded in the 2 component entity p_β , which can be inferred from the $q_\alpha, R_{\alpha\beta}$ values of stars as $p_\alpha = R_{\alpha\beta}^{-1} q_\beta$.

For a galaxy which happens to have a star nearby, we could simply measure q and R for that star from the final averaged image, and correct the galaxy q value using the above relations to what would have been observed with a telescope with a circular psf.. In general, we need to generate a model for $p = q^*/R^*$. As mentioned, we expect p to be smoothly varying across each (warped) source image, so we could model this as a low order spatial polynomial $p_\alpha(r) = \sum_{l,m} p_{\alpha lm} f_{lm}(r)$, and we could then average these over all the warped images which contribute to the final averaged image to synthesise a model for the effective psf polarisation. However, since p is the ratio of q and R , and is therefore non-linear in the flux f , this would not be quite correct; rather we should generate separate models for q^*, R^* , which are linear in f , average these over contributing images, and then form the ratio of the averages. The result of this, for a grid of points spanning our final image is shown in figure 4.

A comparison of q values measured from the actual psf for the summed images and the synthesised models is shown in figure 5.

For some purposes the reduced description of the psf shape in terms of the 2 numbers p_α is not adequate and we would rather have a model for the full 2-D psf $g(x = r - r_0; r_0)$, this being the shape of a star whose centroid

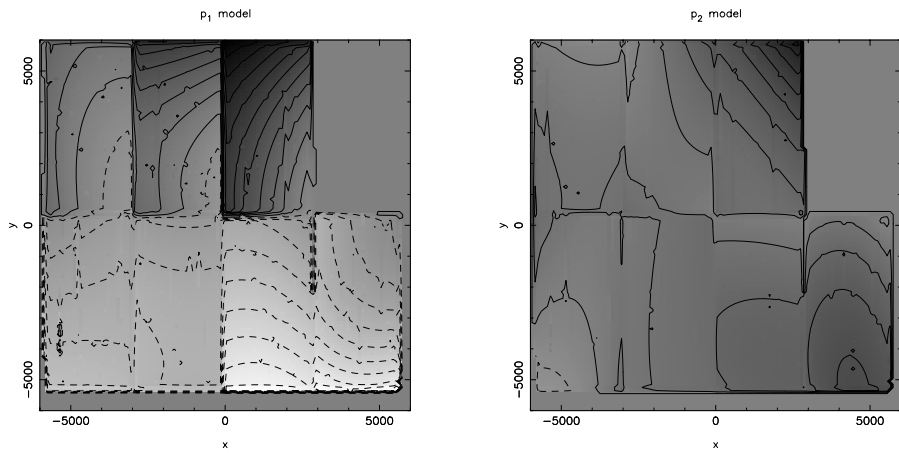


Figure 4: Contour plots of the 2 components of the psf polarisation computed as described in the text. Results for the I-band image are shown. The model used here was quadratic: $l_{\max} = 2$. The high frequency features within the chip arise from the masking of cosmetic defects on the chip (which causes the set of contributing images to change discontinuously in these regions). The psf anisotropy is primarily in the p_1 component. The figures clearly show strong discontinuities at the inter-chip boundaries (at higher resolution these transitions become staircase-like).

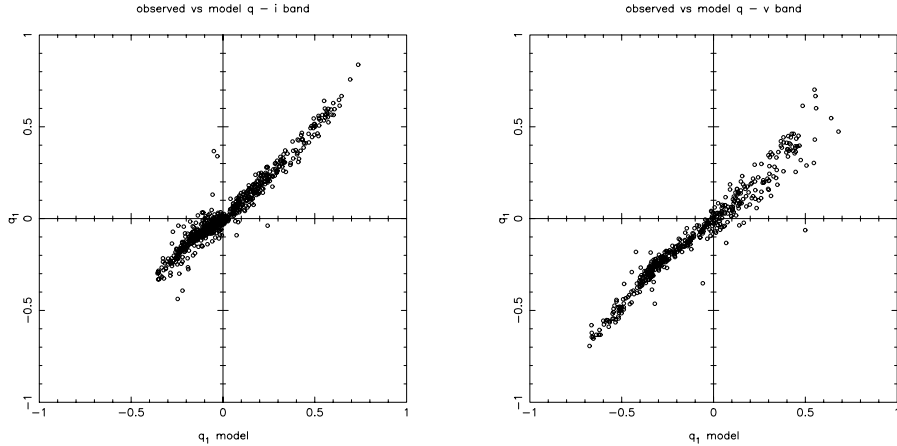


Figure 5: Comparison of model quadrupole moments and actual moments for stars measured from the final summed images.

