- SDSS photometry
- Images:
  - *rerun/run/camcol/field/filter*
  - *units and calibration*
  - *noise properties*
  - *point spread function*
  - *WCS headers*
  - *quality flags*
- Catalogs
  - *parents & children*
  - *magnitudes*
  - *star/galaxy separation*
  - *quality flags*

http://sdss3.org/dr9/imaging
http://sdss3.org/dr9/algorithms

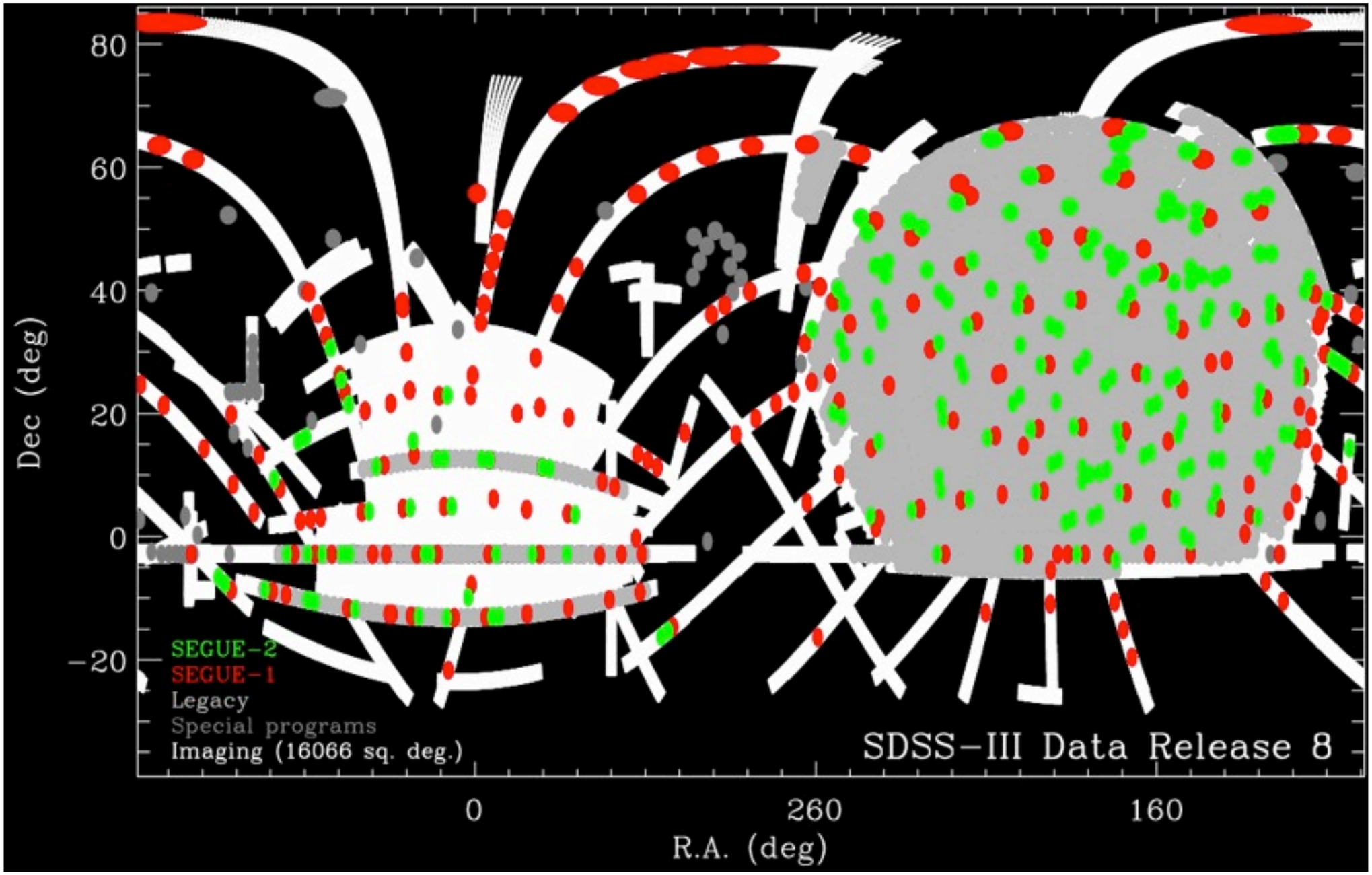
- SDSS data releases
  - *DR1 through DR9 (beginning in ~ 2001)*
  - *roughly yearly, cumulative releases*
- Science Archive Server
  - http://data.sdss3.org
  - http://data.sdss3.org/sas/dr9
  - http://mirror.sdss3.org

