

# SDSS Image Processing II: The *Photo* Pipelines

Robert H. Lupton<sup>1</sup>

Željko Ivezić<sup>2</sup>

James E. Gunn<sup>1</sup>

Gillian Knapp<sup>1</sup>

Michael A. Strauss<sup>1</sup>

Michael Richmond, Nan Ellman, Heidi Newburg, Xiaohui Fan, Naoki Yasuda, Masataka Fukugita, Kazu Shimasaku, Marc Postman, Andy Connolly, David Johnston, Tom Quinn, Connie Rockosi, Brian Yanny, Masataka Fukugita, Steve Kent, Chris Stoughton, Don Petravick

Lynda M.Lee<sup>3</sup>

Pamela Ivezić<sup>4</sup>

<sup>1</sup>*Princeton University Observatory, Princeton, NJ 08544, U.S.A.*

<sup>2</sup>*Department of Astronomy, University of Washington, Seattle, WA 98195;  
ivezic@astro.washington.edu*

<sup>3</sup>*University League Nursery School, Princeton, NJ 08544, U.S.A.*

<sup>4</sup>*2555 NE 85th Street, Seattle, WA 98115, U.S.A.*

## ABSTRACT

A description of the SDSS image processing package, *Photo*.

## Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Book-keeping information; Flags and Masks . . . . .	8
1.2	Choice of data type . . . . .	11
<b>2</b>	<b>The Point Spread Function</b>	<b>11</b>
2.1	Is SDSS Data Band Limited? . . . . .	12

2.2	Modelling the Point Spread Function . . . . .	12
2.3	The Selection of PSF Stars . . . . .	12
2.4	KL Expansion of the PSF . . . . .	14
2.4.1	Application to SDSS data . . . . .	14
2.4.2	Gauging the Success of the KL Expansion . . . . .	16
2.5	Other Uses for Point Sources Detected by the <i>PSP</i> . . . . .	17
<b>3</b>	<b>Determining Bias Vectors, Scattered Light, and Flatfields</b>	<b>17</b>
3.1	The Scattered Light Correction and Flatfield Determination . . . . .	17
3.2	External Determination of Flatfield Vectors . . . . .	19
3.3	Flatfield “Seasons” . . . . .	19
<b>4</b>	<b>Removing the Instrumental Signature</b>	<b>19</b>
4.1	Flat Fielding, Bias Subtraction, and Linearization . . . . .	22
4.2	Interpolation over CCD Defects and Bleed Trails . . . . .	22
4.2.1	Application to Seeing-convolved Data . . . . .	24
4.3	Cosmic Ray Rejection . . . . .	25
4.3.1	Cosmic Ray Rejection: pass I . . . . .	29
4.4	Serial register artifacts . . . . .	30
<b>5</b>	<b>Sky Subtraction and Object Detection</b>	<b>31</b>
5.1	Sky Subtraction . . . . .	31
5.2	Object Detection . . . . .	32
<b>6</b>	<b>Determining the Centres of Objects</b>	<b>33</b>
6.1	Introduction . . . . .	33
6.2	Astrometric Centering: Gaussian Quartic Interpolation Schemes . . . . .	34
6.2.1	Finding an Object’s Center in One Dimension . . . . .	34
6.2.2	Extension to Two Dimensions . . . . .	36
6.2.3	Error Estimates . . . . .	36

Objects much fainter than the sky . . . . .	37
Objects brighter than the Sky . . . . .	39
6.2.4 Monte-Carlo Simulations . . . . .	39
6.3 Correcting for non-Gaussian, Asymmetrical, PSFs . . . . .	39
<b>7 Merging per-band detections</b>	<b>42</b>
<b>8 Deblending Overlapping Objects</b>	<b>42</b>
<b>9 Extracting Radial Profiles</b>	<b>42</b>
9.1 Photometry . . . . .	45
9.2 Measuring Surface Brightnesses . . . . .	45
9.3 Radial Profiles . . . . .	46
<b>10 <i>Photo</i>'s Flux Measures</b>	<b>47</b>
10.1 Fiber Magnitudes . . . . .	47
10.2 The Calculation of Petrosian Quantities by <i>Photo</i> . . . . .	47
10.2.1 Errors in Petrosian Quantities . . . . .	47
10.3 PSF magnitudes . . . . .	48
10.3.1 Introduction . . . . .	48
10.3.2 Calibrating TDI Data . . . . .	48
10.3.3 PSF Magnitudes in <i>Photo</i> . . . . .	49
10.4 Fitting Models to Detected Objects . . . . .	50
10.4.1 Model Fitting . . . . .	50
10.4.2 Model Magnitudes . . . . .	53
Composite Model Magnitudes . . . . .	53
10.5 Saturated Objects . . . . .	54
<b>11 Object Classification</b>	<b>54</b>
11.1 Cosmic Ray Rejection: pass II . . . . .	54

<b>12 Measurements of Objects Shapes</b>	<b>55</b>
12.1 ‘Stokes’ Parameters . . . . .	55
12.2 Adaptive Shape Measures . . . . .	56
12.3 Isophotal Measures . . . . .	56
<b>13 <i>Photo</i>’s Outputs and Performance</b>	<b>56</b>
<b>14 Testing and Quality Assurance (QA)</b>	<b>56</b>
<b>15 Future Directions</b>	<b>56</b>
<b>16 Conclusions</b>	<b>56</b>
<b>A Input Parameters</b>	<b>57</b>
A.1 Parameters Describing the Exposures, Camera, and Electronics . . . . .	57
A.2 Parameters which Control Data Processing . . . . .	58
A.3 Parameters which Control Algorithms used in Processing the Data . . . . .	58
<b>B Determining Bad Columns in the SDSS Camera</b>	<b>58</b>
<b>C Image Formation</b>	<b>58</b>
<b>D The Calculation of the Centroid of Properly Sampled Data</b>	<b>59</b>
<b>E Übercalibration of SDSS Data</b>	<b>60</b>
<b>F Some Statistical Properties of Poisson Distributions</b>	<b>60</b>
<b>G Reconstructing the KL-PSF given a psField file</b>	<b>60</b>
G.1 Interpretation of PSP status codes . . . . .	61

## 1. Introduction

The Sloan Digital Sky Survey (SDSS) is an imaging and spectroscopic survey of the sky (York et al. 2000) using a dedicated wide-field 2.5m telescope (Gunn et al. 2005) at Apache Point Observatory, New Mexico. Imaging is carried out in Time Delay Integrate (TDI; also known as drift-scan) mode using a dedicated camera containing 30  $2048 \times 2048$  SITe CCDs (Gunn et al. 1998) which gathers data in five broad bands,  $ugriz$ , spanning the range from 300 to 1000 nm (Fukugita et al. 2004), on moonless photometric (Hogg et al. 2001) nights of good seeing. The images are processed using the specialized software, *Photo*, described in this paper, and are astrometrically calibrated (Pier et al. 2003) using the UCAC catalog (Zacharias et al. 2000), and photometrically (Tucker et al. 2005) calibrated using observations of a set of primary standard stars (Smith et al. 2002) on a neighboring 20-inch telescope (see also Appendix E). The median seeing of the imaging data is 1.4 arcsec in the  $r$  band, and the 95% completeness limit in the  $r$  band is 22.2. An overview of the SDSS data processing is given in Stoughton et al. (2002).

The SDSS camera generates data at about 4.6Mby/s, covering about  $98.9 \text{ deg}^2/\text{hr}$  (but only  $19.8 \text{ deg}^2/\text{hr}$  per band) with an exposure time per band of about 55s. The camera’s footprint consists of 6 long strips, each 13.5 arcmin wide and up to  $120^\circ$  long. The sky is scanned past the filters in the order  $r, i, u, z, g^1$ , taking 4.9 minutes to pass from the center of the  $r$  to the center of the  $g$  detector. These strips are cut into a set of *frames* of size  $2048 \times 1361^2$  pixels (the plate scale is  $0.396 \text{ arcsec/pixel}$ , so a frame is  $13.5 \times 9 \text{ arcmin}^2$ ), and the frames taken of the same patch of sky in the five bands are assembled to form a *frame*.

This paper describes the SDSS pipelines *PSP* and *Frames*, collectively referred to as *Photo*. The *SSC* pipeline runs before *PSP* to pre-process the raw data into a more convenient form, and to provide lists of objects for *PSP*. *PSP* is responsible for all tasks that require more than a single field of data (e.g. scattered light corrections and PSF determination). We then run the astrometric pipeline, *Astrom* (Pier et al. 2003), to provide absolute, and, more importantly in this context, relative astrometry between objects detected in the 5 SDSS photometric bands. We then run *Frames*, which removes the instrumental signature; detects objects in each band and merges the detections into objects; deblends the objects, and produces a catalog of objects with well-measured parameters, in instrumental units (pixels; data-numbers; angles relative to the scan direction). Finally, additional pipelines (Tucker et al. 2005) are run after *Photo* has finished to provide calibrated quantities (e.g. magnitudes; positions in  $(\alpha, \delta)$ , and angles East of North).

(**XXX** *Do we want a pretty colour picture of a piece of the sky?*)

---

<sup>1</sup>Mnemonic: *Robert Is Under Ze Gunn* (Pier 1998)

<sup>2</sup>The strange number 1361 is set by the physical layout of the SDSS camera; the distance between the serial registers of two successive CCDs is 65.3mm, which corresponds to  $2 \times 1361$  pixels

In the early 1990s, when the SDSS was conceived, the data rate of 16 Gb/hr was intimidating<sup>3</sup> and the total data volumes were far beyond the disk capacity available even at large computing facilities. We accordingly spent considerable effort optimising the SDSS data processing in ways which would not make sense if we were starting the task today; in particular we took pains to minimise the total number of times that we’d process each pixel; the number of floating-point operations per pixel, and the total memory footprint of the codes. Our goal was to be able to process all of the imaging data within a lunation, and we thought that the *Frames* processing would dominate the compute budget. Figure Fig. 1 shows the performance of *Frames* on about 400s of data from one dewar on a single 3GHz Xeon processor. It will be seen that we’d be able to keep up with the *Frames* processing all of our data in realtime using only 8 processors. In reality, the *PSP* processing takes a similar time, but we are still easily able to keep up with the data flow using very modest computing resources.<sup>4</sup>

All of the SDSS imaging software<sup>5</sup> is built using a infrastructure package called *Dervish* (Kent et al. 1995)<sup>6</sup> written in portable C, and using TCL as a command language. *Dervish* provides standard data structures such as linked lists and vectors; a sophisticated memory manager with debugging facilities; a set of astronomical data structures (such as 2-dimensional images and n-dimensional tables); FITS (Wells et al. 2001) reading and writing; and utilities to parse C include (‘.h’) files and provide access to the schema of the C data structures from TCL.

In this paper we make no sharp distinction between algorithms and duties implemented in *PSP* and *Frames*. For those interested, Sec. 2 and Sec. 3 are principally concerned with *PSP*, and the rest of the paper *Frames*.

Although some aspects of the SDSS data are encoded in the pipeline’s source code (for example, the flat fields were taken to be 1-dimensional as we *knew* that the data would be taken in TDI; the data is taken to be well sampled, as we *knew* the plate scale and thought that we knew the seeing), much of the processing is specified via input parameter files; for example, the order and number of photometric bands may be freely changed. Appendix A provides a few more details on these parameters, and also the input files used to specify where inputs and outputs should be found and placed, and how we specify the properties of the camera and CCDs.

---

<sup>3</sup>A 20MHz Motorola 68020 delivered about 3MIPS, and 150Mby of RAM was conceivable.

<sup>4</sup>The situation is rather different in crowded fields where *Frames* slows down dramatically; see Sec. 15

<sup>5</sup>But not the spectroscopic pipelines (Schlegel et al. 2005; Subbarao et al. 2005)

<sup>6</sup>Né Shiva; Kent et al. (1995).

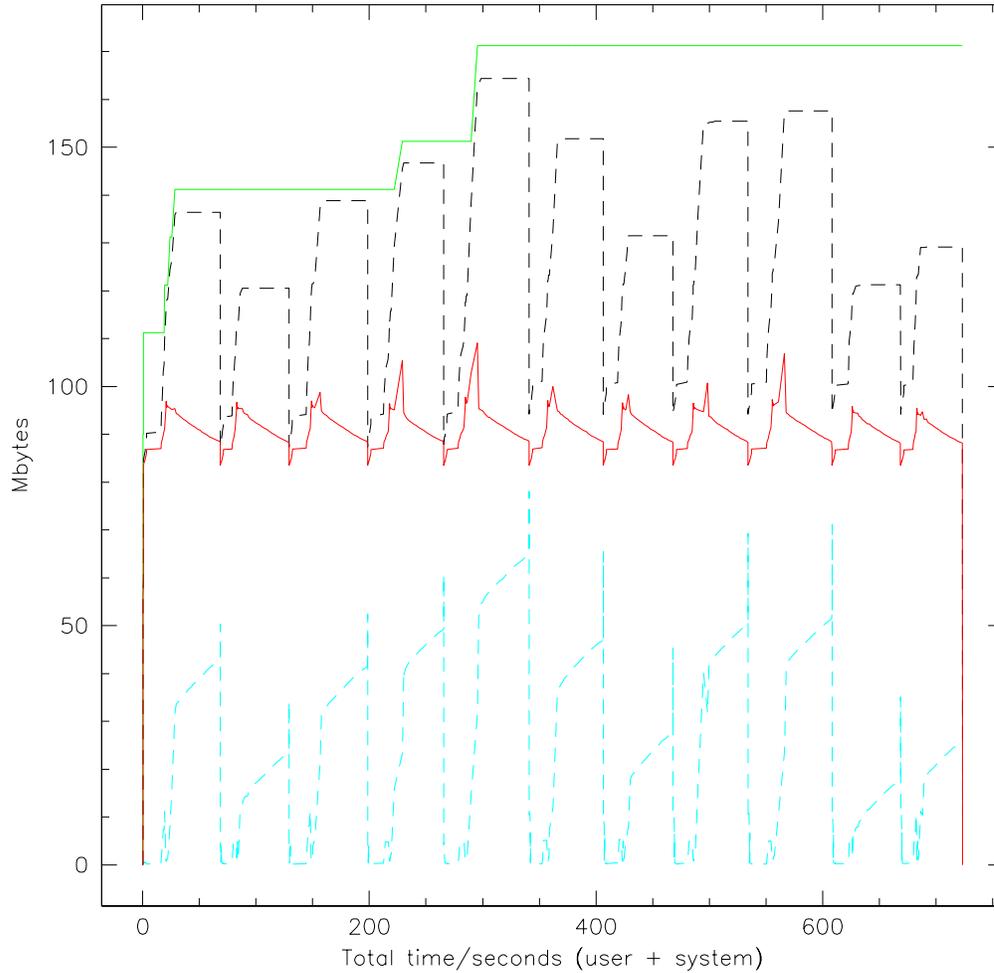


Fig. 1.— The amount of memory used while processing 11 fields of SDSS data (a total of 320Mby) on a 3GHz Xeon processor, taking 66s per field. The red line shows the amount of memory being used, while the cyan line shows how much memory is currently unused, but *has* been used while processing the field; the black dashed line shows the sum of the cyan and red lines. The top green line shows the total amount requested from the system. Each ‘bump’ in the cyan line corresponds to processing a single field, after which inactive memory is released. About 75% of the total processing time (the downward sloping segments of the red line) is spent measuring the properties of the objects.

### 1.1. Book-keeping information; Flags and Masks

It is as important to know when processing has failed as to produce accurate positions and fluxes for well-behaved isolated objects. *Photo* accordingly associates a set of status bits with each pixel, and another set of flag bits with each object in each band. A list of these bits is given in Stoughton et al. (2002); a slightly updated list version is given in Table 1.

Name	Bit	Description
CANONICAL_CENTER	1	<b>Measure Objects</b> used canonical centre, not the one determined in this band
BRIGHT	2	Object's properties were measured in BRIGHT pass
EDGE	3	Object is too close to edge of frame to be measured
BLENDED	4	Object is a blend
CHILD	5	Object is a child
PEAKCENTER	6	The quoted center is the position of the peak pixel
NODEBLEND	7	No deblending was attempted
NOPROFILE	8	Object was too small to estimate a profile
NOPETRO	9	No Petrosian radius could be measured
MANYPETRO	10	Object has more than one Petrosian radius
NOPETRO_BIG	11	No Petrosian radius could be estimated as object is too big
DEBLEND_TOO_MANY_PEAKS	12	Object has too many peaks to deblend
CR	13	Object's footprint contains at least one CR pixel
MANYR50	13	Object has more than one Petrosian 50% radius
MANYR90	15	Object has more than one Petrosian 90% radius
BAD_RADIAL	16	Radial profile includes some low S/N points
INCOMPLETE_PROFILE	17	Object is within the Petrosian radius of the edge of the frame
INTERP	18	Object contains interpolated pixels
SATUR	19	Object contains saturated pixels
NOTCHECKED	20	Object contains NOTCHECKED pixels
SUBTRACTED	21	Object had wings subtracted
NOSTOKES	22	Object has no measured 'stokes' shape parameters

Name	Bit	Description
BADSKY	23	The sky level at the position of the object that the object’s peak pixel is negative
PETROFAINT	24	The surface brightness at the position of at least one candidate Petrosian radius was too low
TOO_LARGE	25	Object is too large to be processed
DEBLENDED_AS_PSF	26	Object was deblended as a PSF
DEBLEND_PRUNED	27	Deblender pruned the list of candidate children
ELLIPFAINT	28	The center’s fainter than the desired elliptical isophote
BINNED1	29	Object was detected in the smoothed $1 \times 1$ binned image
BINNED2	30	Object was detected in the smoothed $2 \times 2$ binned image
BINNED4	31	Object was detected in the smoothed $4 \times 4$ binned image
MOVED	32	Object may have moved (but probably didn’t; see DEBLENDED_AS_MOVING)
DEBLENDED_AS_MOVING	33	Object was deblended as a moving object
NODEBLEND_MOVING	34	Object is a rejected candidate to be deblended as moving
TOO_FEW_DETECTIONS	35	Object has too few detections to deblend as moving
BAD_MOVING_FIT	36	Object’s centroids as a function of time were inconsistent with moving at a constant velocity
STATIONARY	37	The object’s measured velocity is consistent with zero
PEAKS_TOO_CLOSE	38	At least some peaks were too close, and thus merged
BINNED_CENTER	39	Object was found to be more extended than a PSF while centroiding, and the image was thus binned to use a more appropriate smoothing scale
LOCAL_EDGE	40	Object’s center in at least on band was too near the edge of the frame
BAD_COUNTS_ERROR	41	The PSF or fiber magnitude’s error is bad or unknown

Name	Bit	Description
BAD_MOVING_FIT_CHILD	42	A potential child's fit as a moving object was poor, and the child was thus taken to be stationary
DEBLEND_UNASSIGNED_FLUX	43	A significant part of the total flux was divided among the children using the special algorithm for handling otherwise unassigned flux
SATUR_CENTER	44	Object's center's is very near (or includes) saturated pixels
INTERP_CENTER	45	Object's center's is very near (or includes) interpolated pixels
DEBLENDED_AT_EDGE	46	Object's too close to the edge to apply the deblending algorithm cleanly, but was deblended anyway
DEBLEND_NOPEAK	47	Object had no detected peak in this band
PSF_FLUX_INTERP	48	A significant amount of object's PSF flux is interpolated
TOO_FEW_GOOD_DETECTIONS	49	Object has too few good detections to be deblended as moving
CENTER_OFF_AIMAGE	50	Object contained at least one peak whose center lay off the atlas image in some band
DEBLEND_DEGENERATE	51	At least one potential child has been pruned as being too similar to some other template
BRIGHTEST_GALAXY_CHILD	52	Object is the brightest child galaxy in a blend
CANONICAL_BAND	53	This band was primary (usually r)
AMOMENT_UNWEIGHTED	54	Object's 'adaptive' moments are actually unweighted
AMOMENT_SHIFT	55	Object's center moved too far while determining adaptive moments
AMOMENT_MAXITER	56	Too many iterations while determining adaptive moments
MAYBE_CR	57	Object may be a cosmic ray
MAYBE_EGHOST	58	Object may be an electronics ghost
NOTCHECKED_CENTER	59	Object's center is NOTCHECKED
HAS_SATUR_DN	60	Object's counts include DN in bleed trails
DEBLEND_PEEPHOLE	61	Deblend was modified by optimiser
SPARE3	62	Unused
SPARE2	63	Unused
SPARE1	64	Unused

Name	Bit	Description
------	-----	-------------

Table 1:: *Photo*'s bit flags which capture the processing carried out on an object.