lies at position r_0 . We can model this in an analogous manner. For each warped image we extract a set of ‘postage stamp’ images of the stars, and model these as

$$g(x; r_0) = \sum_{l,m} g_{lm}(x) f_{lm}(r_0) \quad (6)$$

then, for any point on the final image plane, we can synthesize a set of psf’s, one for each contributing image, and then average these. A set of such synthetic psf’s is shown in figures 6, 7. These figures show that psf does not vary dramatically in size across the $1/2$ degree field, and that departures from circularity are not very large. As with the p_α contour plot, these figures show that the chip-to-chip variation dominates over classical aberration effects (which are expected to produce a circularly symmetric variation).

Provided with our on-line database are a much finer grid of synthetic psf images for each passband, and a script which can generate the psf at an arbitrary position on the field.

The detailed model for the psf is potentially very useful in the detection of moving or time-varying objects. Given these data and a similar observation taken at a different epoch, one can generate two images with identical psf’s by convolving each image with the psf from the other epoch.

l-band psf

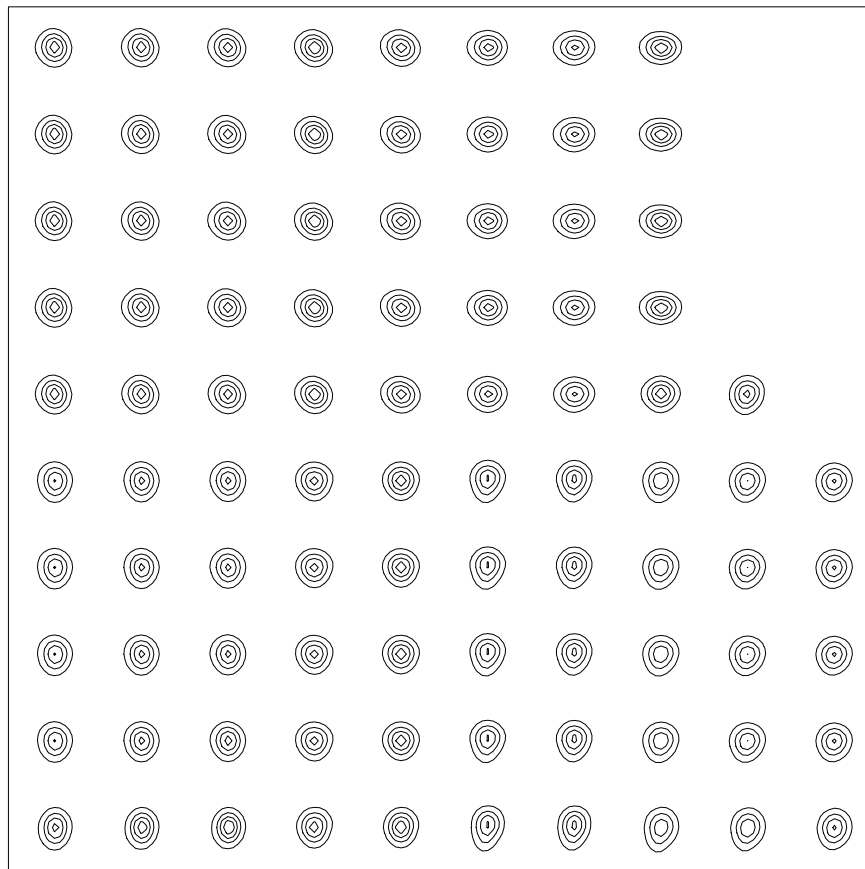


Figure 6: Grid of synthetic stellar psf models computed as described in the text. Each postage stamp image is 16×16 pixels in size. The contours are linearly spaced.