spectra
images
raw data
"flat files"

- Catalog Archive Server
  - http://skyserver.sdss3.org/casjobs
  - http://skyserver.sdss3.org/

catalog database
visual tools

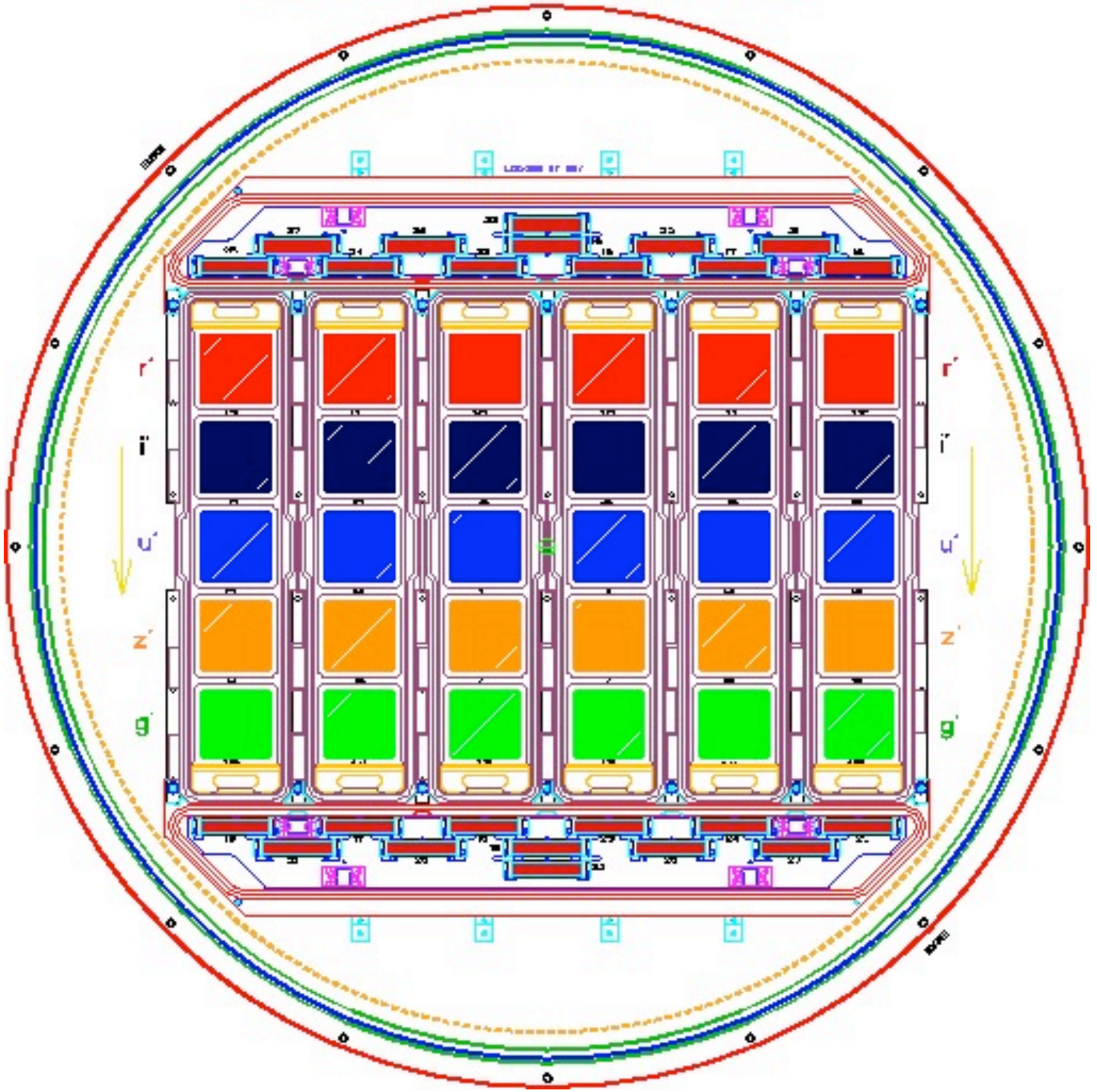
- The SDSS-III API
  - http://api.sdss3.org