The per-pixel information identifies pixels that have been interplated over (Sec. 4.2); that were saturated (Sec. 4.1); that were parts of cosmic rays (Sec. 4.3); that were not checked for objects, part of a BRIGHT object, or part of any object (Sec. 5.2); and which had flux subtracted as part of bright object subtraction (Sec. 5.1).

(**XXX** *EDGE objects*)

## 1.2. Choice of data type

In common with almost all CCD controllers, the SDSS raw data consists of 16-bit unsigned numbers. In view of the scary (for 1993) data rates of 4.6Mby/s, and the high cost of memory, *Photo* processes the data as unsigned short ints ('U16') rather than converting to float. While this leads to significant added complexity in the code, it does mean that *Photo*'s memory footprint while processing a 30Mby frame is only around 100Mby, that we gain a factor of two in D-cache efficiency, and that we avoid much floating point arithmetic (an important consideration for some ancient processors).

Operationally, operating on U16 data has a number of consequences. One is that we add a 'soft bias' of 1000DN ('Data Numbers') to the data during debiasing, to prevent it becoming negative after sky subtraction. Furthermore, when carrying out operations that change the dynamic range of the data (e.g. smoothing) we have to scale up the data by some number of bits to preserve precision. Finally, when carrying out operations that can generate a floating point result (e.g. flat fielding; sky subtraction) we have to be sure to convert back to U16 by adding a uniform random number in the  $[0, 1]$ <sup>7</sup>

## 2. The Point Spread Function

*Photo* makes extensive use of knowledge of the telescope's Point Spread Function (PSF), and it also assumes that the PSF is well sampled by the pixels. Appendix C describes image formation, and also discusses exactly what is meant by the value of a pixel.

---

<sup>7</sup>These numbers are in fact pre-computed and generated inline so there is no significant efficiency hit from this requirement. The added noise,  $1/12$  added in quadrature, is negligible even at  $u$ , our quietest band.

### 2.1. Is SDSS Data Band Limited?

Fig. 2 shows the Fourier power at the Nyquist frequency (Press et al. 1992; Bracewell 2000), ( $f = 1/2$  with our convention) as a function of the PSF’s FWHM; curves are shown for the full Kolmogorov-Fried PSF (Eq. C2), a 2-Gaussian approximation to it, and a single-Gaussian of the same FWHM. In  $1''$  seeing (assuming 0.4 arcsec pixels) the single Gaussian’s amplitude at the band limit is only 0.39% (or 0.56% for a Kolmogorov-Fried PSF); for a FWHM of  $0.8''$  (the best images that the optics can deliver, for which the PSF is admittedly non-Gaussian), the band limit amplitude is still only 2.84% (2.78% for the Kolmogorov-Fried PSF); when the seeing is worse than an arcsecond, the situation only improves.

### 2.2. Modelling the Point Spread Function

Even in the absence of atmospheric inhomogeneities the SDSS telescope delivers images whose FWHMs vary by up to 15% from one side of a CCD to the other (Gunn et al. 2005); the worst effects are seen in the chips furthest from the optical axis. There is also a small amount of fourth-order astigmatism, so the delivered images depend somewhat upon the focus. (~~XXX~~ *JEG: Is all this correct?*)

As the data is taken in Time Delay Integrate (TDI) mode, *temporal* variation of the PSF leads to *spatial* variation of the observed image quality.

If the seeing were constant in time one might hope to understand these effects ab initio (the focus is accurately controlled using a closed-loop servo (Gunn et al. 2005)), but when coupled with time-variable seeing the delivered image quality is a complex two-dimensional function and we chose to model it heuristically using a Karhunen-Loève (KL) transform (Hotelling 1933; Karhunen 1947; Loève 1948); this approach is introduced in Lupton et al. (2001).

### 2.3. The Selection of PSF Stars

The selection of PSF stars is done in two steps. A simple object-finder is used to find a list of candidates brighter than roughly 19-20th magnitude, and in the first crude step objects that are clearly not good candidates to be isolated stars are rejected based on their individual properties (i.e. without considering the overall sample properties). This category includes objects that are too faint, those with saturated or cosmic ray pixels, objects with very close neighbors, and significantly elongated objects (more sophisticated star/galaxy information is not yet available at this stage in the processing).

In the second step, the distribution of image size and ellipticity is used to reject stars that deviate significantly ( $\sim 3\sigma$  or more) from the median. Typically about 50% of the initial objects

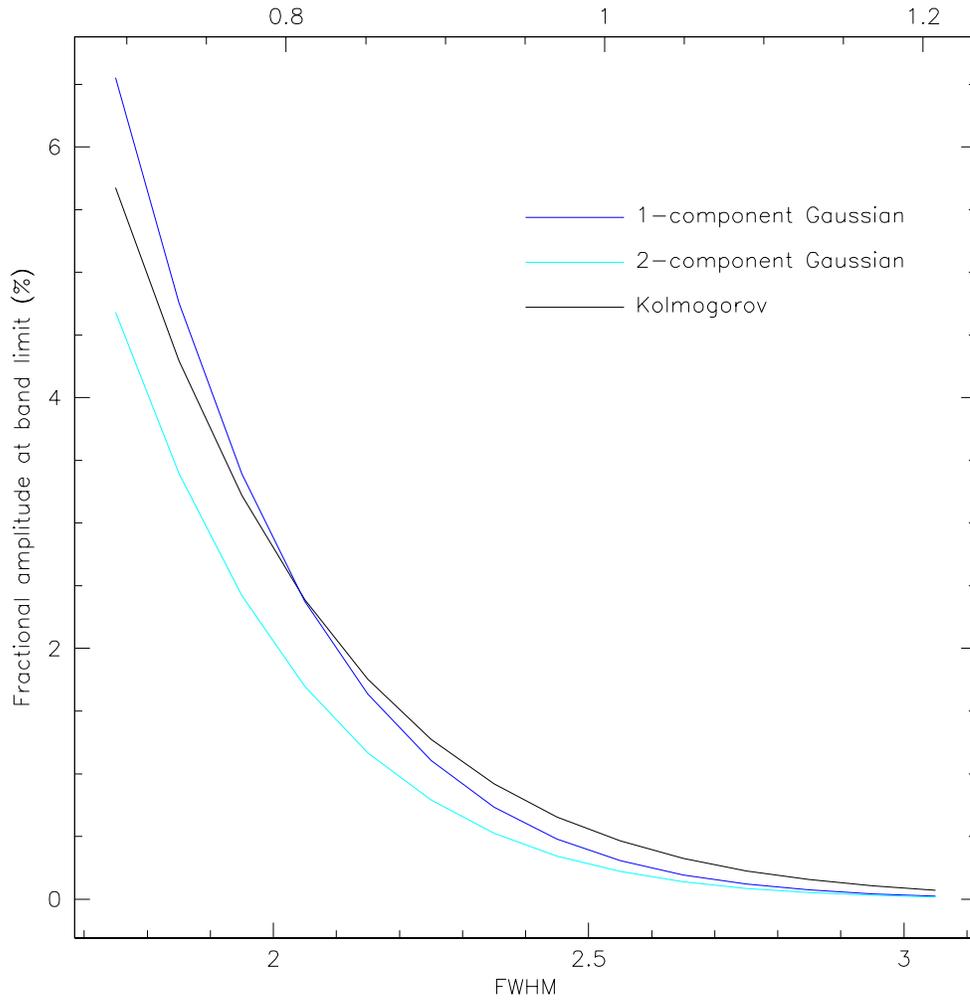


Fig. 2.— The amplitude of the PSF at the Nyquist limit for a PSF normalised to unit flux. The bottom axis is in pixels; the top axis is in arcsec, assuming 0.4 arcsec/pixel.

brighter than 19th magnitude survive both rejection steps.

(**XXX** *Željko: I think that we need rather more detail here. Isn't there some profile clipping? Comparison between bands for non-star?*)

## 2.4. KL Expansion of the PSF

We use these stars to form a KL basis, retaining the first  $n$  terms of the expansion:

$$P_{(i)}(u, v) = \sum_{r=1}^{r=n} a_{(i)}^r B_r(u, v) \quad (1)$$

where  $P_{(i)}$  is the  $i^{\text{th}}$  PSF star, the  $B_r$  are the KL basis functions, and  $u, v$  are pixel coordinates relative to the origin of the basis functions. In determining the  $B_r$ , the  $P_{(i)}$  are normalised to have equal peak value, to avoid uncertainties in the baseline level (Szokoly 1999).

Once we know the  $B_r$  we can write

$$a_{(i)}^r \approx \sum_{l+m \leq N} b_{lm}^r x_{(i)}^l y_{(i)}^m \quad (2)$$

where  $x, y$  are the coordinates of the centre of the  $i^{\text{th}}$  star,  $N$  determines the highest power of  $x$  or  $y$  to include in the expansion, and the  $b_{lm}^r$  are determined by minimising

$$\sum_i \left( P_{(i)}(u, v) - \sum_{r=1}^{r=n} a_{(i)}^r B_r(u, v) \right)^2 ; \quad (3)$$

note that all stars are given equal weight as we are interested in determining the spatial variation of the PSF, and do not want to tailor our fit to the chance positions of bright stars. An alternative way to achieve this would have been to weight each star by  $\sigma^2 + \Upsilon^2$  where  $\sigma$  is a measure of the uncertainty in the star's flux, and  $\Upsilon$  is a floor designed to prevent bright stars from dominating the fit; in the limit  $\Upsilon \rightarrow \infty$  we would recover the equal-weights scheme that we in fact adopted.

### 2.4.1. Application to SDSS data

For each CCD, in each band, there are typically 15-25 stars in a frame that we can use to determine the PSF Sec. 2.3; we usually take  $n = 3$  and  $N = 2$  (i.e. the PSF spatial variation is quadratic). We need to estimate  $n$  KL basis images, and a total of  $n(N+1)(N+2)/2$   $b$  coefficients, and at first sight the problem might seem underconstrained. Fortunately we have many *pixels* in each of the  $P_{(i)}$ , and thus only the number of spatial terms  $((N+1)(N+2)/2$ , i.e. 6 for  $N = 2$ ) need be compared with the number of stars available.

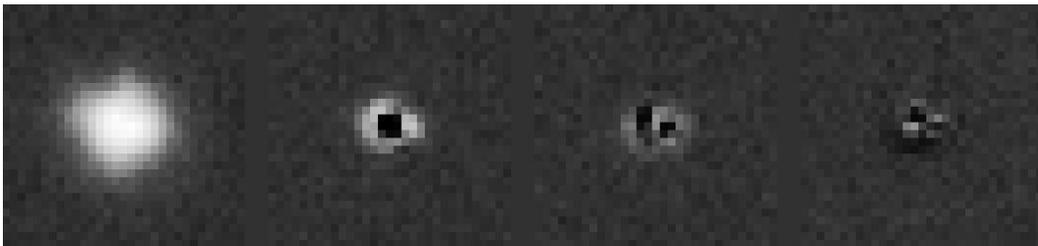


Fig. 3.— The KL basis images for frame 756-z6-700, using a histogram-equalised stretch.

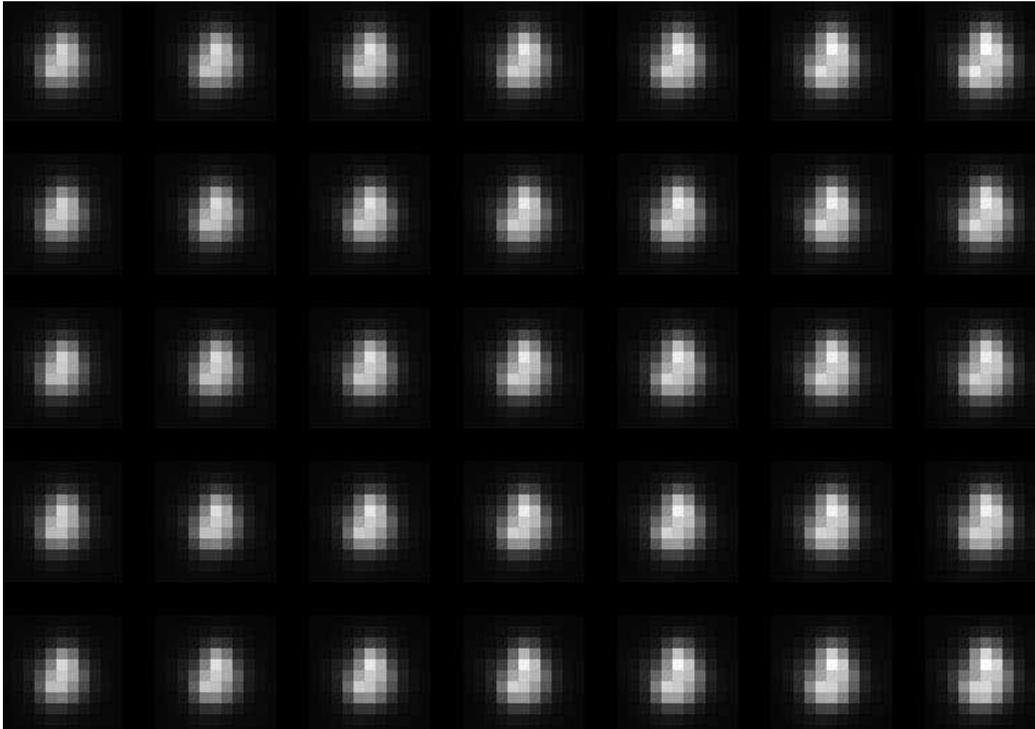


Fig. 4.— The estimated PSF for 35 positions in frame 756-z6-700, using a linear stretch. This is early SDSS data, and for one of the CCDs with worst image quality, and the astigmatism is clearly seen. The first component of Fig. 3 includes this astigmatism, although it is not obvious with the stretch used for that figure.

In fact, rather than use only the stars from a single frame to determine that frame’s PSF, we include stars from both preceding and succeeding frames in the fit. This has the advantage that the spatial variation is better constrained at the leading and trailing edges of the frame; that the PSF variation is smoother from frame to frame; and that we have more stars available to determine the PSF. The 4 KL basis images from an early SDSS run are shown in Fig. 3 and the resulting reconstructed PSFs at 35 points within the frame are shown in Fig. 4; this was early data, and the images are clearly astigmatic, with the astigmatism varying as a function of position.

In practice we use a range of  $\pm 2$  frames to determine the KL basis functions  $B_r$  and  $\pm 1/2$  frame to follow the spatial variation of the PSF. If we try to use a larger window we find that variation of the  $a^r$  coefficients is not well described by the polynomials that we have assumed. We have not tried using a different set of expansion functions (e.g. a smoothing spline, or Fourier series).

Additionally, we force the KL basis images with  $n > 0$  to have 0 mean in their outermost pixels, those more than approximately  $7''$  from the center of the star ( $7''$  is approximately the radius of the region used to determine the KL PSF)<sup>8</sup>. This doesn’t force the PSF to be zero in its outermost parts, but it does mean that the PSF is taken to be constant at  $7''$ .

This modification to the basis functions means that they are no longer strictly orthogonal, but it makes the PSF determination significantly more robust to errors in the background level.

Consider the case that all stars have identical PSFs beyond  $7''$  from their centres, while their cores vary. The  $0^{th}$  KL basis image is the mean of all the input stars, and (in the absence of noise) *all* of the stars are perfectly described by this image, and all higher basis images must be exactly zero at the edge. What happens if the stars’ fluxes or background level are not perfectly measured? This leads to basis images (with  $n > 0$ , i.e. not just the mean input star) with non-zero background level, as well as a structure near the centre of the star that describes some part of the PSF’s true variability. Let’s assume that most of this non-zero background appears in the  $r^{th}$  basis image. When we come to estimate the PSF at some point in the frame where the PSF’s core has significant variation described by this  $r^{th}$  image, then the background level of the estimated PSF can be badly off — the initial small error in the sky can be amplified by the need to match the core.

The information about the PSF is written out to `psField` files; see Appendix G for information on how to reconstruct the PSF given these files.

#### 2.4.2. Gauging the Success of the KL Expansion

The success of the KL expansion is gauged by comparing the PSF photometry based on the modeled KL PSFs, to the aperture photometry for the same (bright) stars. The width of the

---

<sup>8</sup>This was introduced in version `v5.4.20`; some older reductions may still be available.

distribution of these differences is typically 1% or less, and indicates an upper limit on the accuracy of the PSF photometry. Without accounting for the spatial variation of the PSF across the image, the photometric errors would be as high as 15%.

(**XXX** *Reducing the order of the expansion/expanding the window in case of difficulty*)

Each band has an associated status, and a summary status is provided for each field; the interpretation of these values is discussed in Appendix G.1.

An example of the instantaneous image quality across the imaging camera is shown in Fig. 5, where each rectangle represents one chip. The stretch is histogram equalized and the dynamic range is from 1.4 arcsec to 2.0 arcsec (FWHM). (**XXX** *Do we have a more recent version of this?*)

## 2.5. Other Uses for Point Sources Detected by the *PSP*

In addition to the stars employed to characterize the PSF, *PSP* also detects and processes ‘wing’ and ‘frame’ stars. These are stars bright enough to be saturated, either mildly in the case of wing stars, or dramatically in the case of frame stars. The frame stars were intended to be used to characterise the very outermost parts of the PSF, but are not currently exploited. The wing stars are used, in concert with the unsaturated stars used to determine the KL expansion of the PSF, to determine a composite radial profile over a wide range of radii; this is used in determining aperture corrections; see Sec. 10.3.3. The composite profile is a non-parametric maximum-likelihood model, which patches together a set of overlapping pieces of profiles from different stars.