V-band psf

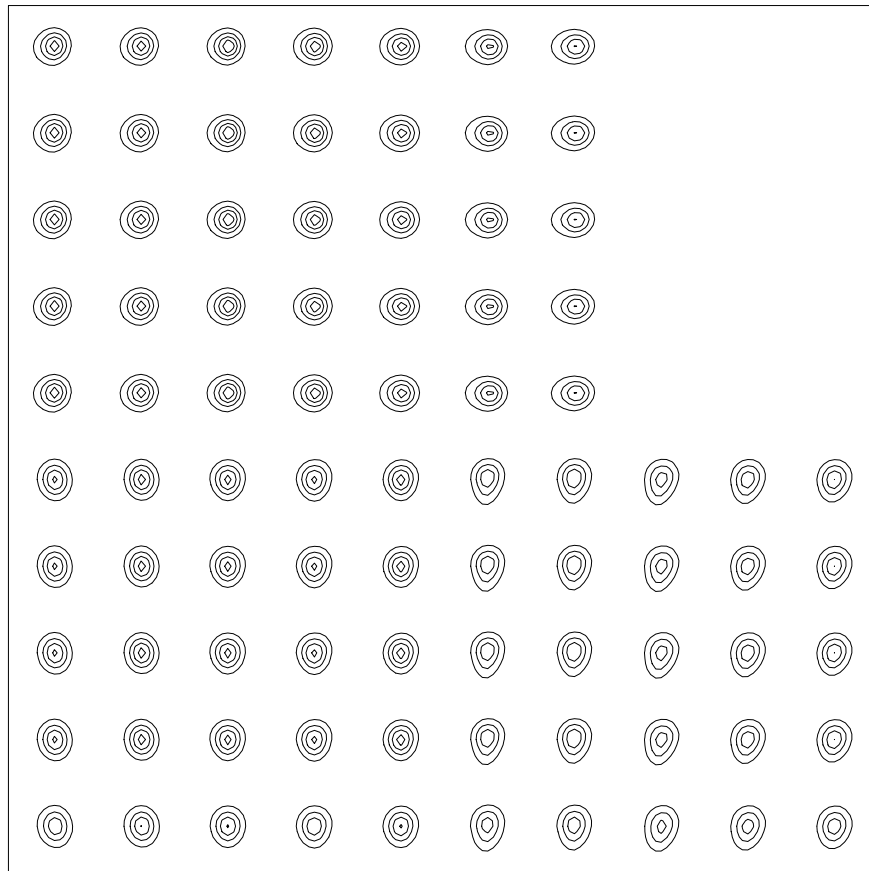


Figure 7: As figure 6, but now for the V-band images.

3 Cross-correlation Registration

The registration process described in §1 is fully automated. An essential ingredient is the algorithm for approximately registering a pair of catalogues. The algorithm only seeks to find a scaling, rotation and displacement that effects the transformation from one coordinate system to the other. While crude, this may at least allow one to merge or match the catalogues — i.e. find pairs whose coordinates agree to some small tolerance — and once the correspondence between objects in the different catalogues is established one can readily solve for more elaborate transformation models (such as the low order spatial polynomial models used extensively here).

The problem is illustrated in the figure 8. The solution we use is conceptually very simple; if for each catalogue we make a larger catalogue containing all pairs of objects from the source catalogue, and then plot these pairs in $\phi, \log d$ space (where ϕ is the orientation angle of the pair and d is the pair separation) then the two catalogues should just be shifted with respect to each other, with the shift in ϕ simply being the rotation between the two coordinate systems, and the shift in $\log d$ being the logarithm of the scale factor. This is illustrated 9.

These rotation and scale factors can readily be determined by autocorrelation using the FFT. We simply generate a pair of images of the counts of pairs in $(\phi, \log d)$ space and compute their cross-correlation, which shows a strong peak at the location of the real shift. In fact we wrap the $\log d$ coordinate.

Once we have the scale and rotation, we can apply these to the first catalogue. This should now simply be a shifted version of the second catalogue — but now shifted in regular x -coordinate space — and we can solve for this by again making and cross-correlating a pair of images of the counts of the objects.