http://data.sdss3.org/sas/dr9/boss/photoObj/photoRunAll-dr9.fits

```
runs= pyfits.open('photoRunAll-dr9.fits')
print runs[1].data.columns
```

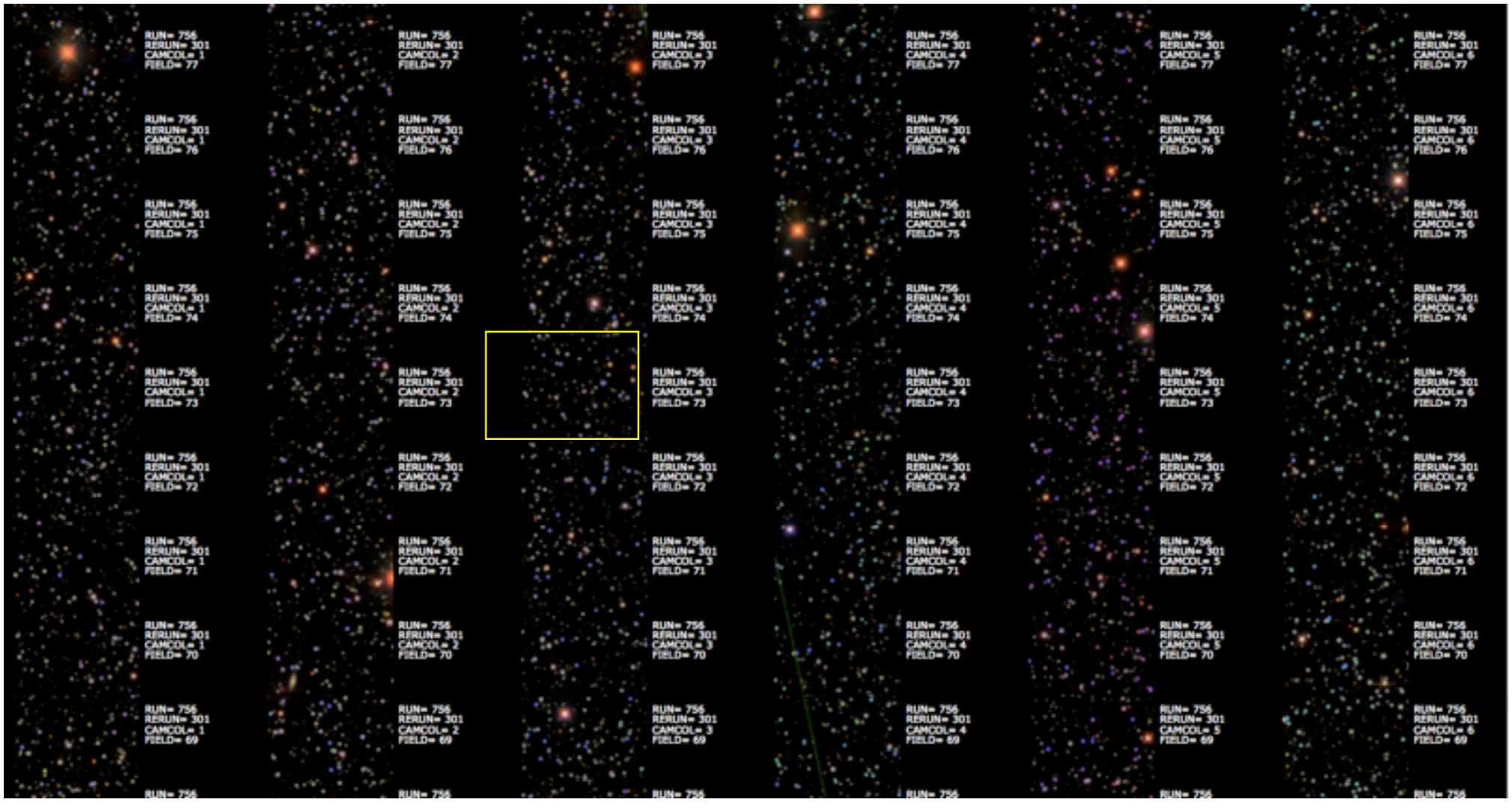
SDSS CAMERA



Imaging drift scan layout

e.g. run 756

http://data.sdss3.org/sas/dr9/boss/photoObj/frames/301/3647/frames-run-003647.html