(**XXX** *More details? A figure?*)

## 3. Determining Bias Vectors, Scattered Light, and Flatfields

### 3.1. The Scattered Light Correction and Flatfield Determination

An initial attempt to derive flatfield vectors by assuming a flat sky background failed because of the significant scattered light contribution. This scattered skylight presumably comes from solid angles on the sky in very close proximity to the area being imaged, and can have amplitudes as large as 10% of the background intensity (for  $u$  chips close to the chip edges, in other bands it is typically several times smaller). This variation then propagates with the same amplitude to photometric errors when flatfield is determined using sky background. Instead, the flatfield vectors are determined externally and provided as input to image processing pipelines. We use provided flatfield vectors and measured background to derive the scattered light correction. This additive correction is defined as the background excess over its minimum and formally treated as a bias contribution in subsequent processing.

(**XXX** *Smoothing/averaging lengths. Use of overscan/extended registers*)

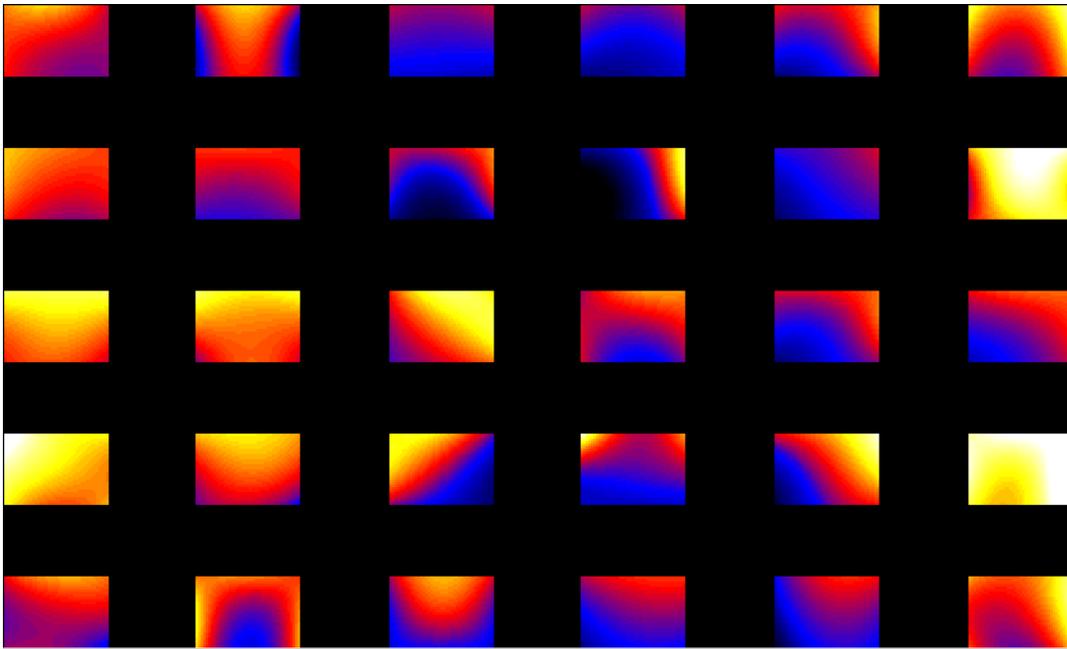


Fig. 5.— The instantaneous second moment (width) of the point spread function as a function of position in the camera. Each square represents one of the photometric CCDs. The width variation is due to time variability during the 54 s integration, plus effects in the telescope optics. The stretch is linear, from 1.4 to 2 arcsec.

### 3.2. External Determination of Flatfield Vectors

The external (in sense that it is not part of regular run-by-run processing) determination of flatfields is an iterative procedure that utilizes two different constraints. It is iterative because it is based on successive corrections to adopted flatfield vectors until the difference between measured stellar magnitudes and their “true” values becomes uncorrelated with chip column. Ideally, the “true” values would be provided by an external catalog. However, due to faint levels probed by SDSS, its large sky coverage and high photometric accuracy requirement, such a catalog does not yet exist. A partial solution is to use measurements of photometric calibration stars provided by the Photometric Telescope (Tucker et al. 2005). However, the number density of these stars on the sky is insufficient to constrain flatfield variation with required spatial resolution across chips, especially in the  $u$  band. Additional constraints come from a requirement that the mean position of the stellar locus in a four-dimensional color space is not correlated with chip column. This position is reproducible across the sky to better than 0.01 mag. (for details see Ivezić et al. (2004)) and provides a powerful method to constrain flatfield vectors. Since it is a color-based method, it provides one constrain fewer than needed, and the system is closed by using PT measurements in the  $g$ ,  $r$  and  $i$  bands. These measurements are used in an equally-weighted linear combination to increase the signal-to-noise ratio.

### 3.3. Flatfield “Seasons”

It was an unpleasant discovery that flatfield vectors vary abruptly with time: they are constant within the measurement noise (typically a fraction of percent) for many months and then change by several percent, most strongly in the  $u$  band. These abrupt changes are illustrated in Fig. 6 and correspond to changes in the vacuum state of the camera due to regular maintenance (~~XXX~~ *I’d ask Jim to read this*). Flatfield corrections due to such temporal variations are constrained and applied separately for each season. After only one flatfield iteration, the residual systematic photometric (color) errors do not exceed a few tenths of a percent, as illustrated in Fig. 7.

## 4. Removing the Instrumental Signature

TDI data differs from staring-mode CCD data only in so much as the flat-fields and biases are 1-dimensional, averaged over the height of the CCD, and this washes out most small scale structure.<sup>9</sup>

---

<sup>9</sup>This averaging also effectively removes all traces of fringing, and we accordingly don’t have to remove a fringe frame, even in the  $i$  band (we employ thick CCDs at  $z$ ).

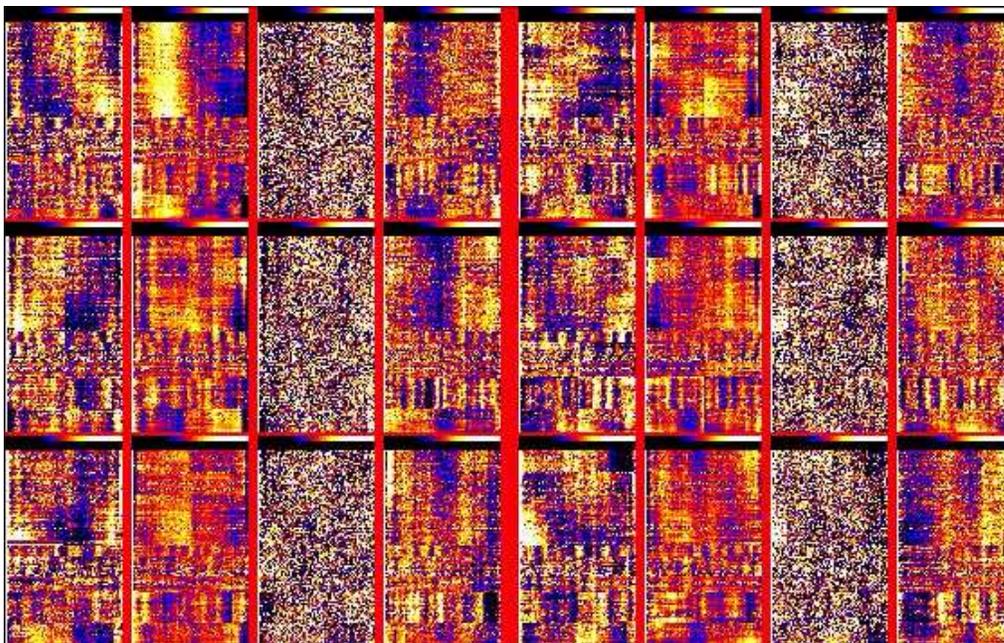


Fig. 6.— The position of the stellar locus as a function of time and chip column. This mosaic contains 24 panels: 6 groups of four horizontally arranged images for each camera column, with columns 1 and 4 in the top row, 2 and 5 in the middle row, and 3 and 6 in the bottom row. The four panels in each group correspond to four principal colors ( $s$ ,  $w$ ,  $x$  and  $y$ , as described by Ivezić et al. (2004)). Each panel displays the deviation of the principal color from the mean for the whole chip as a function of chip column position ( $x$  axis) for  $\sim 100$  runs sorted by time ( $y$  axis), on a linear scale with  $\pm 1\%$  stretch corresponding to blue/yellow colors (the bar on the top of each panel is stretched from  $-2\%$  to  $2\%$ ). The strong abrupt changes of the structure in the  $y$  direction at the

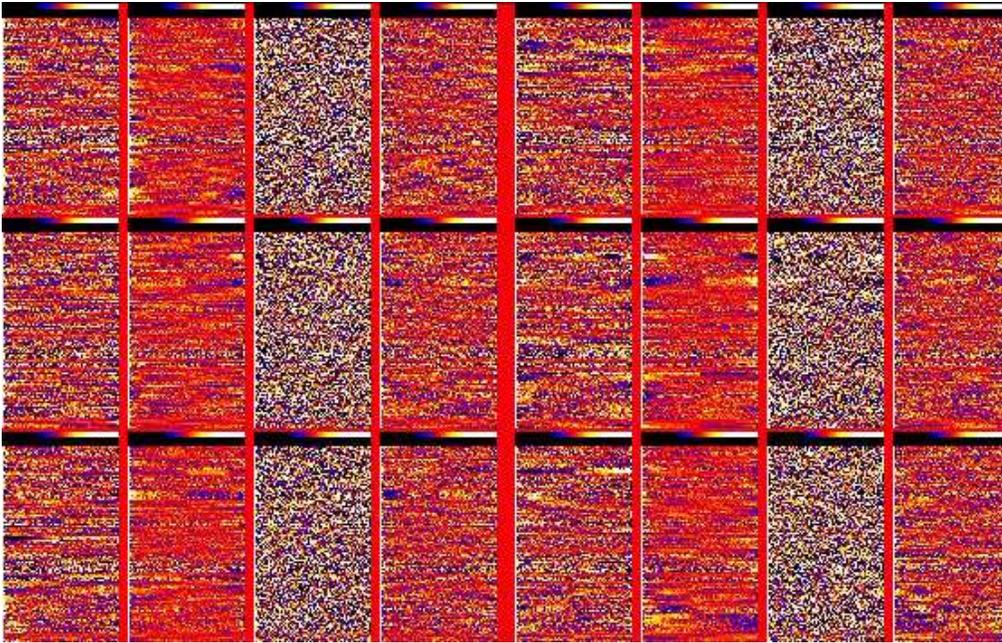


Fig. 7.— Analogous to Fig. 6, except that here a corrected flatfield vector for each flatfield “season” visible in Fig. 6 was used to process the data. The residual systematic photometric (color) errors do not exceed a few tenths of a percent.

#### 4.1. Flat Fielding, Bias Subtraction, and Linearization

A  $512 \times 412$  region of an SDSS image is shown in Fig. 4.1, and displays most of the characteristic problems associated with CCDs. (**XXX** *Where does the description of this figure belong?*)

Before bias-subtraction and flatfielding, we run an object detection algorithm looking for saturation (bleed) trails. We detect all pixels over a given threshold (which is different for each CCD, and for each half of the CCDs that have two working amplifiers). The resulting set of pixels is grown out by one pixel<sup>10</sup> to allow for the pixels affected by bleeding at the top and bottom of the trails, and for bad charge transfer efficiency (CTE) in the serial register.

The total number of counts within each trail is stored, along with the list of affected pixels.<sup>11</sup> These counts are included in the flux of some, but not all, saturated objects — see Sec. 10.5.

The known ‘hot’ columns (see Sec. A.1) are removed from the list of saturated pixels (although they are, naturally, included in the list of bad columns to be interpolated over).

We then correct for non-linear response of the amplifiers using a 16-bit lookup table; as described in Doi (2005) we measure the response of each amplifier to varying light levels, and find that it is well described by a function of the form:

$$D = d + \begin{cases} 0 & \text{if } d < T; \\ c(\lg(d) - \lg(T)) & \text{otherwise.} \end{cases}$$

where  $d$  is the raw value and  $D$  the corrected value of a pixel.  $c$  and  $T$  are coefficients that are read from one of *Photo*’s parameter files (Sec. A).

#### 4.2. Interpolation over CCD Defects and Bleed Trails

*Photo* uses linear prediction (e.g. Press et al. (1992); Press (1993)) to interpolate over missing data, which comes in three flavours: bad columns (whose positions we know *a priori*), bleed trails, and cosmic rays. We merge the lists of saturated pixels with the list of known bad columns in the CCD (see Appendix B) before interpolating over them as part of the process of removing the instrumental signature; cosmic rays are interpolated over later.

The number of bad columns per CCD varies from 2 to 44, with a mean of 15. The majority of these have identified traps, with a sprinkling of depressed and noisy columns. The median separation between bad columns is about 75 pixels, although each CCD has on average a couple of adjacent bad columns, which are harder to interpolate over satisfactorily. As mentioned in Sec.

---

<sup>10</sup>i.e. the set of saturated pixels is augmented by every pixel which has a saturated neighbor, horizontally, vertically, or diagonally.

<sup>11</sup>the data structure employed actually stores sets of `row`; `column0`, `column1` triples rather than each pixel

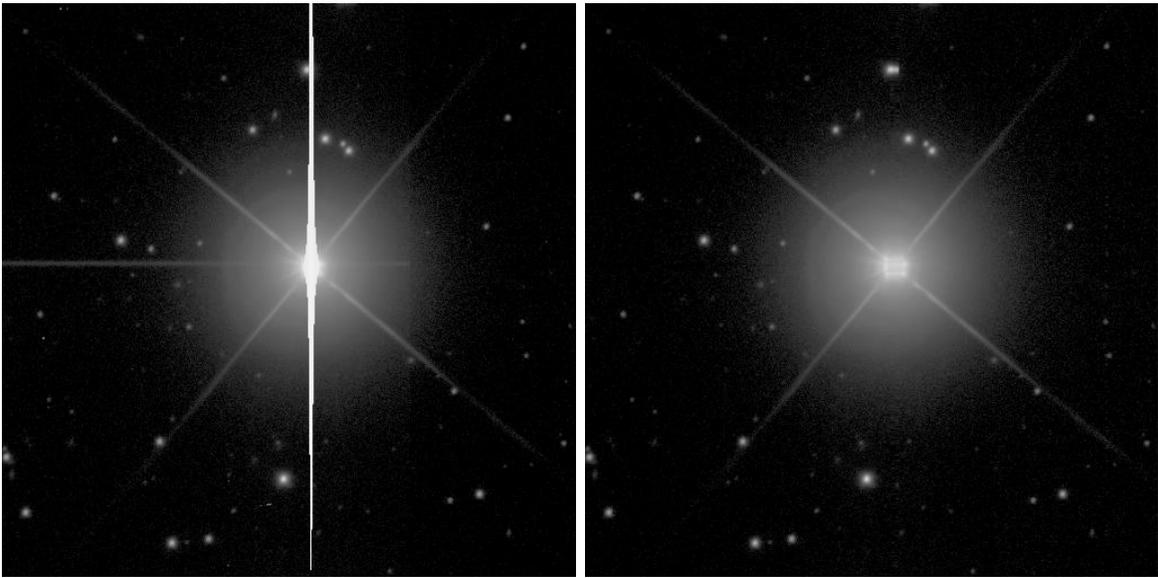


Fig. 8.— A  $512 \times 512$  region of an SDSS image, showing a bright star. The left hand panel is of the raw data, showing cosmic rays, and the star’s diffraction spikes, bleed trails, ghosts, and serial register artifact. The split between the CCD’s two amplifiers is visible to the right of the star. The right hand panel shows the corrected version of the same image. The data is taken in the  $i$  band, which is why the star has such an extensive halo (see Sec. 5.1).

4.3, there are of order 150 cosmic rays hits in each frame (and therefore about 500 contaminated pixels), with the exception of the  $g$  band CCD with about 330 CR hits.

Given a stationary process,

$$y_i \equiv s_i + n_i$$

where  $s_i$  is the signal, and  $n_i$  the noise, we may predict the value of the signal  $s_p$  at some point  $p$  as

$$\tilde{s}_p = \sum_j D_{pj} y_j; \quad (4)$$

it is clear that the estimator is unbiased if  $\sum_j D_{pj} = 1$ .

Let us choose the  $D_{pj}$  to minimise the expectation value of the error  $(s_p - \tilde{s}_p)^2$  subject to the constraint that  $\sum_j D_{pj} = 1$ ; that is we wish to minimise

$$\langle (s_p - \sum_j D_{pj} y_j)^2 \rangle - 2\lambda \sum_j D_{pj}.$$

Differentiating with respect to  $D_{pj}$  and noting that  $\langle sn \rangle = 0$ , this becomes

$$D|_{i=p} = (S + N)^{-1} (S|_{i=p} - \lambda \mathbf{E}) \quad (5)$$

where we have written  $\langle s_i s_j \rangle$  and  $\langle n_i n_j \rangle$  as the symmetric matrices  $S$  and  $N$ ,  $\langle s_p s_j \rangle$  and  $D_{pj}$  as the vectors  $S|_{i=p}$  and  $D|_{i=p}$ , and defined  $\mathbf{E}$  as the vector with all components equal to one. The Lagrange multiplier  $\lambda$  is determined by the condition  $\sum D_{pj} = 1$ , i.e.

$$\lambda = \frac{\mathbf{E}^T (S + N)^{-1} S|_{i=p} - 1}{\mathbf{E}^T (S + N)^{-1} \mathbf{E}} \quad (6)$$

#### 4.2.1. Application to Seeing-convolved Data

Let us apply this theory to interpolation over defects in images, so  $p$  will be the index for a bad pixel. We shall assume that the images consist purely of point sources, so the signal  $s$  is given by

$$s(\mathbf{x}) = P \otimes \sum_l (B + \delta(\mathbf{x} - \mathbf{x}_l))$$

where  $B$  is the background,  $P$  is the PSF,  $\otimes$  signifies a convolution, and the  $\mathbf{x}_l$  are the positions of the sources. Setting  $B = 0$  (which will not affect  $D$  as our estimator is unbiased), and assuming that the sources are scattered randomly,

$$\begin{aligned} S &\equiv \langle s_i s_j \rangle = \langle P_i P_j \rangle = \langle P(\mathbf{x}_i) P(\mathbf{x}_j) \rangle = \langle P(\mathbf{0}) P(\mathbf{x}_i - \mathbf{x}_j) \rangle \\ S|_{i=p} &\equiv \langle s_p s_j \rangle = \langle P(\mathbf{0}) P(\mathbf{x}_p - \mathbf{x}_j) \rangle. \end{aligned}$$

We may take the noise to be uncorrelated, so that  $N$  is diagonal.