Even if the input catalogues satisfy the scale, rotation and translation transformation exactly, the result of `acfregister` will be somewhat imprecise due to the finite pixel size in the images used here.

This algorithm works quite well with real data, provided there is a reasonable overlap between the objects in the two input catalogues, and will still usually generate an acceptable solution even if a substantial fraction of the objects in one catalogue are missing in the other.

We have found that the algorithm can become confused with certain pairs of input catalogues — because of the periodic boundary conditions implied

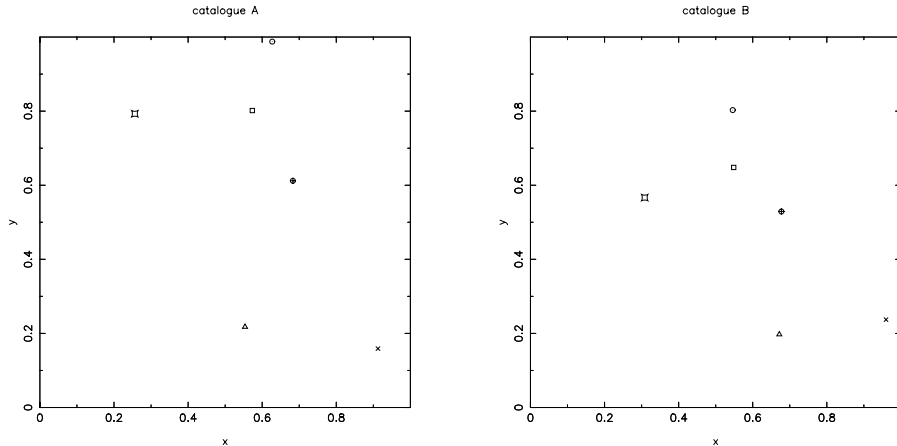


Figure 8: The two panels show the same set of 6 objects, but with coordinates in catalogue-B being scaled (by a factor 0.8), rotated (by 0.3 radians) and shifted (by $(0.3, -0.1)$) with respect to catalogue-A.

by the use of the FFT, it can, for instance, give a solution which is an alias of the desired solution (with the shift incorrect by the side of the box say). We are working on avoiding such problems.

4 Two-Dimensional Interpolation

Here we describe our sub-pixel interpolation scheme. We use interpolation at two points in the process: first when we determine the centroids of stars and second when we warp the images. The interpolation schemes for these two steps are different.

The object detection algorithm we have used for detecting stars for registration is very simple; we simply smooth the image f with a gaussian smoothing kernel with size similar to the psf and then find the local maxima of the smoothed flux F to obtain a integer pixel position (i_x, i_y) . To refine this we compute first derivatives

$$\begin{aligned} F_x &= (F_{i_x+1, i_y} - F_{i_x-1, i_y})/2 \\ F_y &= (F_{i_x, i_y+1} - F_{i_x, i_y-1})/2 \end{aligned} \tag{7}$$