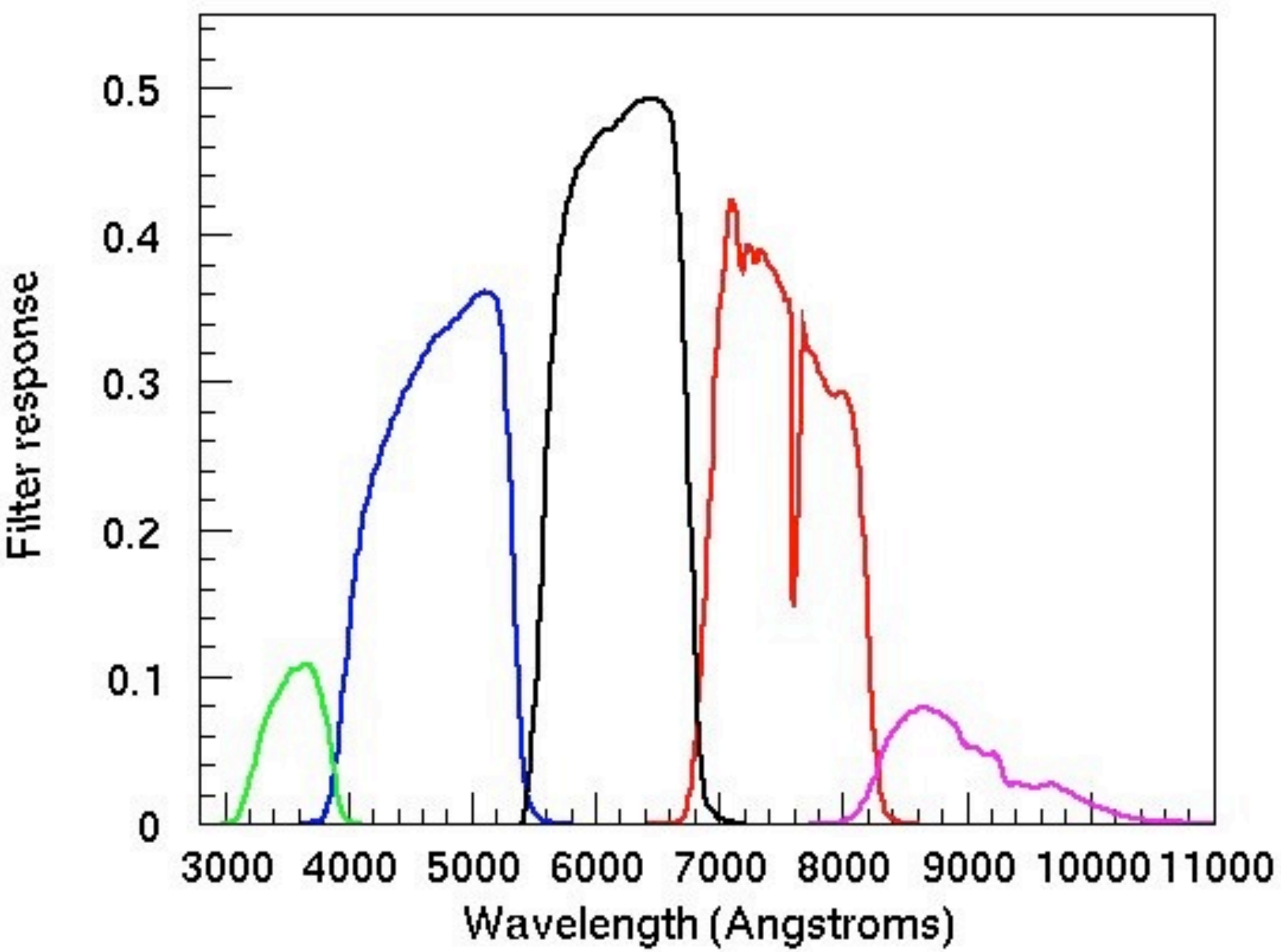


field

1 2 3 4 5 6

camcol

Imaging bandpasses



Imaging data layout

Imaging data layout

rowc (0-1488)

colc (0-2047)

- SkyServer stitches into JPGs that you can navigate:
  - http://skyserver.sdss3.org/public/en/tools/chart/navi.asp
- Science Archive Server can make FITS mosaics
  - http://data.sdss3.org/mosaics

Imaging data layout

JPG file

http://data.sdss3.org/sas/dr9/boss/photoObj/frames/301/3647/3/frame-irg-003647-3-0061.jpg

Reduced, calibrated FITS file

http://data.sdss3.org/sas/dr9/boss/photoObj/frames/301/3647/3/frame-g-003647-3-0061.fits.bz2

http://data.sdss3.org/datamodel/files/BOSS_PHOTOOBJ/frames/RERUN/RUN/CAMCOL/frame.html

Raw data from telescope

http://data.sdss3.org/sas/dr9/boss/photo/data/3647/fields/3/idR-003647-g3-0061.fit.Z

Look at these with ds9!

Flux units

The "fluxes" are in units of AB nanomaggies. That is:

$$\mathrm{flux} = 10^9 \times \frac{f_\nu}{3631\ \mathrm{Jy}}$$

flux of Vega at 5500 Angstroms

$$f_\lambda \rightarrow \mathrm{erg\ cm^{-2}\ s^{-1}\ \AA^{-1}}$$

$$f_\nu \rightarrow \mathrm{erg\ cm^{-2}\ s^{-1}\ Hz^{-1}}$$

$$f_\nu \mathrm{d}\nu = f_\lambda \mathrm{d}\lambda$$