In order to apply this to interpolation, let us set the noise to be infinitely larger in the bad columns than the good. Consider the case of infinite signal-to-noise ratio; Eq. 5 shows, unsurprisingly, that if there are no bad columns the best estimate of  $y_j$  is  $y_j$  itself ( $N = 0$ , so  $\lambda = 0$  and  $D = S^{-1}S = I$ ). Let us now take the PSF  $P$  to be a Gaussian  $G(\alpha)$  (Eq. C1), i.e.

$$S(\mathbf{x}_i - \mathbf{x}_j) = \exp(-|\mathbf{x}_i - \mathbf{x}_j|^2/(4\alpha^2)),$$

and restrict ourselves to only  $2M + 1$  terms (centred on the bad column) in estimating the  $D|_{p=i}$ . Table 2 shows the results for a variety of values of  $M$ , for  $\alpha = 1$  and considering only one-dimensional interpolation; Fig. 9 shows the result of applying the  $M = 2$  coefficients to simulated SDSS data in slightly worse seeing (1.03" FWHM, i.e.  $\alpha = 1.1$  pixels). The image is the sum of a wide Gaussian and two PSFs separated by 1.75 FWHM. Fig. 10 shows the effect of trying to interpolate over two bad columns, and Fig. 11 shows the effect of decreasing the pixel size by about 27% (or, equivalently, making the seeing 27% worse).

In addition to the coefficients, the table gives the values of  $\sum D_i^2$ , the variance in the interpolated value in units of the variance of the good pixels. By choosing  $M$  suitably it is possible to tune the variance of the interpolated pixels, although whether this is worthwhile is not clear.

The coefficients in Table 2 were calculated on the assumption of infinite signal-to-noise; Table 3 gives the coefficients for interpolating over one bad column as a function of the S/N ratio.

In the limit of zero S/N the interpolated values are all equal to the mean of the end points (as there is no information in the signal, the best estimate of  $s_p$  is simply the background, and the best estimate of *that* is the mean of the available data).

### 4.3. Cosmic Ray Rejection

For most of the area covered, the SDSS has only one image in each band, making the traditional technique for rejecting cosmic rays (comparing two or more images) inapplicable. Even when we have multiple images, they were taken at different epochs and variability would significantly complicate a cosmic ray rejection scheme based upon comparing repeated images. (XXX Need

M	$\sum D_i^2$	$D _{i=p}$								
1	0.500				0.500	0	0.500			
2	1.347			-0.274	0.774	0	0.774	-0.274		
3	2.004		0.166	-0.510	0.844	0	0.844	-0.510	0.166	
4	2.717	-0.108	0.351	-0.643	0.900	0	0.900	-0.643	0.351	-0.108

Table 2: Interpolation coefficients for a single bad column, infinite signal-to-noise, and various values of  $M$ . Also listed are the values of  $\sum D_i^2$ , which give the variance in the interpolated pixels (in units of the variance of the good pixels).

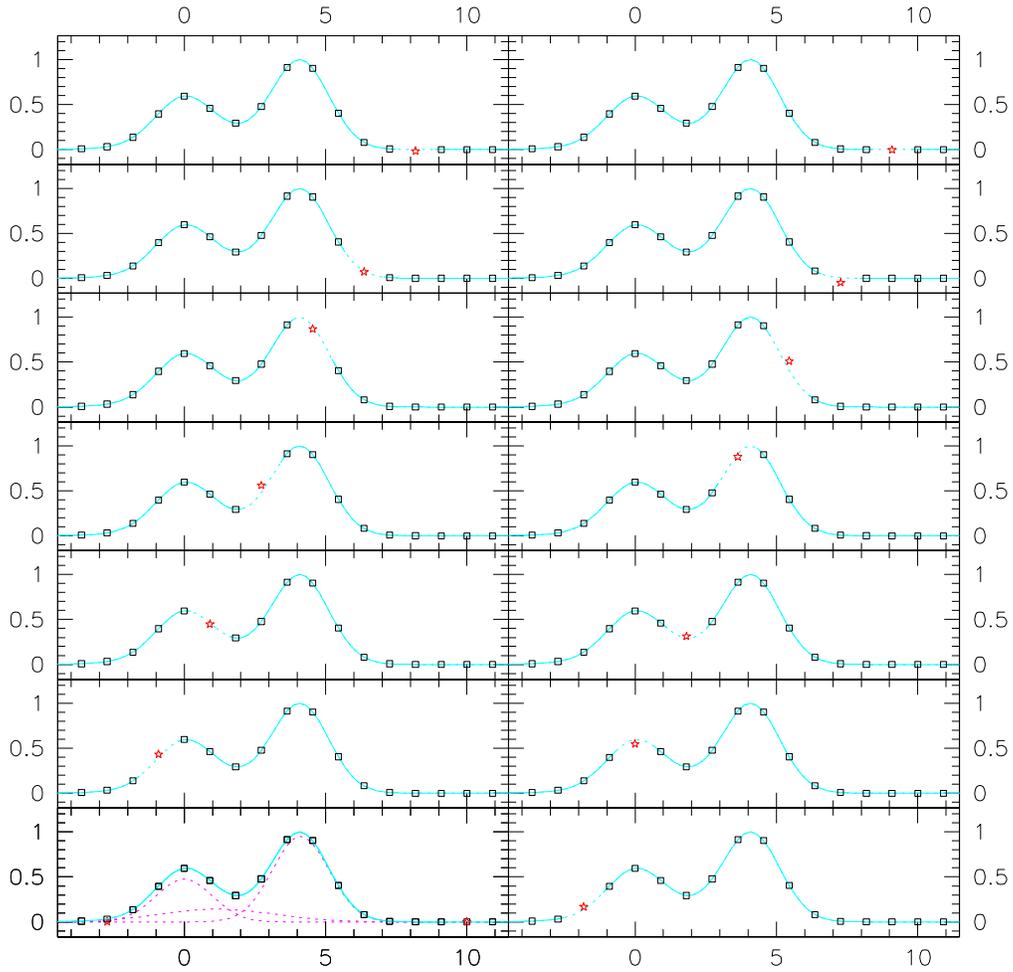


Fig. 9.— Interpolating over a single bad column in  $1.03''$  FWHM Gaussian seeing with  $0.396''$  pixels (i.e.  $\alpha = 1.1$  pixel). The bottom left panel shows the model in dotted magenta, a pair of PSFs and a wider “galaxy”. The ‘good’ pixels are shown as open squares, and the interpolated values are shown as a red star. The missing data point is shifted one pixel to the right in each successive panel.

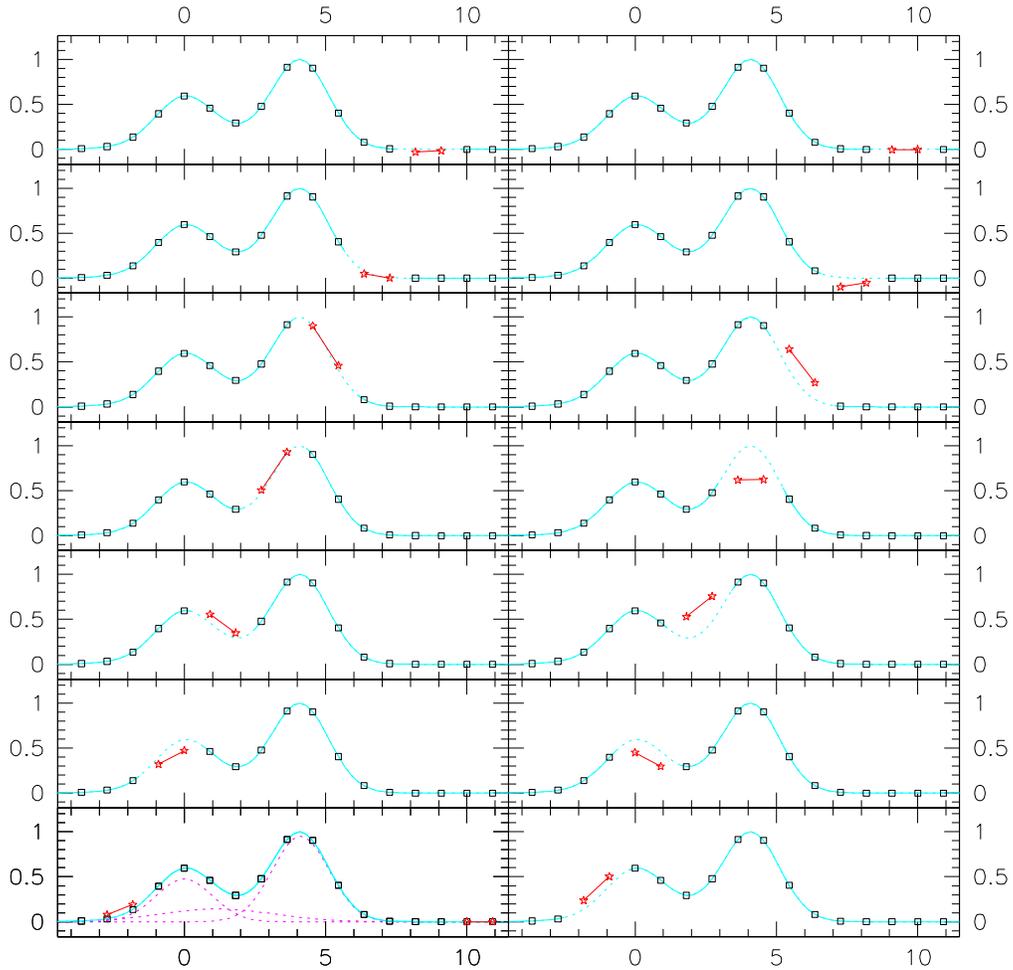


Fig. 10.— Interpolating over two bad columns, otherwise the simulation is identical to Fig. 9; note that the recovery of the missing values is worse.

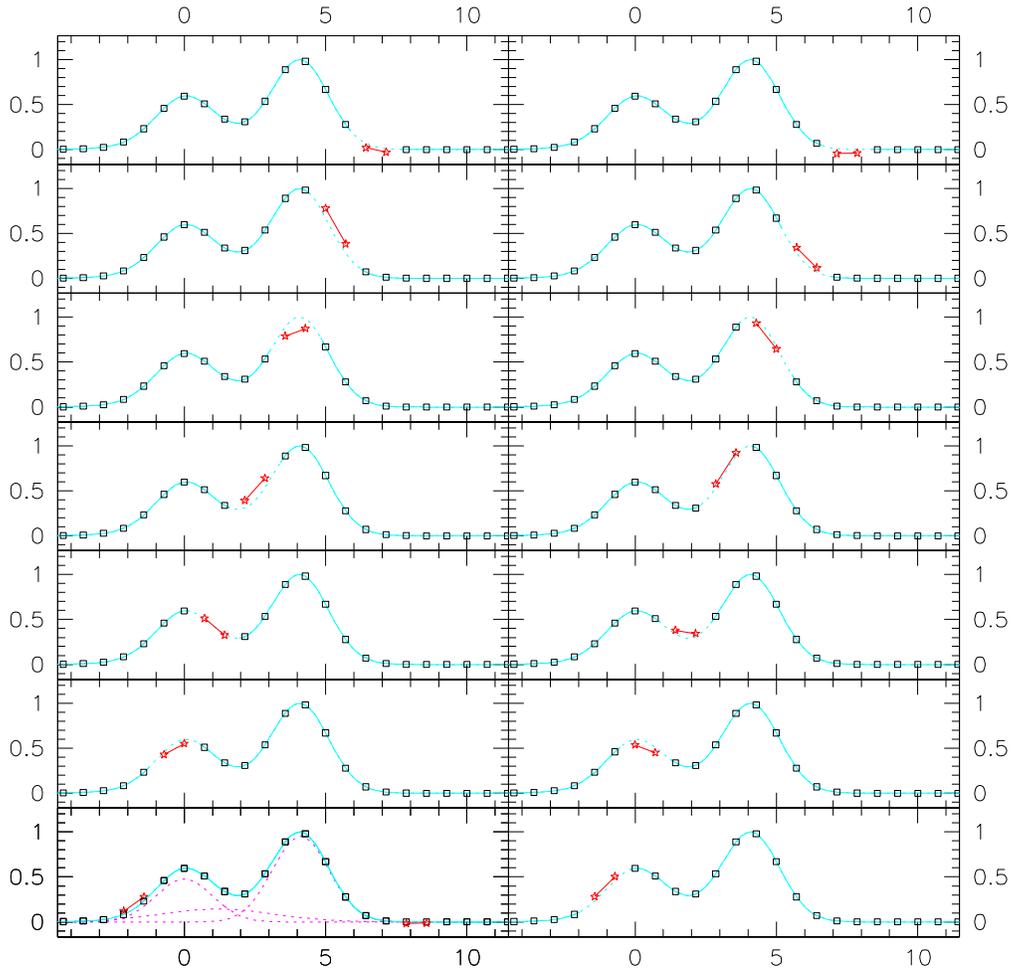


Fig. 11.— Interpolating over two bad columns with  $0.311''$  pixels ( $\alpha = 1.4$  pixel), otherwise the simulation is identical to Fig. 9; note that the recovery of the missing values is significantly better than in Fig. 10.

*literature search*).

*Photo* looks for cosmic rays in two different ways. Firstly, during the removal of the instrumental signature (Sec. 4.1) we search every image looking for ‘easy’ cosmic rays which we record (by setting the `CR` flag bit) and interpolate over. We then proceed with the image processing on the assumption that there are no surviving cosmic rays. Then, while classifying each detected object, we ask the question, ‘Is this object consistent with being a cosmic ray?’; if the answer is ‘Yes’ we set a flag bit `MAYBE_CR`. The motivation for this two-stage approach is that we have found it impossible to tune the first algorithm to be perfect, finding all true cosmic rays with no false detections. The consequences of a false detection are severe, as interpolating over a ‘cosmic ray’ compromises the photometry of any neighbouring real objects. We shall discuss the second pass as part of the discussion of star-galaxy classification (Sec. 11.1).

#### 4.3.1. Cosmic Ray Rejection: pass I

The first-pass algorithm is to search for all pixels which satisfy a series of conditions:

1. That the candidate bad pixel  $p$  not be adjacent to a saturated pixel.
2. That  $p$ ’s intensity  $I$  exceed the locally-determined background (actually the mean of pairs of neighbouring pixels) by  $n\sigma$  where  $\sigma^2$  is the sky variance. We have usually taken  $n = 6$ .
3. That the gradients near the pixel not exceed that band-limit imposed by the PSF; specifically we require that no pixel be part of a peak which is sharper than the centre of a star centred in a pixel. Allowing for noise, this condition becomes

$$I - c * N(I) > P(d) (\bar{I} + cN(\bar{I})) \tag{7}$$

where  $c$  is a constant,  $N(I)$  is the standard deviation of  $I$ ,  $P(d)$  is the value of the PSF at a distance  $d$  from the centre of a star,  $\bar{I}$  is the average of two pixels a distance  $d$  away from our

S/N	$\sum D_i^2$	$D _{i=p}$				
$\infty$	1.347	-0.274	0.774	0	0.774	-0.274
500	1.333	-0.270	0.770	0	0.770	-0.270
100	1.279	-0.257	0.757	0	0.757	-0.257
50	1.218	-0.242	0.742	0	0.742	-0.242
10	0.876	-0.146	0.646	0	0.646	-0.146
5	0.654	-0.068	0.568	0	0.568	-0.068
2	0.410	0.050	0.450	0	0.450	0.050
1	0.311	0.126	0.374	0	0.374	0.126
0	0.250	0.250	0.250	0	0.250	0.250

Table 3: The interpolation coefficients for a single column defect as a function of S/N.

pixel. We have found that in practice we must multiply  $P(d)$  by some fiddle factor,  $c_2 < 1$ , to avoid flagging the cores of stars as ‘cosmic rays’. We have found that the values  $c = 3.0$ ,  $c_2 = 0.8$  work well in practice.

These conditions are applied sequentially to the pixel being studied using the four pairs of neighbouring pixels (NS, EW, NW-SE, and NE-SW,  $d = 1, 1, \sqrt{2}, \sqrt{2}$ ). The candidate cosmic ray must exceed condition 2 for all four pairs of neighbours, and condition 3 for at least one pair.

The thinking behind this choice is that while most cosmic rays contaminate more than one pixel, they pass through the CCD in a straight line so almost all pixels have at least one pair of uncontaminated neighbours.

Once a cosmic ray contaminated pixel is identified, its location is noted and its value is replaced by an interpolation based on the pair of pixels that triggered condition 3; the interpolation algorithm used is the same as that used for fixing bad columns (Sec. 4.1). This removal of contaminated pixels as they are found makes it easier to find other pixels affected by the same cosmic ray hit.

Once the entire frame has been processed, the pixels identified individually as being contaminated by cosmic rays are assembled into cosmic ray ‘events’ of contiguous pixels. Each such event must contain more than a minimum number of electrons (not DN); we have adopted a threshold of  $150e^-$ .

We then go through the frame again, looking at pixels adjacent to these cosmic ray events. Processing is identical, except that we set  $c_2 = 0$  for these extra contaminated pixels.

We find about 160 cosmic rays per field in all bands except  $g$ , where we find about 330 (presumably due to radioactivity in the filter) and  $z$  where there are about 135; in total we thus find about as many cosmic rays as objects in the 5 frames that make up a field.

#### 4.4. Serial register artifacts

The serial registers in the SITE CCDs used in the SDSS camera (Gunn et al. 1998) are light-sensitive. Because the SDSS data is taken in TDI mode, every object deposits a small amount of charge as it crosses the serial register (the fraction would be  $1/2048$  for our  $2048 \times 2048$  CCDs if the serial register were the same width as a row of the CCD, in fact the fraction is rather larger). Because the serial register is being clocked as objects move across it, the charge is smeared out across the entire width of the detector, or half the width if two amplifiers are being used; see Fig. 4.1.

## 5. Sky Subtraction and Object Detection

### 5.1. Sky Subtraction

*This describes the v5\_4 sky subtraction algorithms. The v5\_3 algorithms differ in two ways:*

- *No debiasing is applied; it's assumed that the mean  $\sim$  median for Poisson variables in the large  $\mu$  limit*
- *The radial profiles (see Sec. 9.2) are clipped at  $2.326\sigma$ , not  $4.0\sigma$ .*

*These changes largely, but not entirely, cancel out*

It is quite clear what astronomers mean by ‘sky’: the *mean* value of all pixels in an image which are not explicitly identified as part of any detected object. It is this quantity which, when multiplied by the effective number of pixels in an object, tells us how much of the measured flux is not in fact associated with the object of interest. Unfortunately, means are not very robust, and the identification of *pixels ... not explicitly identified as part of any detected object* is fraught with difficulties.

There are two main strategies employed to avoid these difficulties: the use of clipped means, and the use of rank statistics such as the median. Appendix F discusses some of the issues to which these approaches lead when applied to data with a Poisson error distribution.

*Photo* performs two levels of sky subtraction; when first processing each frame it estimates a global sky level, and then, while searching for and measuring faint objects, it re-estimates the sky level locally (but not individually for every object).

The initial sky estimate is taken from the median value of every pixel in the image (more precisely, every fourth pixel in the image), clipped at  $2.32634\sigma$ . This estimate of sky is corrected for the bias introduced by using a median, and a clipped one at that. The statistical error in this value is then estimated from the values of sky determined separately from the four quadrants of the image.