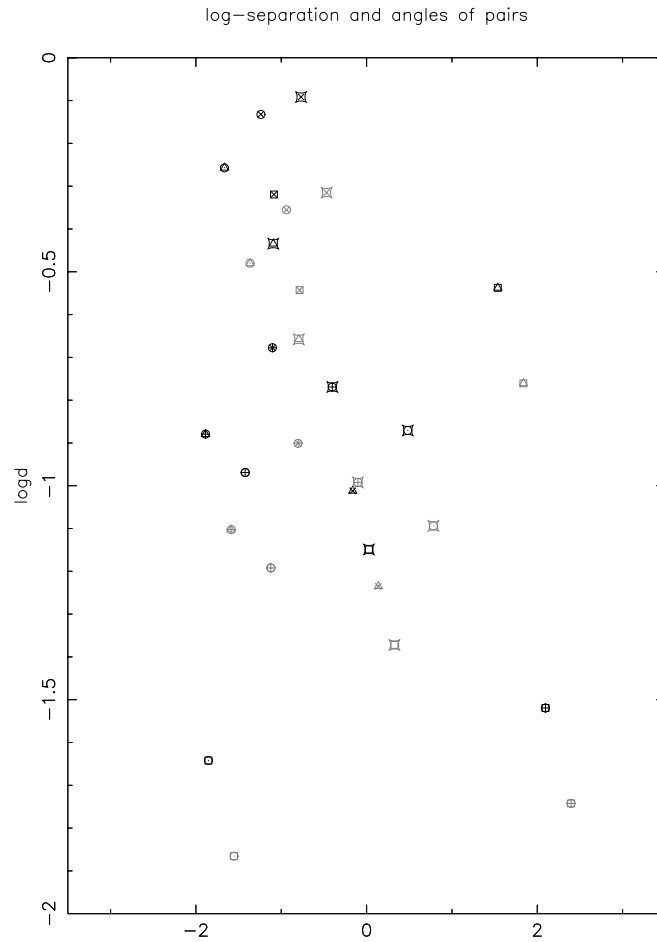


Figure 9: Plot of pairs from the each of the catalogues in orientation vs log separation space (with the pairs from catalogue-B shaded). It can easily be shown, and is readily apparent from the figure, that the pairs are simply shifted with respect to each other.

and second derivatives

$$\begin{aligned} F_{xx} &= F_{i_x+1, i_y} - 2F_{i_x, i_y} + F_{i_x-1, i_y} \\ F_{yy} &= F_{i_x, i_y+1} - 2F_{i_x, i_y} + F_{i_x, i_y-1} \end{aligned} \quad (8)$$

We then compute refined positions

$$\begin{aligned} x &= i_x - F_x/F_{xx} \\ y &= i_y - F_y/F_{yy} \end{aligned} \quad (9)$$

Our pixel coordinate labeling convention differs from the FITS standard in which pixel centers have integer values with FORTRAN style unit offset indexing — so the physical region covered by a $N \times N$ chip is defined to be $0.5 < x, y < N + 0.5$. Here our pixel centers have half-integer values and we use the ‘c’-style zero offset indexing so we have $0 < x, y < N$, and the centre of the corner pixel, for instance, is $(x, y) = (0.5, 0.5)$.

When we warp the images we use a different type of interpolation. We think of our pixel values as samples of some continuous function f (being the convolution of the sky flux with some boxy pixel response function) on a grid points $(i_x + 1/2, i_y + 1/2)$. A point (x, y) lies within a square defined by four of these samples. To interpolate the f value we add a fifth sample at the centre of the square which is the average of the four corner values. Joining the four corners to the center divides the square into four triangles, with the interpolation point lying in one of these. We take as our estimate of $f(x, y)$ the height at (x, y) of the plane which passes through the three samples at the corners of the triangle. The interpolated f values thus obtained are continuous, but have discontinuity of gradient along the vertical, horizontal and diagonal lines connecting the original sample grid.

5 Catalogue Format

Along with the summed images, our database also contains catalogues of objects and numerous auxilliary tabular information. The format of these catalogues and other tables is defined by the program ‘`lc`’ (for `list catalogue`) which, as its name implies, in its default mode, simply generates an ascii format listing of a catalogue. However, with its numerous command line options, `lc` becomes a fairly versatile filter for manipulating catalogues.

Our `lc` program is very similar to ‘`awk`’, in that it processes catalogues one object at a time; reading an object from standard input - performing

some manipulation on the contents of the object as specified by instructions supplied on the command line - and writing the result to standard output, but with the following distinctions: a) fields or entries in the object are referred to by symbolic names rather than by column number; b) entries may be scalars, vectors or matrices of arbitrary rank (there is also some limited support for textual entries) and c) `lc` can read and write in binary format, resulting in a large gain in efficiency.

Operations to be performed are specified as command line strings in a simple ‘reverse-Polish’ notation. All of the standard *c* math library functions and operators as well as a number of specialised functions and operators (such as vector products, matrix inversion etc.) are supported.

Our photometry packages are implemented as ‘filters’ which read `lc`-format catalogues and add size, magnitude, shape etc information consecutively. All the auxiliary files used in the reduction process described here are stored as `lc`-format catalogues for ease of manipulation.

Users of our database will find `lc` useful for extracting variables of interest from our catalogues and for applying selection criteria to select subcatalogues.