$$f_\nu = f_\lambda \left|\frac{\mathrm{d}\lambda}{\mathrm{d}\nu}\right| = f_\lambda \frac{c}{\lambda^2}$$

A Jansky (Jy) is a special unit:

$$\mathrm{Jy} = 10^{-23}\mathrm{erg\ cm^{-2}\ s^{-1}\ Hz^{-1}}$$

Nanomaggies are related to magnitudes as follows:

$$\mathrm{mag} = 22.5 - 2.5 \log_{10} \mathrm{flux}$$

- Charge-Coupled Devices images convert photons to electrons on 1-to-1 basis
- Images are usually reported in ADU (Analog-to-Digital Unit), sometimes called DN (Data Number), related to electron counts by the detector "gain"
- Need to calibrate DN to flux units on the basis of the observations.

$$\mathrm{DN} = \frac{N}{\mathrm{Gain}}$$

- CCD detectors have a "dark current" that needs to be *subtracted*
- Across the field and from pixel-to-pixel telescope throughput and detector quantum efficiency: need to *divide* by the *flat field*
- Atmospheric transparency needs to be determined, by measuring fluxes of "standards"
- Angle through atmosphere accounted for ("airmass," or secant of zenith angle)
- SDSS's imaging strategy allows relative transparency and flat field determination from multiple observations of the same stars