Using this initial sky estimation, *Photo* proceeds to find all the bright objects (typically those with more than  $60\sigma$  detections). Among these are any saturated stars present on the frame, and *Photo* is designed to remove the scattering wings from at least the brighter of these — this should include the scattering due to the atmosphere, and also that due to scattering within the CCD membrane, which is especially a problem in the i band. In fact, we have chosen *not* to aggressively subtract the wings of stars, partly because of the difficulty of handling the wings of stars that do not fall on the frame, and partly due to our lack of a robust understanding of the outer parts of the PSF (**XXX Discuss further?**). With the parameters employed, only the very cores of the stars (out to 20 pixels) are ever subtracted, and this has a negligible influence on the data. Information about star-subtraction is recorded in the fpBIN files, in HDU 4.

Once the BRIGHT detections have been processed, *Photo* proceeds with a more local sky estimate. This is carried out by finding the same clipped median, but now in  $256 \times 256$  pixel boxes, centered every 128 pixels. These values are again debiased.

This estimate of the sky is then subtracted from the data, using linear interpolation between these values spaced 128 pixels apart; the interpolation is done using a variant of the well-known Bresenham algorithm (Bresenham 1965) usually employed to draw lines on pixellated displays.

This sky image, sampled every  $128 \times 128$  pixels is written out to the `fpBIN` file in HDU 2; the estimated uncertainties in the sky (as estimated from the interquartile range and converted to a standard deviation taking due account of clipping) is stored in HDU 3. The value of sky in each band and its error, as interpolated to the centre of the object, are written to the `fpObjc` files along with all other measured quantities. In the case of blended objects, there are other contributions to the sky value associated with each objects; see Sec. 8

After all objects have been detected and removed (as described below), *Photo* has the option of redetermining the sky using the same  $256 \times 256$  pixel boxes; in practice this has not proved to significantly affect the photometry.

## 5.2. Object Detection

**(XXX Needs more work. What's in the EDR? At least this discusses atlas images)**

Consider the problem of fitting a model PSF to a faint star in an image; The log-likelihood of the fit is proportional to  $\sum_i (O_i - fP_i)^2$  where  $O_i$  are the measured pixel intensities (with the sky subtracted),  $P_i$  is the form of the PSF, the sum is over all the pixels in the image, and the noise is taken to be the same in all pixels (i.e. the sky noise dominates). If we expand this out, the only term that depends on the position of the star is  $\sum_i O_i P_i$ , i.e. the convolution<sup>12</sup> of the PSF with the data. Furthermore, the flux in the star is also proportional to  $\sum_i O_i P_i$ , so the PSF-convolved image is proportional to the signal-to-noise ratio for a stellar detection at any point in the image.

MLE estimates are known to be optimal in the large-sample limit (e.g. Lupton (1993)), so the best way to find objects of known profile in an image is to smooth with that profile, and then find all pixels above a threshold level; that threshold is directly related to the desired signal-to-noise ratio threshold. Moreover, a single pixel detection is perfectly acceptable, it's simply an object close to the detection threshold.

As the area in the smoothed image tells us more about significance than extent, how should we decide which pixels belong to a detected object? As the detection algorithm assumes that we know the object's profile, the answer's immediate: the object's consists of all pixels where the profile is

---

<sup>12</sup>actually the correlation, which is equivalent to correlation with the mirror-image of the PSF. The distinction is only important if the PSF lacks a centre of symmetry

non-zero, centred at the highest point in the smoothed image.

In more detail *Photo*'s algorithm is:

- Smooth with the PSF
- Detect groups of connected pixels above a threshold; these are known as ‘objects’.
- Grow each object approximately isotropically by an amount equal to approximately the FWHM of the PSF. For a single pixel detection this results in an area approximately equal to the seeing disk; for more extended objects it merely enlarges the detected region a little. If objects overlap during this operation they are merged.

We then replace any pixel in an object by the background level (with suitable noise added), bin the image  $2 \times 2$ <sup>13</sup>, repeat this operation (using the original PSF parameters, i.e. we’re using a filter twice as large as the true PSF), and merge the original set of objects with those detected in this pass. Then bin  $2 \times 2$  once more and repeat the whole procedure.

This operation is carried out independently in the 5 SDSS bands, and the objects detected in each band are also merged together.

(**XXX** *Describe BRIGHT v. faint processing*)

One of the SDSS data products is the ‘atlas images’ which contain all detected pixels, and another is the  $4 \times 4$  binned image with all detected pixels removed. Each pixel in this image corresponds to a  $4 \times 4$  ‘superpixel’ in the unbinned image. To produce the atlas images we take the merged objects and grow their boundaries out to the edges of the superpixels; this means that either a superpixel contains no pixels assigned to an object, or else the entire superpixel is included in an object. These objects, the union of all groups of detected pixels (as isotropically grown), detected either unbinned or binned in any band, and expanded to superpixel boundaries, define the pixels included in the atlas images.

(**XXX** *Known objects?*) (**XXX** *Sky objects*) (**XXX** *Merge objects*)

## 6. Determining the Centres of Objects

### 6.1. Introduction

A brief introduction to *Photo*'s centering algorithm is given in Pier et al. (2003).

As discussed in Sec. C, the real PSF is the combination of a Kolmogorov term and the effects of aberrations and telescope misalignment. In this section we shall generally approximate the PSF

---

<sup>13</sup>as all data regions are represented as integers, we also scale up by a factor of 2 to preserve signal-to-noise

as a single Gaussian  $G(\alpha)$  (Eq. C1) and occasionally comment upon the consequences of a second Gaussian component. Sec. 6.3 discusses how we deal with the errors resulting from PSFs that are not only non-Gaussian, but are also skewed.

For a star much brighter than the sky, the optimal way to estimate its position is to simply calculate the centroid of its light, that is its center of ‘gravity’; after all, that is what you mean by its true position (see Appendix D for a discussion of how to actually calculate this centroid for pixellated data). Unfortunately, this is a very noisy estimate for most objects in the sky, and we are forced to use other algorithms. We shall return to true centroids in Sec. 6.3.

As discussed in Sec. 5.2 if you want to estimate the position of a faint star, you should smooth the frame with the PSF<sup>14</sup> and look for the position of the maximum in that smoothed image.

## 6.2. Astrometric Centering: Gaussian Quartic Interpolation Schemes

Independent of the question of the optimal choice of smoothing filter, we must also find a good estimate of the centre of objects in the smoothed, pixellated, image. A common approach is to fit a parabola to the central few pixels of the object, and take the peak of that parabola (or paraboloid) to be the centre. Unfortunately, this approach leads to systematic positional errors as the true centre of the object moves relative to the pixel grid. For critically sampled images, we find effects of order of 0.025 – 0.04 pixels (10 – 15 mas for the SDSS’s 0.400 arcsecond pixels).

While random errors of this sort are barely acceptable for the measurements of individual objects (we expect at best to have 30 mas errors from atmospheric effects) they are not acceptable for either the secondary or primary astrometric standards, which define the coordinate system. The next section presents an interpolation scheme that essentially removes these systematics.

### 6.2.1. Finding an Object’s Center in One Dimension

Let us start by looking at the problem in one dimension. Consider a function  $f(x)$  sampled at  $x = 0, \pm 1, \pm 2, \dots$ ; let us take the point  $x = 0$  to be a local maximum of our sampled function, and consider how to estimate the position  $x$  of the maximum of  $f$  itself. We know  $f_0 \equiv f(x = 0)$ , and  $f_{\pm} \equiv f(x = \pm 1)$  where  $f_0$  is at least as large as the other two.

If  $f$  is a Gaussian  $G(\beta)$  (e.g. for a PSF  $G(\alpha)$  smoothed with itself,  $\beta^2 = 2\alpha^2$ ), we have

$$f_0 = A \exp\left(-\frac{x^2}{2\beta^2}\right) = A \left(1 - \frac{x^2}{2\beta^2} + \frac{x^4}{8\beta^4} + \dots\right)$$

---

<sup>14</sup>actually, the PSF inverted through its centre

and similar equations for  $f_{\pm}$ . Neglecting the terms above fourth order, we see that

$$\frac{\beta^2 (2f_0 - (f_+ + f_-))}{A} = 1 - \frac{1+6x^2}{4\beta^2} \quad (8)$$

and

$$\frac{\beta^2(f_+ - f_-)}{2A} = x - \frac{x(1+x^2)}{2\beta^2}. \quad (9)$$

Dropping the last term on the right hand sides of these equations we arrive at the standard quadratic estimates for  $A$ ,  $x$ , and  $\beta$ :

$$\begin{aligned} A &= f_0 + \frac{s^2}{2d}, \\ \beta^2 &= \frac{A}{d}, \\ x &= \frac{s}{d}. \end{aligned} \quad (10)$$

where  $s \equiv (f_+ - f_-)/2$  is the mean slope across the object and  $d \equiv 2f_0 - (f_+ + f_-)$  is the curvature.

Substituting these expressions into Eq. 8 leads to an improved estimate of  $\beta^2$ :

$$\beta^2 = \frac{A}{d} - \frac{1 + 6(s/d)^2}{4};$$

substituting this expression for  $\beta^2$  on the left hand side of Eq. 9 leads to

$$\begin{aligned} x &= \frac{s}{d} + \frac{s}{4A} (1 - 4x^2) \\ &\equiv \frac{s}{d} \left( 1 + \frac{d}{4A} (1 - 4(s/d)^2) \right) \end{aligned} \quad (11)$$

Note that the correction term has the correct ‘symmetry’ properties vanishing at  $x = 0$  ( $s = 0$ ) and  $x = \pm 1/2$  ( $s = \pm d/2$ ).

Now, in reality, the PSF is not exactly Gaussian, and for the parameters of interest ( $\alpha^2 \approx 1$ ;  $\beta^2 \approx 2$ ) the fourth-order and higher terms are not *so* small, so the amplitude of the correction term for best results may not be exactly what this simple development would predict. In fact, a series of simulations suggest that one needs a term about a third larger to produce the best astrometric results; i.e.

$$x = \frac{s}{d} \left( 1 + k \frac{d}{4A} (1 - 4(s/d)^2) \right) \quad (12)$$

with  $k \approx 4/3$ . This correction arises because the second-order estimate for  $\beta^2$  (Eq. 10) is too large by about this factor. Some of this is due to the fact that the amplitude  $A$  one measures is contaminated by the much flatter large Gaussian component of our assumed two-component PSF, so its effective value is about 20 percent smaller than the whole amplitude; this effect would lead to a value  $k \approx 1.2$ . The remaining  $\Delta k \approx 0.15$  is presumably due to the breakdown of our expansions. The correction term has a maximum value of  $0.024k/\beta^2$ , so changing  $k$  from 1 to  $4/3$  changes the correction by only about 3 – 4 mas for a pixel size of 0.4” and  $\alpha \approx 1$ ; although this is almost negligibly small we have adopted a value of  $k = 1.33$  for the SDSS reductions.

### 6.2.2. Extension to Two Dimensions

The extension of this interpolator to two-dimensional images is not completely straightforward. In those cases in which the images are round or elongated in such a manner that the major axes of the elongation are along the coordinate axes, a simple extension of the technique using either the x and y stripes through the maximum pixel in the smoothed image or suitably interpolated stripes to better ‘center’ the data on the supposed real center works satisfactorily. In general, however, this condition is not met for imperfect images, and they are elongated at some arbitrary angle to the axes. The *ad hoc* cross-stripe methods fail quite badly for these images, incurring errors of up to 15 mas or so; the errors can easily exceed the quadratic ones, even for images which are quite accurately Gaussian near the peak.

It is the case, however, that a two-dimensional Gaussian of arbitrary ellipticity and inclination has the property that any one-dimensional cut through it is also a Gaussian, which leads immediately to a two-dimensional extension which works very well indeed. Consider a point which is a maximum in the smoothed image and its eight neighbors. We may apply the algorithm of the previous section to each row of this  $3 \times 3$  region (since they are approximately three values of some Gaussian) to find three maxima:  $mx_-$ ,  $mx_0$ , and  $mx_+$ ; likewise the maxima of the three columns are  $my_-$ ,  $my_0$ , and  $my_+$ . The  $mx$ ’s lie on a curve along which the two-dimensional maximum must lie, as do the  $my$ ’s, and the desired maximum must lie at the intersection of those curves. In simulated SDSS images we found that the sets of three maxima lay essentially on straight lines (which is strictly the case for a real two-dimensional Gaussian), and one could either use the least-squares straight line and find the intersection analytically, or, as we chose to do, allow the three points to define a quadratic and use a simple iteration to find the intersection.

### 6.2.3. Error Estimates

We can now proceed to estimate the statistical errors in our quartic estimator for the object’s center (Eq. 11). Linearising the expression with respect to  $d$  and  $s$  (and therefore implicitly assuming that their errors are small enough that higher terms in the expansion are irrelevant), we can find  $x$ ’s variance in terms of the variances of  $d$  and  $s$  (it is easily shown that their covariance vanishes):

$$var(x) = var(s) \left( \frac{1}{d} + \frac{k}{4A} \left( 1 - \frac{12s^2}{d^2} \right) \right)^2 + var(d) \left( \frac{s}{d^2} - \frac{k}{4A} \frac{8s^2}{d^3} \right)^2 \quad (13)$$

The next two sections deal with the calculation of these variances in the two limits of objects much fainter than, and much brighter than, the sky. These two contributions are independent, and may simply be added together to arrive at the total variance for any given object.

*Objects much fainter than the sky*

Let us assume that the original picture has pixel values  $p_{\mathbf{i}}$ ; the subscript  $\mathbf{i}$  is of course a two-dimensional object  $(x, y)$ . We assume that the noise in the image is such that the noise in distinct pixels is independent. The smoothed image in which one searches for maxima is given by

$$P_{\mathbf{i}} = \sum w_{\mathbf{i}-\mathbf{j}} \times p_{\mathbf{j}}.$$

We assume that the weights are normalized so that  $\sum w_{\mathbf{i}} = 1$ ; in that case if we adopt a  $G(\alpha)$  smoothing filter, we have

$$\text{var}(P_{\mathbf{i}}) = n^2 \sum (w_{\mathbf{k}})^2 = \frac{n^2}{4\pi\alpha^2},$$

where  $n$  is the per pixel RMS noise (cf. Eq. 10.3.1). We shall initially assume that this is constant — i.e. that we are dealing with images sufficiently faint that the background noise dominates the photon noise from the object. We will deal with the case that the noise is dominated by photons from the star in Sec. 6.2.3. We consider the errors in one dimension at a time, and assume that the center has been arrived at by the single quartic interpolation of Sec. 6.2. The full two-dimensional analysis is complex, and our Monte-Carlo simulations (Sec. 6.2.4) indicate that this simple treatment is adequate.

The  $P_k$  are, of course, *not* independent;

$$\text{cov}(P_i, P_k) = \frac{n^2}{4\pi\alpha^2} \exp\left(-\frac{(i-k)^2}{4\alpha^2}\right) \quad (14)$$

Thus the variances of the key quantities are

$$\text{var}(s) \equiv \text{var}((f_+ - f_-)/2) = \frac{n^2}{8\pi\alpha^2} \left(1 - e^{-1/\alpha^2}\right), \quad (15)$$

$$\text{var}(d) \equiv \text{var}(2f_0 - (f_+ + f_-)) = \frac{n^2}{2\pi\alpha^2} \left(3 - 4e^{-1/4\alpha^2} + e^{-1/\alpha^2}\right). \quad (16)$$

The relative importance of these terms is shown in Fig. 12, where a Monte-Carlo simulation of the 1-dimensional centering algorithm is illustrated. The top panel shows the bias, the bottom the various terms contributing to the variance; in both cases the simulation is shown as a set of open squares.

It is instructive to consider the case of stars (i.e.  $\beta^2 = 2\alpha^2$ ) in the quadratic approximation of Eq. 10 (i.e.  $A = d\beta^2$ , and  $k = 0$  in Eq. 13). In this approximation, we may write the center's variance in terms of the signal-to-noise ratio  $\Sigma_F \equiv A\sqrt{4\pi\alpha}/n$  of the total flux;

$$\text{var}(x) = \frac{2\alpha^4}{\Sigma_F^2} \left( \left(1 - e^{-1/\alpha^2}\right) + 4 \left(\frac{s}{d}\right)^2 \left(3 - 4e^{-1/4\alpha^2} + e^{-1/\alpha^2}\right) \right)$$

for  $\alpha = 1$  (a FWHM of about 1arcsec for the SDSS 0.400 arcsecond pixels),

$$\text{var}(x) = \frac{1}{\Sigma_F^2} (1.12^2 + 1.42^2 x^2)$$

where  $|x| \equiv |s/d| \leq 0.5$

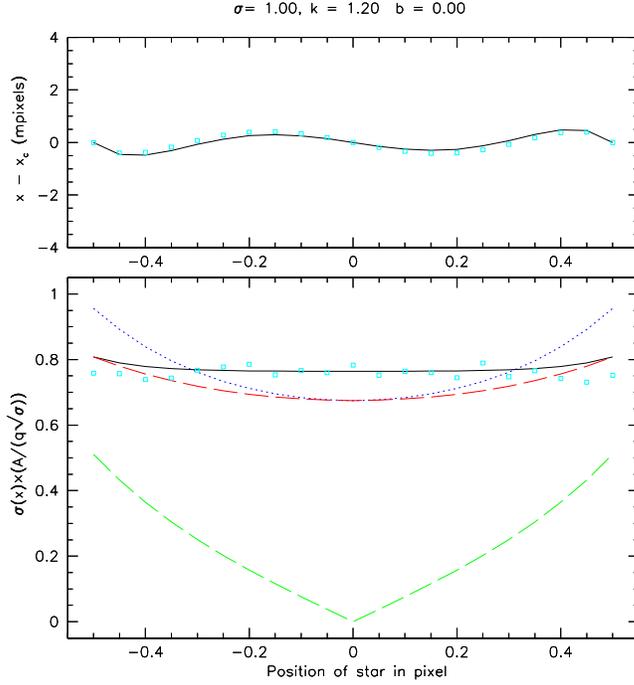


Fig. 12.— The results of 1000 1-dimensional Monte-Carlo realisations are shown as open (cyan) squares. The object was a pure Gaussian  $G(\alpha = 1)$ , and the smoothing kernel was also taken to be a  $\alpha = 1$  Gaussian. The noise was Normal with a constant variance. The top panel shows the bias in millipixels, using the quartic formula of Eq. 13 with  $k = 1.2$ . The bottom panel shows the variance, scaled by  $A/(n\sqrt{\alpha})$ . The two lines drawn with long dashes are  $\sqrt{\text{var}(s)/d^2}$  (red; non-zero at the origin) and  $\sqrt{(s/d)^2(\text{var}(d)/d^2)}$  (green; zero at the origin). The dotted (blue) line is the error of the quadratic approximation (i.e. the RMS sum of the two preceding lines), while the solid line is the estimated error of the quartic formula.