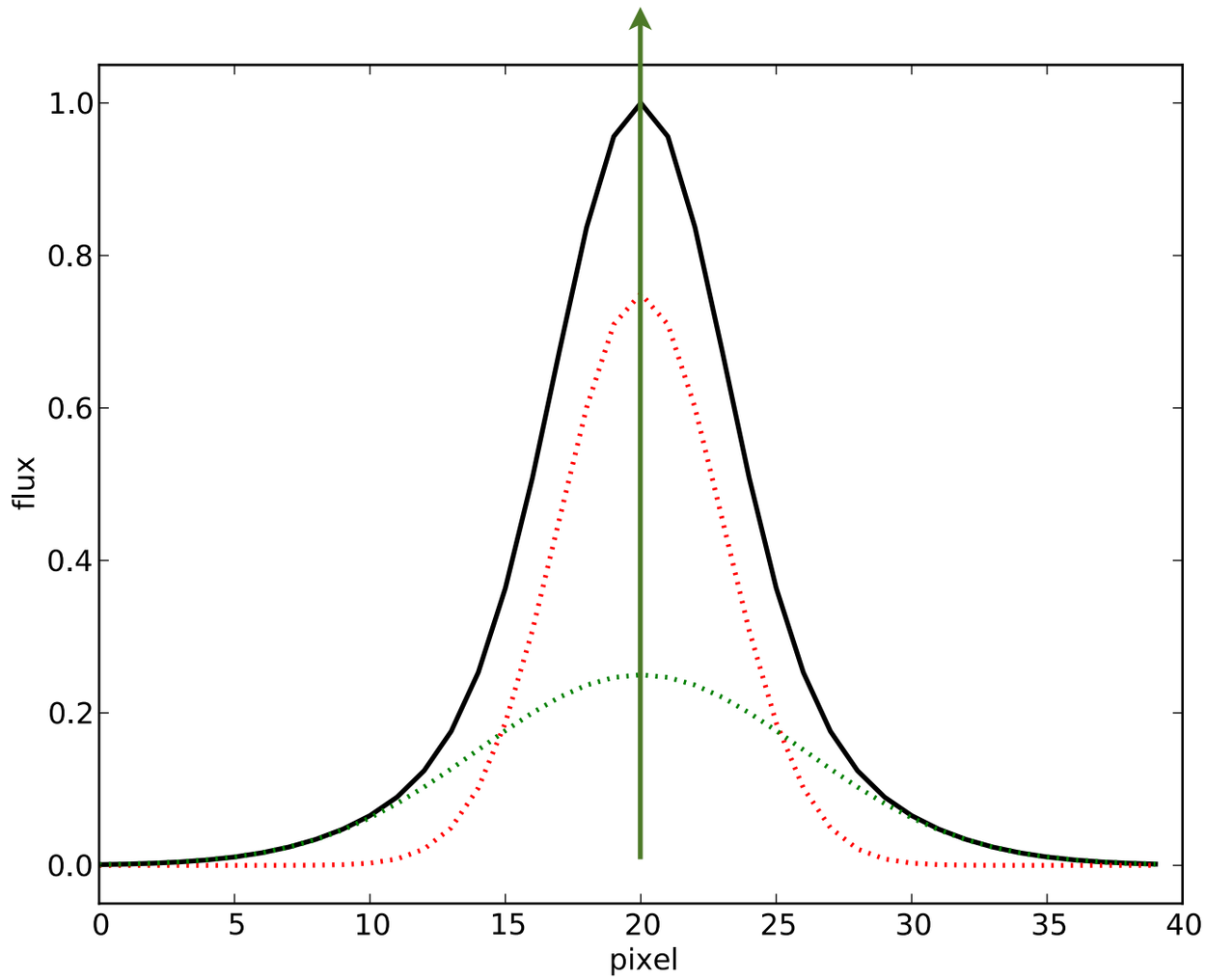
where this includes sky emission from atmosphere, which needs to be subtracted even from a calibrated image

$$\mathrm{DN} = \left(f_\nu \times (\mathrm{trans.})^{1/\cos z} \times \mathrm{flat} \right) + \mathrm{dark}$$