*Objects brighter than the Sky*

Bright objects no longer have the convenient property that each pixel’s variance is simply  $n^2$ , rather it’s equal to the total flux in the pixel. If the *unsmoothed* object is a  $G(\tau)$  Gaussian, the simple expression Eq. 14 for the covariance of the smoothed image is replaced by

$$\text{cov}(P_i, P_k) = \frac{a\tau^2}{2\pi\alpha^2(\alpha^2 + 2\tau^2)} \exp\left(-\frac{(i^2 + k^2)(\alpha^2 + \tau^2) - 2\tau^2 ik}{2(\alpha^2 + 2\tau^2)}\right).$$

For the case  $\tau = \alpha$ , this reduces to:

$$\text{cov}(P_i, P_k) = \frac{a}{6\pi\alpha^2} \exp\left(-\frac{i^2 + k^2 - ik}{3\alpha^2}\right) \quad (17)$$

where  $a$  is the amplitude of the *unsmoothed* image. While it would be possible to calculate errors based upon the actual position of an object relative to the pixel boundaries and keeping track of  $\tau$  and  $\alpha$  separately, the Monte-Carlo simulations of Sec. 6.2.4 show that this is unnecessary.

For an object centred in a pixel,  $d$  and  $s$ ’s variances are:

$$\text{var}(s) \equiv \text{var}((f_+ - f_-)/2) = \frac{a}{12\pi\alpha^2} \left( e^{-1/3\alpha^2} - e^{-1/\alpha^2} \right), \quad (18)$$

$$\text{var}(d) \equiv \text{var}(2f_0 - (f_+ + f_-)) = \frac{a}{3\pi\alpha^2} \left( 2 - 3e^{-1/3\alpha^2} + e^{-1/\alpha^2} \right). \quad (19)$$

These are, of course, to be added to the variances given by Eq. 16 and Eq. 15.

*6.2.4. Monte-Carlo Simulations*

We carried out a set of Monte-Carlo simulations to test the 2-dimensional centering algorithm of Sec. 6.2.2. Each test was repeated 2500 times, for stars of the form  $I_0/1.1 \times (G(\tau) + 0.1G(3\tau))$  randomly positioned within the pixels; the noise was Gaussian with variance equal to the pixel intensity. Each simulation used a  $N(\alpha)$  smoothing filter. The other properties of simulations are given in Table 6.2.4; it will be seen that we probe both the sky- and object-dominated regimes.

The table shows that the biases are small (a maximum of 0.014 pixels), and that the variances given by the sum of Eqns. 15 and 18 are reasonable accurate; very accurate if the correct smoothing scale is chosen (i.e.  $\alpha = \tau$ ).

**(XXX Do we need to discuss  $\alpha \neq \tau$  further?)**

**6.3. Correcting for non-Gaussian, Asymmetrical, PSFs**

In reality, the combination of time-varying seeing and non-perfect optics delivers a PSF that varies with position on the CCD; this variation of the PSF causes astrometric offsets of order

Test	sky	$I_0$	$\tau$	$\alpha$	bias <sub>row</sub>	bias <sub>col</sub>	$\chi_{\text{rms, row}}$	$\chi_{\text{rms, col}}$
RA0	100	10000	1.0	1.0	-0.000	-0.000	1.05 (1.06)	1.06 (1.04)
RA2	100	10000	1.0	0.0	-0.000	-0.000	0.98 (0.98)	0.97 (0.98)
RA3	100	1000	1.0	1.0	-0.000	-0.000	0.93 (0.99)	1.01 (1.01)
RA4	100	100	1.0	1.0	-0.000	0.000	0.95 (0.96)	0.96 (0.97)
RA5	400	100	1.0	1.0	-0.009	0.002	1.00 (1.00)	0.95 (0.97)
RA6	900	100	1.0	1.0	0.007	-0.002	1.04 (1.07)	1.01 (1.02)
RA7	1600	100	1.0	1.0	-0.013	-0.013	1.07 (1.12)	1.02 (1.12)
RA8	4000	1000	1.0	1.0	0.000	0.001	0.99 (0.97)	1.01 (0.98)
RA9	100	10000	1.0	1.2	-0.000	-0.000	1.12 (1.13)	1.07 (1.07)
RA10	100	10000	1.0	1.5	-0.000	-0.000	1.15 (1.14)	1.08 (1.09)
RE0	100	10000	2.0	2.0	0.000	0.000	1.01 (1.03)	1.06 (1.03)
RE7	1600	100	2.0	2.0	-0.007	0.014	1.04 (1.01)	0.97 (1.00)
RE9	100	10000	2.0	2.4	0.000	-0.000	1.06 (1.06)	1.07 (1.07)
RE10	100	10000	2.0	3.0	-0.000	-0.000	1.04 (1.06)	1.06 (1.05)

Table 4: The results of Monte-Carlo tests of the 2-dimensional centering algorithm. The bias (in pixels) is the difference between the true position and the measured position. Two values are quoted for the RMS value of  $\chi$  (i.e. the ratio of the measured error to the estimated error) in pixels. The first value is estimated from the semi-interquartile range (and converted to an RMS on the assumption of Gaussianity); the second is a true RMS.

Fig. 13.— Histograms of the astrometric biases in the column centroid in each of the 30 CCDs of the SDSS camera, sampled every hour from all the data in DR4 (Adelman et al. 2005). Each panel is normalised to unit height, and the seeing is divided into four seeing bins (red:  $> 1.7$ , yellow:  $1.7 - 1.5$ , green:  $1.5 - 1.25$ , blue:  $< 1.25$ ).

100mas. Some of these offsets result in apparent field distortions which can be removed during standard astrometric processing, but some are of higher order and are best handled by explicit modelling.

As discussed in Sec. 2.2, *Photo* knows the PSF at the position of each star, and we can use this to correct for these biases. The estimate of the PSF has a sufficiently high signal-to-noise ratio that we can simply calculate its centroid; we can then apply the theory of Sec. 6.2.2 to arrive at a second, in general different, estimate of its centre. The difference between the two gives an estimate of the bias in the center due both the limitations of the quartic theory presented here and to departures of the PSF from the assumed form. This estimate of the needed correction is then applied to the measured center of the, much fainter, real object and the debiased position reported. The histograms of the applied column corrections, sampled every hour and at 12 positions across each CCD, for all the SDSS photometric CCDs are given in Fig. 13 (The corresponding figure for row centroids is similar).

Comparing the amplitudes of the colored histograms shows that the seeing is significantly worse in the u than in other bands, and that the image quality degrades away from the camera center. Close inspection will reveal that the poor-seeing data has bias corrections systematically offset from the better data, as would be expected as the atmospheric- and optical-contributions to the image quality varied in relative importance.

## 7. Merging per-band detections

(**XXX** *Write me*)

## 8. Deblending Overlapping Objects

(**XXX** *Refer to (Lupton 2005)*)

## 9. Extracting Radial Profiles

(**XXX** *Intro to measure objects*)

10	0.004	0.003	-0.002	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
9	-0.007	-0.004	0.004	0.000	-0.001	0.000	0.000	-0.001	0.000	0.000	0.000
8	0.012	0.007	-0.007	0.000	0.001	-0.001	0.001	0.000	-0.001	0.000	0.000
7	-0.019	-0.011	0.013	0.001	-0.001	0.000	-0.001	0.001	0.000	0.000	-0.001
6	0.032	0.018	-0.020	-0.001	0.002	-0.001	0.000	-0.001	0.001	0.001	0.000
5	-0.063	-0.035	0.032	0.001	-0.004	0.002	-0.001	0.000	-0.001	0.000	0.000
4	0.260	0.142	-0.060	-0.005	0.005	-0.004	0.002	-0.001	0.001	0.001	-0.001
3	1.042	1.011	0.637	0.017	-0.005	0.001	-0.001	0.001	0.000	0.000	0.000
2	1.009	0.947	1.146	0.637	-0.060	0.032	-0.020	0.013	-0.007	0.004	0.004
1	1.019	1.016	0.947	1.011	0.142	-0.035	0.018	-0.011	0.007	-0.004	-0.004
0	0.956	1.019	1.009	1.042	0.260	-0.063	0.032	-0.019	0.012	-0.007	-0.007
	0	1	2	3	4	5	6	7	8	9	

Table 5: Weights for an  $R = 3.75$  pixel circular aperture.

9	0.04	0.04	0.03	-0.02	-0.01	0.01	0.00	0.00	0.00	0.00	0.00
8	-0.09	-0.08	-0.06	0.02	0.02	-0.01	0.01	0.00	0.00	0.00	0.00
7	0.49	0.42	0.22	-0.03	-0.05	0.02	-0.01	0.00	0.00	0.00	0.00
6	1.09	1.10	1.03	0.81	0.23	-0.06	0.02	-0.01	0.01	0.00	0.00
5	0.96	0.93	0.99	1.03	1.08	0.37	-0.06	0.02	-0.01	0.01	-0.01
4	1.03	1.05	1.01	0.98	0.96	1.08	0.23	-0.05	0.02	-0.01	0.01
3	0.94	0.99	0.95	1.06	0.98	1.03	0.81	-0.03	0.02	-0.02	0.01
2	1.05	1.01	1.04	0.95	1.01	0.99	1.03	0.22	-0.06	0.03	-0.02
1	0.99	0.95	1.01	0.99	1.05	0.93	1.10	0.42	-0.08	0.04	-0.02
0	1.06	0.99	1.05	0.94	1.03	0.96	1.09	0.50	-0.09	0.04	-0.03
	0	1	2	3	4	5	6	7	8	9	10

Table 6: Weights for an  $R = 7.0$  pixel circular aperture.

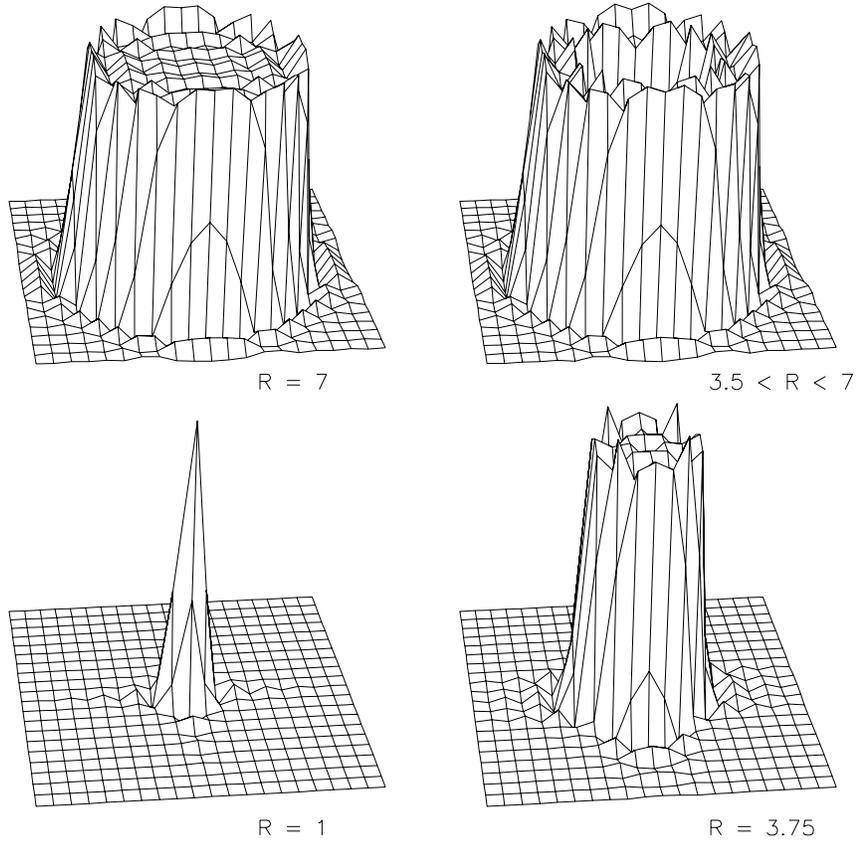


Fig. 14.— The coefficients for exact integration over a set of four circular apertures, with radii  $R = 1$ ,  $R = 3.75$  (see Table 5),  $R = 7$  (see Table 6), and an annulus  $3.5 < R < 7$ ; all radii are in pixels.

### 9.1. Photometry

If our data is band limited we can use the very well-known sinc interpolation formula (e.g. Bracewell (2000)) to evaluate the (pixel-convolved) intensity at any point in the image:

$$f(\xi) = \sum_{i=-\infty}^{\infty} f(i) \frac{\sin((\xi - i)\pi)}{(\xi - i)\pi}. \quad (20)$$

We can then use this result to evaluate the intensity on a grid with the object's center at some desired point, say  $(0, 0)$ ; i.e. we can shift the image to be centered on a pixel. After this operation let us write the intensity at pixel  $i, j$  as  $f_{ij} \equiv f(i, j)$ .

This process is exact. We can then evaluate the total flux  $F$  within an aperture of radius  $R$  as

$$F = \int_{x^2+y^2 < R^2} f(x, y) dx dy \quad (21)$$

$$\begin{aligned} &= \int_{x^2+y^2 < R^2} \sum_{ij} f_{ij} \frac{\sin((x - i)\pi)}{(x - i)\pi} \times \frac{\sin((y - j)\pi)}{(y - j)\pi} dx dy \\ &= \sum_{ij} f_{ij} \int_{x^2+y^2 < R^2} \frac{\sin((x - i)\pi) \sin((y - j)\pi)}{(x - i)\pi(y - j)\pi} dx dy \\ &\equiv \sum_{ij} f_{ij} C_{ij} \end{aligned} \quad (22)$$

where

$$C_{ij} \equiv \int_{x^2+y^2 < R^2} \frac{\sin((x - i)\pi) \sin((y - j)\pi)}{(x - i)\pi(y - j)\pi} dx dy. \quad (23)$$

The integral defining  $C_{ij}$  cannot be evaluated analytically, but it is independent of the image being measured, so it can be evaluated numerically and then tabulated. Tables 5 and 6 give the values of  $C_{ij}$  in the first quadrant for  $R = 3.75$  (the SDSS 3" aperture) and  $R = 7$ ; these weights are illustrated in Fig. 14.

### 9.2. Measuring Surface Brightnesses

*Photo* measures the radial profile of every object by measuring the flux in a set of annuli, spaced approximately exponentially (successive radii are larger by approximately 1.25/0.8); the outer radii and areas are given in Table 9.2. Each annulus is divided into 12 30° cells. For the inner 6 annuli (to a radius of about 4.6asec) the flux in each cell is calculated by exact integration over the pixel-convolved image after shifting the object by sinc-interpolation (Sec. 9) so that its centre lies in the centre of a pixel;<sup>15</sup> for larger radii the cells are defined by a list of the pixels

---

<sup>15</sup>We actually use a cosine-bell to taper the sinc coefficients, i.e. we multiple Eq. 20 by  $(1 + \cos(\pi(\xi - i)/L))/2$  for  $|\xi - i| < L$  and 0 otherwise; we have adopted  $L = 6$ . This is similar in spirit to Lanczos filters, which taper the

which fall within their limits (these lists are predetermined, based on the sub-pixel centre of the object rounded to  $1/32$  of a pixel). Usually the straight mean of the pixel values is used, but for cells with more than 2048 pixels a very mild clip is applied (only data from the first percentile to the point  $4.0\sigma$  above the median is used). This clipping produces a negligible shift in the mean for Poisson data, but guards against wildly non-Gaussian outliers such as unrecognised cosmic rays and fragments of bleed trails.

One might worry about the noise properties of the pixel-centered image; after all, noise is uncorrelated from pixel to pixel, and thus can have features that are sharper than the PSF — i.e. it breaks the band limit (Sec. 2.1). Remarkably, if the noise per pixel is constant (as is the case for objects with surface brightness appreciably lower than the sky), the noise in the sinc-interpolated image has the same variance as in the input image, and the noise is uncorrelated from pixel to pixel; for Gaussian noise, this implies that the noise in each pixel is independent.

### 9.3. Radial Profiles

Given a set of cells, *Photo* can measure the radial profile. If the mean flux within each of the cells in an annulus are  $M_j (j = 1, \dots, 12)$ , it calculates a point on the profile (‘profMean’) as

$$P_i = \frac{1}{12} \sum_{j=1}^{j=12} M_j$$

The error of this quantity (‘profErr’) is a little trickier. If we knew that the object had circular symmetry, we would estimate it as the variance of the  $M_j$  divided by  $\sqrt{12}$ . Unfortunately, in general the variation among the  $M_j$  is due to both noise and the radial profile and flattening of the object. To mitigate this problem, we estimate the variance as

$$\text{Var}P_i = \frac{4}{9} \times \frac{1}{12} \sum_{j=1}^{j=12} (M_j - 0.5 * (M_{j-1} + M_{j+1}))^2 \tag{24}$$

where obviously we interpret ‘ $j \pm 1$ ’ modulo 12, and where the factor  $4/9$  would be correct if all the  $\langle M_j \rangle$  were equal. This use of a local mean takes out linear trends in the profile around the annulus, and results in an estimate of the uncertainty in the profile that is a little conservative, but which includes all effects. If for some reason you wanted to know the error due to photon noise alone, this could be calculated from the  $P_i$  and the known gain of the CCD.

In practice, *Photo* doesn’t extract the profile beyond the first point that the surface brightness within an annulus falls to (or below) zero; if any points *do* fall below zero, the BAD\_RADIAL flag bit is set (Sec. 1.1).

---

coefficients with sinc functions rather than cosines.

## 10. *Photo*'s Flux Measures

(**XXX** *luptitudes and (non-)procedure for upper limits*)

(**XXX** *Calibration*)

### 10.1. Fiber Magnitudes

The SDSS spectrographs are fed with 3" optical fibers (York et al. 2000), and there one of the *Photo* outputs is therefore the flux within a 1.5" aperture. This is calculated in the way described in Sec. 9, and the fiber magnitude coefficients are given in Table 5 and Fig. 14.

This flux is naturally strongly dependent on the seeing, and we therefore convolve the image to a canonical value before measuring the fiber magnitude (we use 2.0", but it is in fact adjustable for each photometric bands; Sec. A.3). This is done by estimating the per-object seeing as  $\sqrt{(M\_rr\_cc\_psf/2)}$  (see Sec. 12.2), and if this exceeds 2.0, convolving the image with a  $G'(\sqrt{(2^2 - M\_rr\_cc\_psf/2)})$  Gaussian.

Unlike all other *Photo* flux measures, the fiber magnitude for a child object does *not* reflect our best estimate of the properties of the child. Rather, it is calculated from the *non*-deblended image, at the position of the child. This allows its use in estimating the total flux that would be expected to fall down the fiber, and to allow us to avoid taking spectra of faint objects detected near very bright stars.

### 10.2. The Calculation of Petrosian Quantities by *Photo*

Details of *Photo*'s calculation of Petrosian quantities are given in Appendix A of Strauss et. al. (2002).

#### 10.2.1. Errors in Petrosian Quantities

A comparison on objects imaged more than once allows us to discover how realistic these error estimates are; Table 10.2.1 lists the RMS value of  $\chi$ , the ratio of the measured differences between the two values and the RMS sum of their quoted errors. In each case, only objects with no PETRO flags set, and with the relevant errors available in each run (see Sec. 1.1 for a discussion of *Photo*'s flags). (**XXX** *Need more discussion*).

### 10.3. PSF magnitudes

#### 10.3.1. Introduction

As is well known, the Maximum Likelihood Estimator (MLE) for an object’s flux is given by a sum over the pixels in the image:

$$f_{MLE} \equiv \frac{\sum_i P_i O_i / \sigma_i^2}{\sum_i P_i^2 / \sigma_i^2} \quad (25)$$

where the  $P_i$  are the object’s true profile (normalised to have unit sum); the  $O_i$  are the observed intensities, with the sky background subtracted; and the  $\sigma_i^2$  are the errors in  $O_i$ . (~~XXX~~ *There’s overlap with Sec. 5.2*)

Unfortunately, in reality, we never know even a star’s profile exactly, so the expectation value of  $f_{MLE}$  is not exactly equal to the total flux in the object. For studies of stars this might appear not to matter, as we’re only interested in relative photometry, but in fact the inclusion of  $\sigma^2$  in Eq. 25 means that we will make brightness-dependent errors. For very bright stars  $\sigma^2 \propto P$ , while for very faint stars  $\sigma^2$  is a constant, so systematic discrepancies between  $P_i$  and  $O_i$  are weighted differently for different stars.

Fortunately we can easily avoid this problem by *always* setting  $\sigma^2 \equiv n^2$  where  $n^2$  is the per-pixel variance of empty parts of the frame (a combination of read noise, dark current, and photon noise from the sky). For bright stars the loss of efficiency due to using an incorrect noise model is irrelevant as the flux errors will be dominated by systematics, and for faint objects the per-pixel variance is indeed constant. We should thus modify our MLE magnitude to

$$f_{MLE'} \equiv \frac{\sum_i P_i O_i}{\sum_i P_i^2} \quad (26)$$

For stars,  $P$  is of course the point spread function, so these MLE fluxes are often referred to as PSF fluxes. For a Gaussian PSF  $G(\alpha)$  the PSF flux has variance

$$4\pi\alpha^2 n^2$$

if the noise due to the object itself can be neglected; in other words it has the same noise as an aperture of radius  $2\alpha$  or only about 0.8 arcsec in 1 arcsecond seeing. Furthermore, Eq. 26 is an optimal measure of an object’s flux, so errors in the  $P_i$  only increase  $f_{MLE'}$ ’s variance by amounts that are second order in the errors in the assumed profile.

#### 10.3.2. Calibrating TDI Data

TDI data is different from pointed data in two ways: the spatial variation of the PSF is usually considerably greater, and the photometric standard stars don’t appear on the same images as the objects being calibrated.

This asynchronous calibration means that we have to take great care to remove all seeing-dependent terms from the measured PSF magnitudes, as terms that would normally have no effect can be important.

### 10.3.3. PSF Magnitudes in Photo

As explained in Sec. 9.2, *Photo* has available the central part of every object’s image centred on a pixel, so we need only calculate the  $P_i$  for a centred PSF. We noted in Sec. 10.3.1 that it’s not important that the  $P_i$  be exactly equal to the true PSF, and we therefore adopt a single double-Gaussian model of the  $P_i$  for an entire frame.

Adopting a constant variance for each pixel removed magnitude-dependent biases, but in reality the PSF varies with position and time. We may correct for this by employing the PSF model from Sec. 2.2: At the position of each object we reconstruct the PSF from our KL expansion, and measure its PSF flux  $f_{MLE',KL}$  using that frame’s set of  $P_i$ s. We then measure some large aperture flux  $f_{aper,KL}$  for the reconstructed PSF. The reconstructed PSF is determined from a number of bright stars so both  $f_{MLE',KL}$  and  $f_{aper,KL}$  are essentially noise-free, so we may estimate our real object’s ‘PSF’ magnitude as

$$f_{PSF} \equiv f_{MLE'} \times \frac{f_{aper,KL}}{f_{MLE',KL}}. \quad (27)$$

This composite flux has the noise properties of an MLE estimate, combined with an aperture flux’s insensitivity to PSF variation. In practice, we have used ‘aperture 5’ of Sec. 9.2, with a radius of 7.5 pixels, 3 arcsec.

For sufficiently poor seeing (worse than around 2 arcseconds) we found that even this procedure doesn’t completely remove the effects of seeing from our photometry, as the apertures available from our KL expansion are not large enough to enclose all the variation in the PSF’s wings. To cover this case, we use the composite profile determined in Sec. 2.5 to further correct the magnitude of the previous paragraph to an even larger radius, ‘aperture 7’ of Sec. 9.2, with a radius of 18.6 pixels, 7.4 arcsec.

*(XXX Željko (or someone — me?) wrote the following. Is it correct? How does it match the previous paragraph that RHL just wrote)*

*(XXX Studies of stars measured twice under different conditions showed that the corrections required were well described by a simple function of the observed FWHM, and applying such a correction removed any remaining dependencies of the photometry on the seeing. )*

## 10.4. Fitting Models to Detected Objects

*Photo* fits a number of model profiles to each detected object in each band; these models are used in star/galaxy separation, and also in the morphological classification of galaxies. There is a considerable literature on fitting one- and two-dimensional profiles to galaxies, both in terms of bulge/disk decomposition and the choice of radial profile. For example, a spiral galaxy may be decomposed into a deVaucouleurs bulge and an exponential disk; or a sum of two exponential components; or maybe a Sérsic profile  $I \sim r^{1/n}$  is to be preferred (Sérsic 1968).

In the interests of computational efficiency, and because most objects observed in the SDSS don't have sufficient signal-to-noise and resolution to justify more detailed modelling, we have restricted ourselves to fitting three models: a PSF, a pure exponential disk (truncated beyond  $3 r_e$  in such a way that the profile goes to zero with zero derivative at  $4 r_e$ ) convolved with the PSF, and a deVaucouleurs profile (truncated beyond  $7 r_e$  in such a way that the profile goes to zero with zero derivative at  $8 r_e$ ) convolved with the PSF. Each of these galaxy models is specified by four parameters: the central intensity  $I_0$ , the effective radius  $r_e$ , the axis ratio  $a/b$ , and the position angle of the major axis,  $\phi$ .

In Sec. 10.4.2 we discuss a surrogate for a full-up Sérsic model fit.

### 10.4.1. Model Fitting

Fitting these models is a straightforward  $\chi^2$  minimisation problem, which we solve using the standard Levenberg-Marquardt (Marquardt 1963) algorithm. (**XXX** *Discuss why using a continuum method for data given on a grid may not be a smart idea. Discuss better alternatives (v5\_5 (.6?) etc.)*) Unfortunately, even when restricted to fitting pure exponential or deVaucouleurs profiles, the computational task is not trivial. Finding the minimum  $\chi^2$  requires solving a three-dimensional non-linear optimisation problem (solving for the overall intensity  $I_0$ , once  $(r_e, a/b, \phi)$  are known, is a linear problem). Each function evaluation requires building a model galaxy; convolving with the PSF at that point in the frame; and finally determining the value of  $\chi^2$  for that model by summing over all the pixels in the object.

If this were implemented naïvely it would be impractically slow<sup>16</sup>. We take a number of steps to improve performance:

- Rather than work directly in pixel space, we fit the models to the extracted cell profile (see Sec. 9). Furthermore, as all of our models have a two-fold axis of symmetry, we only need consider the average of pairs of cells placed symmetrically about the object's centre.

---

<sup>16</sup>The total time available for processing an object, including detection and measurement of all parameters as well as e.g. flatfielding the data and writing output files, is around 16ms on an 1GHz Pentium III.

- We model the PSF as a sum of Gaussians and a residual table  $R$ :

$$PSF = \alpha G(\sigma) + \beta (G(\tau) + bG(c\tau)) + R \quad (28)$$

where  $b$  and  $c$  are fixed (we adopt 0.1 and 3 respectively(~~XXX~~ cf Moffat)).

- We compute galaxy models of each type for a range of  $(r_e, a/b, \phi)$ , convolve each with a set of PSFs of the forms  $G(\sigma)$  and  $G(\tau) + bG(c\tau)$  for a set of values of  $\sigma$  and  $\tau$ , extract their profiles, and save the results to disk. A total of 3 coefficients determine our PSF model (it's the linear combination of 4 terms, but the overall normalisation is unimportant).
- All of our models have two planes of symmetry, so only 3 of the 12 cells within each annulus have distinct values. We can exploit this by writing the pre-extracted model profiles as Fourier series in which only the  $\cos(2r\theta)$  terms are non-zero. It is in fact these Fourier coefficients (for  $r = 0 \dots 5$ ) that are stored in the pre-calculated profiles. In calculating these Fourier series we use extracted profiles for models with  $\phi = 0$  and  $\phi = \pi/6$  in order to overcome the undersampling in angle of the cell profiles; note that using these pairs of models produces 6 model intensities within the  $\pi/2$  that we are sampling, corresponding to our choice of keeping 6 terms in the Fourier series.

With this Fourier expansion in hand, the profiles are a smooth function of  $\phi$ , and we can therefore use standard efficient techniques such as Brent's algorithm (Brent 1973; Forsythe et al. 1977) to find the optimal value of  $\phi$  for a given  $(r_e, a/b)$ ; this essentially reduces the dimensionality of the non-linear optimisation from three to two.

In Sec. 2.2 we discussed how we determine an accurate PSF at each point in the frame using a KL expansion. In order to use this information in fitting models (where the form of the PSF is of crucial importance), at the position of each object we determine the best representation of that KL PSF in the form of Eq. 28, where the  $\sigma$  and  $\tau$  are restricted to the values present in the pre-computed model tables:  $PSF_{KL} = PSF_{table} + R$ . We may then write

$$\text{model} = \text{model}_0 \otimes PSF_{KL} \quad (29)$$

$$\approx \text{model}_0 \otimes PSF_{table} + R \quad (30)$$

where  $\text{model}_0$  is the model galaxy above the atmosphere and  $\text{model}$  is that model after convolution with the PSF. In other words, we assume that the residual table  $R$  is only of importance to the core of the model, and that that core is adequately represented by a delta function. Note that this prescription is exact for a star, where the  $\text{model}_0$  is also a delta function and Eq. 30 reduces to  $\text{model} = PSF_{KL}$ .

Choosing the best model of the form Eq. 28 is not entirely straightforward, as there are no meaningful errors associated with our KL estimate of the PSF, Eq. 1. Not only is the ratio of the per-pixel errors due to photon noise in the object to those due to sky photons a function of a star's brightness, but the dominant error may be due to the approximations made in our KL approach.

We have chosen to only neglect the sky noise, and to regularise the fit with term dependent on the difference between the width of the true (actually KL) PSF, and the best representation in terms of sums of Gaussians; that is, we minimise the value of

$$\begin{aligned} X^2 &= \sum_i \frac{1}{\text{var}_i} \left( \text{PSF}_{KL,i} - \text{PSF}_{\text{table},i} \right)^2 + \Lambda \left( \tau_{KL}^2 - \tau_{\text{table}}^2 \right)^2 \\ &\equiv \sum_i \frac{1}{\text{var}_i} \left( \text{PSF}_{KL,i} - \sum_r w^{(r)} M_i^{(r)} \right)^2 + \Lambda \left( \tau_{KL}^2 - \frac{\sum_r w^{(r)} \tau_{(r)}^2}{\sum_r w^{(r)}} \right)^2 \end{aligned} \quad (31)$$

where  $\text{var}_i$  is the variance of the  $i^{\text{th}}$  pixel,  $\tau$  is an estimate of the PSF's width, and the sum in  $r$  runs over the PSF models  $M^{(r)}$  (with weights  $w^{(r)}$  and widths  $\tau_{(r)}$ ). Even this is not as simple as it looks, as the term  $\tau_{\text{table}}^2$  is nonlinear in the weights of the constituent models. Eq. 31 may be rewritten as

$$X^2 = \sum_i \frac{1}{\text{var}_i} \left( \text{PSF}_{KL,i} - \sum_r w^{(r)} M_i^{(r)} \right)^2 + \Lambda' \left( \sum_r w^{(r)} \left( \tau_{KL}^2 - \tau_{(r)}^2 \right) \right)^2 \quad (32)$$

where  $\Lambda' \equiv \Lambda / \sum_r w^{(r)}$ ; if we take the weights  $w^{(r)}$  to be known the problem is linear. In practice we iterate a few times, each time using the previous values of the  $w^{(r)}$ . There remains the question of the value of  $\Lambda$ ; we have found that setting it equal to  $X^2|_{\Lambda=0}$  produces satisfactory results.

The specification of the effective variance in each profile cell also requires some care. In addition to the photon noise from the object and sky, the model and data will differ due to errors in our understanding of the PSF, errors in the object's centre, ringing introduced by the sinc-shifting used to extract the central parts of the profile (Sec. 9.2), and features in real galaxies not included in our simplistic models (e.g. bulges, spiral arms, HII regions, and dust lanes). We accordingly increase the photon noise by terms intended to compensate for centering and sinc errors, and additionally add a term that ensures that the signal-to-noise ratio in a cell cannot exceed one hundred.

As discussed just above Eq. 26 in Sec. 10.3.1, if we used the true per-cell variance while estimating parameters, the e.g. axis ratio measured for a galaxy would depend on the signal-to-noise ratio; as for PSF magnitudes we therefore use only the photon noise contribution to the variance while fitting models. When calculating the flux errors, on the other hand, we use the true variances, augmented in the ways discussed in the previous paragraph. (**XXX errors for model magnitudes**)

The goodness-of-fit is reported as a Likelihood; in *Photo* v5\_3 and above it is also reported as a ln-likelihood to avoid problems with underflow for poor fits. For example, 'star\_L' is the probability that an object would have at least the measured value of  $\chi^2$  if it is really well represented by a PSF; the other \_L values are calculated similarly, taking into account the smaller number of degrees of freedom.

### 10.4.2. Model Magnitudes

Once  $(I_0, r_e, a/b)$  are known for a model of a given class (PSF, exponential, or deVaucouleurs) we can easily calculate the total flux; we refer to this as a ‘model’ flux. Unfortunately it isn’t so easy to find the appropriate aperture correction.

If the object is a star we’d like the ‘model’ magnitude to equal that measured any other way. *No* photometric measure of a star<sup>17</sup> ever attempts to include all of the photons entering the telescope; rather we attempt to include the *same* fraction of the light for all stars, and thus arrive at the correct ratios of their brightness to some standard. Let us call this chosen measure the ‘canonical’ flux. If we wish to use the same conversion from counts to flux for model as for any other type of magnitude, we need to apply an aperture correction to force the model flux to agree with the canonical flux. Our KL expansion (Sec. 2.2) allows us to reconstruct the PSF at the position of our object, and to determine the ratio of *its* canonical and model fluxes; multiplying our model flux by this ratio then achieves our goal. This is of course the same procedure as discussed in Sec. 2.4.1, and results in unresolved objects having identical PSF and model magnitudes.

The SDSS has photometry in 5 bands, and two ways of calculating model magnitudes present themselves: either using the best-fit model in each band, or using the model determined in some fixed band to calculate the magnitude in all bands. Both approaches have their virtues.

Fitting a model in each band gives the best estimate of the total flux in that band, but does *not* in general give the best estimate of the object’s colour as the flux contains errors due to both the photon noise in the image and to uncertainties in the model parameters.

Using the model from a fixed band generates better colours, but if the structure of the object is substantially different in different bands the magnitudes may be incorrect (of course, in this case there must be colour gradients, so the definition of the ‘correct’ colour is tricky). Indeed, to obtain good measures of an object’s colour the model need not be an especially good fit; even if it’s a poor representation of reality it can still provide an efficient statistical estimator of the flux in the object, analogous to the use of PSF magnitudes for measuring the flux of faint stars. As when measuring PSF fluxes, the weight given to each profile cell should be independent of the brightness of the object in order to avoid weighting different parts of the profile differently for bright and faint objects.

### Composite Model Magnitudes

We have been discussing fitting pure deVaucouleurs and exponential models to images, justifying this limited choice in terms of the computer resources available. Fortunately, there is an inexpensive

---

<sup>17</sup>with the possible exception of studies of Vega or *BD+17°4708* intended to provide an absolute calibration in Janskys

if rather ad-hoc way to estimate the relative contribution of the two idealised profiles, and we can use this value to estimate a total model magnitude.

Once we have models  $\text{deV}(I_0, r_e, a/b)$  and  $\text{exp}(I_0, r_e, a/b)$  fit to the data in any band, we can ask for the linear combination of these two models which provides the best fit:

$$\text{cmodel}(I_0, r_e, a/b) \equiv g_{\text{deV}} \text{deV}(I_0, r_e, a/b) + g_{\text{exp}} \text{exp}(I_0, r_e, a/b).$$

Let us call the ratio  $g_{\text{deV}}/g_{\text{exp}}$  ‘ $f_{\text{deV}}$ ’; I furthermore restrict it to lie in the range 0—1. This ratio  $f_{\text{deV}}$  is not, of course, a true bulge–disk decomposition as both of the input models have attempted to fit all components of the galaxy.

One application of  $f_{\text{deV}}$  is to estimate the total magnitude of the galaxy in a way that is less sensitive to the departures of the true profile from either of our idealised forms:

$$\text{flux}_{\text{cmodel}} \equiv f_{\text{deV}} \text{flux}_{\text{deV}} + (1 - f_{\text{deV}}) \text{flux}_{\text{exp}}$$

Figure (XXX *petro v. cmodel*) shows that the Petrosian flux for bright galaxies is very well correlated with the cmodel flux. (XXX *Discussion and numbers*).

## 10.5. Saturated Objects

(XXX *Write me*)

## 11. Object Classification

(XXX *Mostly to be written, see S/G paper*)

### 11.1. Cosmic Ray Rejection: pass II

The algorithm described in Sec. 4.3 deals successfully with the vast majority of cosmic rays, but when looking for extremely rare objects with flux in only a single band, for example T-dwarfs and very high redshift quasars, the contamination rate is still unacceptable (only about 20% are real (Fan et al. 2001)).

Fortunately, we have more information available about the PSF than we employed in Sec. 4.3, and we can use this to flag objects which *might* be cosmic rays by setting the `MAYBE_CR` bit. Because this is extra information which doesn’t affect other measured properties, we can afford to malign innocent stars and allow the consumer of the data to decide upon their reality. For example, they might choose to ignore `MAYBE_CR` for all objects detected in more than one band.

For every object detected by *Photo* we have the KL reconstruction of the PSF at that point (Sec. 2.2), and also a copy of the image data shifted to put the centre of the object at the centre of a pixel (Sec. 9.2). We apply a variant of the too-sharp-gradient algorithm (Eq. 7 of Sec. 4.3) to an  $11 \times 11$  pixel region ( $4.4'' \times 4.4''$ ) about the centre of the centred object. Specifically, we count the pixels for which

$$I(0) - c * N(I(0)) > \delta (I + cN(I))$$

where  $I(0)$  is the object’s central intensity,  $\delta \equiv \text{PSF}_{KL}(0)/\text{PSF}_{KL}$ , the noise  $N$  is the photon noise with an additional  $0.05I(0)$  added in quadrature, and  $c = 1.5$ .

If more than 3 pixels satisfy this condition, we consider the object to be a candidate cosmic ray and set the `MAYBE_CR` bit. We find in practice that many bright stars satisfy this condition, but that it provides a very useful guide to the reality of single-band detections.

(**XXX** *Discuss OBJ\_CR*)

We have reclassified all the objects with  $i - z > 2.2$  using the new cosmic-ray classifier. We found that depending on the seeing of a specific run, roughly 50%—70% of the i-dropout objects formerly classified as ”stars” are now classified as cosmic rays, while a negligible fraction of the ”real” objects (based on multiple observations and objects with confirmed spectra) are misclassified as cosmic rays. We then visually inspect the z images of the remaining candidates and reject a further 25% as cosmic rays and other artifacts.

So for 100 i-dropout objects classified as stars,

~ 60% are rejected by `MAYBE_CR`

another 10% are rejected by visual inspection of the z-band image

another 10% are rejected by further photometry

I.e. photo classifies 40% of i-band dropout objects as stars, of which about half are real.

These statements are true only for objects with  $S/N > 12$ . When  $S/N$  is worse than 10, then the number of artifacts increases dramatically: a weak CR and weak real objects are hard to separate in z band.

## 12. Measurements of Objects Shapes

### 12.1. ‘Stokes’ Parameters

(**XXX** *Write me*)

## 12.2. Adaptive Shape Measures

(XXX Write me. Most of it's in Bernstein et al. (2002), but Dave fixed the error analysis)

(XXX M\_rr\_cc)

## 12.3. Isophotal Measures

(XXX Write me)

## 13. Photo's Outputs and Performance

(XXX Write me?)

## 14. Testing and Quality Assurance (QA)

(XXX Write me?)

## 15. Future Directions

(XXX Crowded fields)

(XXX Working off coadded data)

(XXX Making photo applicable to non-SDSS data)

## 16. Conclusions

Blah Blah Blah

(XXX Availability of code)

The source code for all of the software described in this paper is freely available (XXX Describe how. The FNAL pserver cvs access is no more, I think). Unfortunately, in order to build *Photo* requires first building *Dervish* (see Sec. 1) and all of *it's* underpinnings, a task, which, whilst not impossible, is inconvenient.

Despite his appearance in the author list of this paper, RHL would especially like to thank

Jim Gunn for 25 years of advice and argument about image processing; many of the clever ideas embodied in *Photo* were his.

The use of linear prediction in the context of defect removal was suggested by Bill Press; thanks also to Jeremy Goodman for helpful discussions.

We’d like to thank Michael Strauss for helpful comments on an early version of this manuscript, and many of the members of the SDSS collaboration for repeatedly if politely asking when this paper would be finished; Jill Knapp and David Weinberg were especially effective.

Funding for the creation and distribution of the SDSS Archive has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Aeronautics and Space Administration, the National Science Foundation, the U.S. Department of Energy, the Japanese Monbukagakusho, and the Max Planck Society. The SDSS Web site is <http://www.sdss.org/>.

The SDSS is managed by the Astrophysical Research Consortium (ARC) for the Participating Institutions. The Participating Institutions are The University of Chicago, Fermilab, the Institute for Advanced Study, the Japan Participation Group, The Johns Hopkins University, the Korean Scientist Group, Los Alamos National Laboratory, the Max-Planck-Institute for Astronomy (MPIA), the Max-Planck-Institute for Astrophysics (MPA), New Mexico State University, University of Pittsburgh, University of Portsmouth, Princeton University, the United States Naval Observatory, and the University of Washington.

## A. Input Parameters

### A.1. Parameters Describing the Exposures, Camera, and Electronics

Four files are used to pass information about the detectors and camera into *Photo*:

The `opCamera` file which describes the layout of the CCDs within their dewars in the focal plane, and which provides a first estimate of the astrometric distortions. This file is used by *Astrom* and *PSP*, but not by frames.

The `opConfig` file which tells us which amplifiers each CCD used; the location of the extended register, overclock, and data regions associated with each amplifier.

The `opECalib` file characterizes the gain, noise properties, fullwell, and nonlinearity of the detectors.

The `opBC` file lists the location and type of the CCDs’ bad columns.

It will be seen that some of this information is traditionally included in the FITS headers. We decided not to follow this precedent as we didn’t wish to have to modify files containing raw data when our ideas about how to process it changed.

### A.2. Parameters which Control Data Processing

(XXX )

### A.3. Parameters which Control Algorithms used in Processing the Data

(XXX )

## B. Determining Bad Columns in the SDSS Camera

(XXX *Check if this is in a Janesick paper*)

One might hope that most classes of CCD defect would have almost no effect on TDI data; for example, a trap near the top of the chip only slightly depresses the effective quantum efficiency of the column. Unfortunately, most traps also have bad effects upon the CCD’s charge transfer efficiency (CTE), and manifest themselves as streaks extending up the column above bright stars.

We accordingly carried out a careful search for bad CTE columns using staring mode data, in which we deliberately clocked out more 20 rows of ‘data’ than there are physical rows in the detector. These extra 20 rows should be at the bias level, except that columns with poor CTE show an exponential trail into the overclock. It is a simple matter to look for such artifacts, and to mark the columns as bad in the `opBC` file (see Sec. A.1). We also identify strongly depressed and excessively noisy columns, although these defect types are usually also associated with poor CTE.

## C. Image Formation

A common, although not very good, representation of a PSF is a Gaussian

$$G(\alpha) \equiv \exp(-r^2/2\alpha^2) \tag{C1}$$

n.b.  $G(0) = 1$  rather than  $1/(2\pi\alpha^2)$ ; we also define  $G'(\alpha) \equiv G(0)/(2\pi\alpha^2)$ . We shall use the convention in which  $G$ ’s Fourier transform is  $\exp(-(2\pi k\alpha)^2/2)$  (Press et al. 1992).

For most ground-based telescopes the PSF is dominated by turbulence in the atmosphere, producing images whose Fourier transform is given by (Fried 1966):

$$\exp\left(-6.8839(2\pi k\lambda/r_0)^{5/3}/2\right) \tag{C2}$$

where

$$\lambda/r_0 \equiv \text{FWHM}/0.976 \tag{C3}$$

(The 6.8839 comes from the definition of the Fried parameter  $r_0$ ; the 0.976 comes from a numerical inverse Fourier transform; FWHM is the full width at half maximum [ $2\sqrt{2\ln 2}\alpha$  for a Gaussian]). This is not especially convenient, but fortunately this form of the PSF is well described by the sum of two Moffat functions (Racine 1996). A sum of two Gaussians,  $G(\alpha) + 0.1G(2\alpha)$ , is also a convenient and reasonably accurate representation of Eq. C2

This continuous image is then sampled by the detector’s pixels, i.e. the image is convolved with the pixel response function and multiplied by an array of  $\delta$ -functions centred in each pixel. For an ideal CCD with square pixels whose sensitivity is either 0 or 1 this corresponds to a convolution with a rectangular top-hat function and sampling on a rectangular grid. Providing that the response function of each pixel is the same, and that the image is properly sampled, the details of the sensitivity variations within the CCD’s pixels are unimportant.

This PSF-and-pixel convolved image is what *Photo* attempts to measure as accurately as possible. We never attempt to deconvolve the pixel response although this is theoretically possible; the attenuation due to pixellation at the band limit is only about 64% in 1 arcsec (Gaussian) seeing.

The structure of the PSF is dominated by turbulence in the atmosphere and telescope dome, and varies with time; if the telescope’s optics are imperfect they can produce an additional spatial variation along with further colour terms. We do not try to correct the images for these effects, although we do model them in an attempt to remove their signature from our measured object parameters.

#### D. The Calculation of the Centroid of Properly Sampled Data

We can use an approach similar to that of Eq. 21 to find the centroid of properly sampled, pixellated, data. As before, we can use Eq. 20 to write:

$$\begin{aligned}
 \langle x \rangle &= \int_{-\infty}^{\infty} x f(x, y) dx \\
 &= \int_{-\infty}^{\infty} x \sum_i f_i \frac{\sin((x-i)\pi)}{(x-i)\pi} dx \\
 &= \sum_i i f_i + \frac{1}{\pi} \sum_i f_i \int_{-\infty}^{\infty} \sin((x-i)\pi) dx \\
 &= \sum_i i f_i
 \end{aligned} \tag{D1}$$

where the second term vanishes as it’s an *odd* function integrated over an *even* interval, admittedly one whose limits have been permitted to become infinite.

### E. Übercalibration of SDSS Data

(**XXX** *Write or delete me*)

(**XXX** *MAS says:*) Here, I could imagine a qualitative discussion on the difficulties of \*really\* good photometry, which would touch upon many of the things you mention in the main text:

Scattered light and variable flat fields;

Changing filters;

Getting PSF determination right

### F. Some Statistical Properties of Poisson Distributions

(**XXX** *Expand*)

Poisson distributions have some surprising properties, in particular

$$\text{mean} = \text{median} + 1/6$$

When clipping at  $2.326\sigma$ ,

$$\text{median}_{\text{true}} = \text{median}_{\text{clipped}} + 0.060$$

When clipping at  $4.0\sigma$

$$\text{mean}_{\text{true}} = \text{mean}_{\text{clipped}} + (\mathbf{XXX}????)$$

### G. Reconstructing the KL-PSF given a psField file

The PSF information is written to files known as `psField` files; these files are exported in a slightly modified form as `tsField` files.

There is stand-alone code available at <http://www.astro.princeton.edu/~rhl/readAtlasImages> which may be used to reconstruct the PSF at a desired point in the frame.

After downloading and building the utilities, to reconstruct the  $z$  PSF (i.e. the 5th HDU) at the position (row, col) = (500, 600) from run 1336, column 2, field 51 you'd say:

```
read_PSF psField-001336-2-0051.fit 5 500.0 600.0 foo.fit
```

The desired PSF would appear as an unsigned short FITS file in `foo.fits`; the background level is set to the standard ‘soft bias’ of 1000. If you want a floating image, change a line in the `read_PSF.c`; look for `/* create a float region */`

I don’t expect that many users will actually want to use the `read_PSF` executable (although it is perfectly functional). The main use of the product will probably be to link into custom built executables that need to process PSF data. This has indeed already happened; for example, IDL wrappers are available at (~~XXX~~ *Where?*).

### G.1. Interpretation of PSP status codes

Each band in each field has an associated status (`status`), and a summary status (`psp_status`) is provided for each field. These status values are defined by an enumerated type, `PSP_STATUS` (see Table 9), and consist of a small integer (`status & STATUS_MASK`) and a set of bitflags (`status & ~STATUS_MASK`). These status values are written into HDU 6 of the `psField` file.

The `psp_status` is the largest of the per-band (`status & STATUS_MASK`) values.

For example, 65 means (PSF22: SPARSE) and 96 means (OK: EXTENDED\_KL SPARSE).

Name	Value	Description
UNKNOWN	-1	This should never happen
OK	0	Everything seems OK
PSF22	1	We were forced to take the PSF to vary linearly across the field
PSF11	2	We were forced to take the PSF to be a constant across field
NOPSF	3	We forced forced to take a default PSF
ABORTED	4	We aborted processing for this field
MISSING	5	This field was a dummy, used to replace missing data
OE_TRANSIENT	6	This field includes odd/even bias level transient
STATUS_MASK	31	Mask defining which bits are used for status values; higher bits are available to be set with extra information
EXTENDED_KL	0x20	The Window for stars used to determine the PSF’s spatial structure was extended
SPARSE	0x40	The field is only sparsely populated with PSF stars

Name	Value	Description
Table 9:: <i>PSP</i> 's status values, describing how well the PSF is determined for a field.		

## REFERENCES

- Adelman-McCarthy, J. et al. 2005 Submitted to ApJSupp.
- Bernstein, G. M. and Jarvis, M. 2002 AJ 123 583
- C. de Boor, 1978, *A Practical Guide to Splines*, (N.Y. : Springer-Verlag).
- Bracewell, R. N., 2000, *The Fourier transform and its applications*, (McGraw Hill)
- Brent, R.P., 1973, *Algorithms for Minimization without Derivatives* (Englewood Cliffs NJ; Prentice-Hall), Chapter 5.
- Bresenham, J.E., 1965, *Algorithm for computer control of digital plotter*. IBM Syst. J. 4 , 25
- Doi, M. (**XXX** *The calibration system*)
- Fan, X. et al. 2001 AJ 122, 2833
- Forsythe, G.E., Malcolm M.A., and Moler, C.B., 1977 *Computer Methods for Mathematical Computations* (Englewood Cliffs NJ; Prentice-Hall), §8.2.
- Fried, D. L. 1966 JOSA 56, 1372
- Fukugita, M., Ichikawa, T., Gunn, J.E., Doi, M., Shimasaku, K., & Schneider, D.P. 1996, AJ, 111, 1748
- Ivezić, Ž. et al. 2004, *Astronomische Nachrichten*, 325, No. 6-8, 583-589 (also astro-ph/0410195)
- Gunn, J. E., et al., 1998 AJ 116, 3040
- Gunn, J. E., et al., 2005 in preparation
- Hogg, D.W., Finkbeiner, D.P., Schlegel, D.J., & Gunn, J.E. 2001, AJ, 122, 2129
- Hotelling H., 1933, J. Educ. Psychol., 24, 417&498.
- Karhunen, K. 1947, Ann. Acad. Sci. Fennicae, Ser. A137 (Translated by I. Selin in “On Linear Methods in Probability Theory”, T-131, 1960, The RAND corp., Santa Monica, Ca)
- Loève, M., 1948, “Fonctions Alineare de Secondre Ordre.” in P. Levy, *Processus Stochastique et Mouvement Brownien*, Hermann, Paris.

- Lupton, R.H. (1993) *Statistics in Theory and Practice, Chapter 10* Princeton University Press
- Lupton, R. H., Gunn, J. E. & Szalay, A. 1999, AJ, 118, 1406
- Lupton, R. H., Gunn, J.E., Ivezić, Ž., Knapp, G.R., Kent, S., & Yasuda, N. 2001, in *Astronomical Data Analysis Software and Systems X*, edited by F. R. Harnden Jr., F. A. Primini, and H. E. Payne, ASP Conference Proceedings, 238, 269
- Lupton, R. H.. 2005, Submitted to AJ
- Marquardt, D.W., 1963, Journal of the Society for Industrial and Applied Mathematics, 11, 431.
- (XXX *Is even the year correct?*)
- (XXX *Is even the year correct?*)
- Jeffrey R. Pier, 1998, Private Communication.
- Jeffrey R. Pier et al., 2003, AJ 125 1559.
- William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery. Numerical Recipes: the Art of Scientific Computing. 1992, Cambridge University Press.
- William H. Press and George B. Rybicki, 1993, *Large-Scale Linear Methods for Interpolation, Realization, and Reconstruction of Noisy Irregularly Sampled Data*, in *Time Series Prediction: Forecasting the Future and Understanding the past* Eds. A.S. Weigend and N.A. Gershenfeld, SFI Studies in the Sciences of Complexity, Proc. Vol. XV, Addison-Wesley, 1993 (pp 493-512).
- Racine, R., 1996 PASP, 108, 699
- Schlegel, D. J. et al. (XXX *Private Communication*)
- Sérsic, J.L. 1968, *Atlas de Galaxias Australes, Observatorio Astronomico, Cordoba.*
- Smith, J.A. et al. 2002, AJ, 123, 2121
- Stoughton, C. et al. AJ 2002 123 485
- Strauss, M.A., et al. 2002 AJ, 124, 1810-1824
- (XXX *Does this exist?*)
- Szokoly, G.P., 1999, *Deep multicolor surveys of the galaxy population*, PhD thesis (Johns Hopkins University)
- Tucker, D., et al. Submitted to AJ
- Wells, D. (XXX *The author may be wrong*) 2001 A&A 376 359

Yanny, B. (**XXX** *What's the reference? Is 1995 correct?*)

York et al., 2000 AJ 120 1579

Zacharias, N., et al. 2000, AJ, 120, 2131

Aperture	Radius/pixels	Radius/asec	Area/pixels	Area/asec <sup>2</sup>
1	0.56	0.23	1	0.16
2	1.69	0.68	8	1.28
3	2.59	1.03	12	1.92
4	4.41	1.76	40	6.40
5	7.51	3.00	116	18.56
6	11.58	4.63	244	39.04
7	18.58	7.43	664	106.24
8	28.55	11.42	1476	236.16
9	45.50	18.20	3944	631.04
10	70.51	28.20	9114	1458.24
11	110.53	44.21	22762	3641.92
12	172.49	69.00	55094	8815.04
13	269.52	107.81	134732	21557.12
14	420.51	168.20	327318	52370.88
15	652.50	261.00	782028	125124.47

Table 7: Outer radii and areas of *Photo*'s annuli, measured in pixels and arcseconds. Note that the areas in pixels are exact integers, which means that the first radius is  $r_1 \equiv 1/\sqrt{\pi} \approx 0.564190$

Quantity ( $P$ )	$m$	$u$	$g$	$r$	$i$	$z$
$R_P$	18	$\sim 1.5$	0.77	0.83	0.96	0.94
	19	1.09	0.87	0.92	1.00	0.95
	21	0.91	0.96	0.91	0.90	0.73
$R_{50}$	18	$\sim 1.5$	1.90	1.96	1.67	1.42
	19	1.74	1.84	1.82	1.54	1.16
	21	1.19	1.37	1.27	1.15	0.85
$R_{90}$	18	$\sim 1.7$	1.19	1.15	1.13	0.92
	19	0.95	1.10	1.05	1.00	0.92
	21	0.92	0.90	0.80	0.88	0.89
$R_{90}/R_{50a}$	18	0.80	0.65	0.62	0.65	0.53
	19	0.60	0.66	0.60	0.62	0.51
	21	0.55	0.57	0.50	0.52	0.48
$R_{90}/R_{50b}$	18	$\sim 1.3$	1.25	1.09	1.17	0.90
	19	1.03	1.19	1.01	1.01	0.87
	21	1.30	1.25	1.09	1.17	0.90
$F_P$	18	—	2.03	1.79	1.60	1.72
	19	2.12	1.60	1.55	1.35	1.32
	21	1.37	1.30	1.21	1.13	0.94

Table 8: The ratio of the empirical estimates of errors in Petrosian quantities  $P$  to *Photo*'s estimate of its standard deviation,  $\delta P$ . The value quoted is  $(P_1 - P_2)/\sqrt{\delta P_1^2 + \delta P_2^2}$ , and the subscripts 1 and 2 refer to the two runs used in the comparison, 745 and 756. The two sets of values for the concentration parameter,  $R_{90}/R_{50}$ , are calculated on the assumption that  $\text{covar}(R_{90}/R_{50}) = \sigma_{R_{50}}\sigma_{R_{90}}$  (case a) and that  $\text{covar}(R_{90}/R_{50}) = \sigma_{R_{50}}\sigma_{R_{90}}$  (case b).