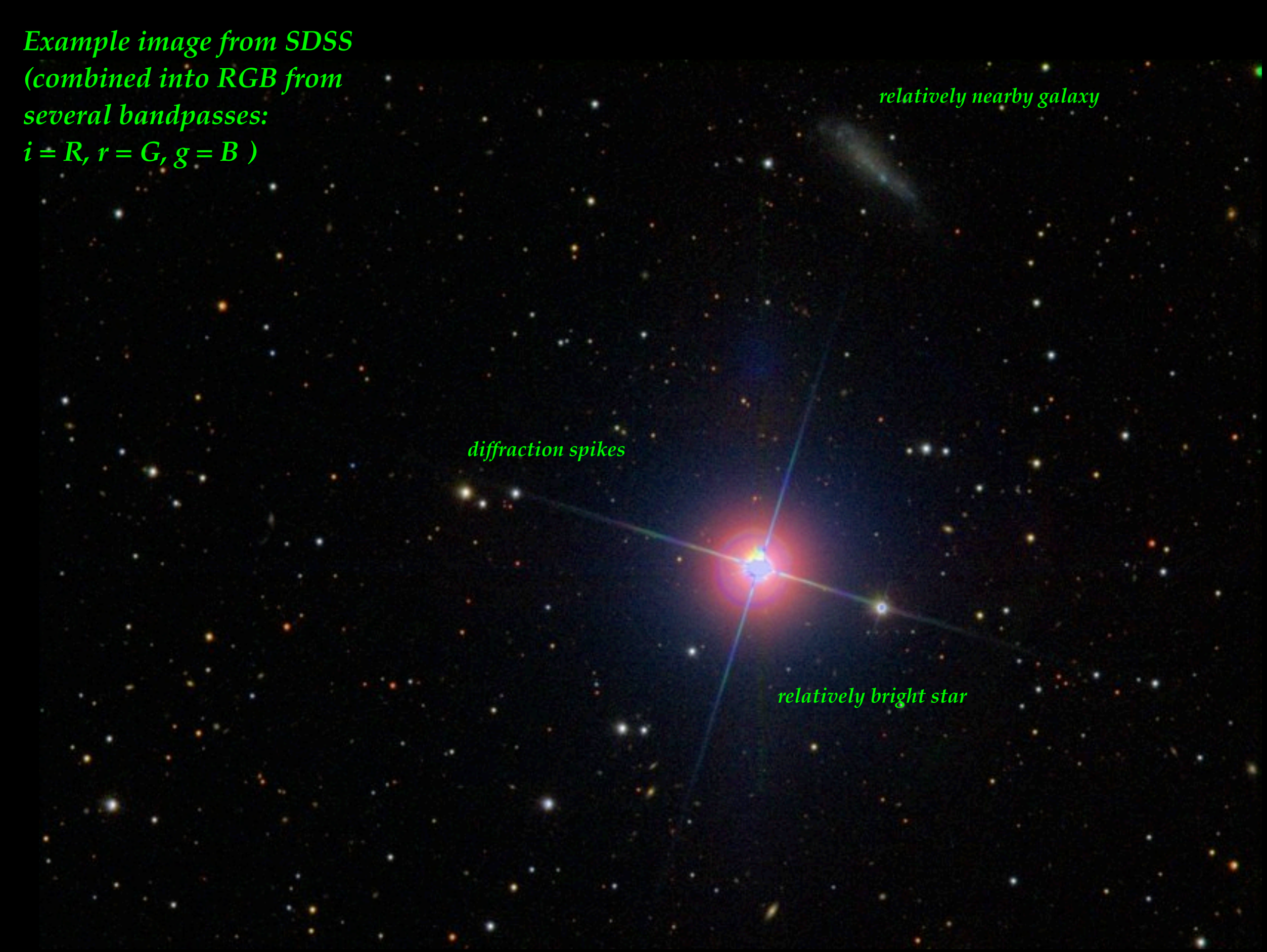
- Charge-Coupled Devices images convert photons to electrons on 1-to-1 basis
- What causes noise in CCD images?
  - *finite number of photoelectrons (Poisson statistics)*
  - *noise in "dark current" (again Poisson noise)*
  - *"read noise" in the amplifiers*

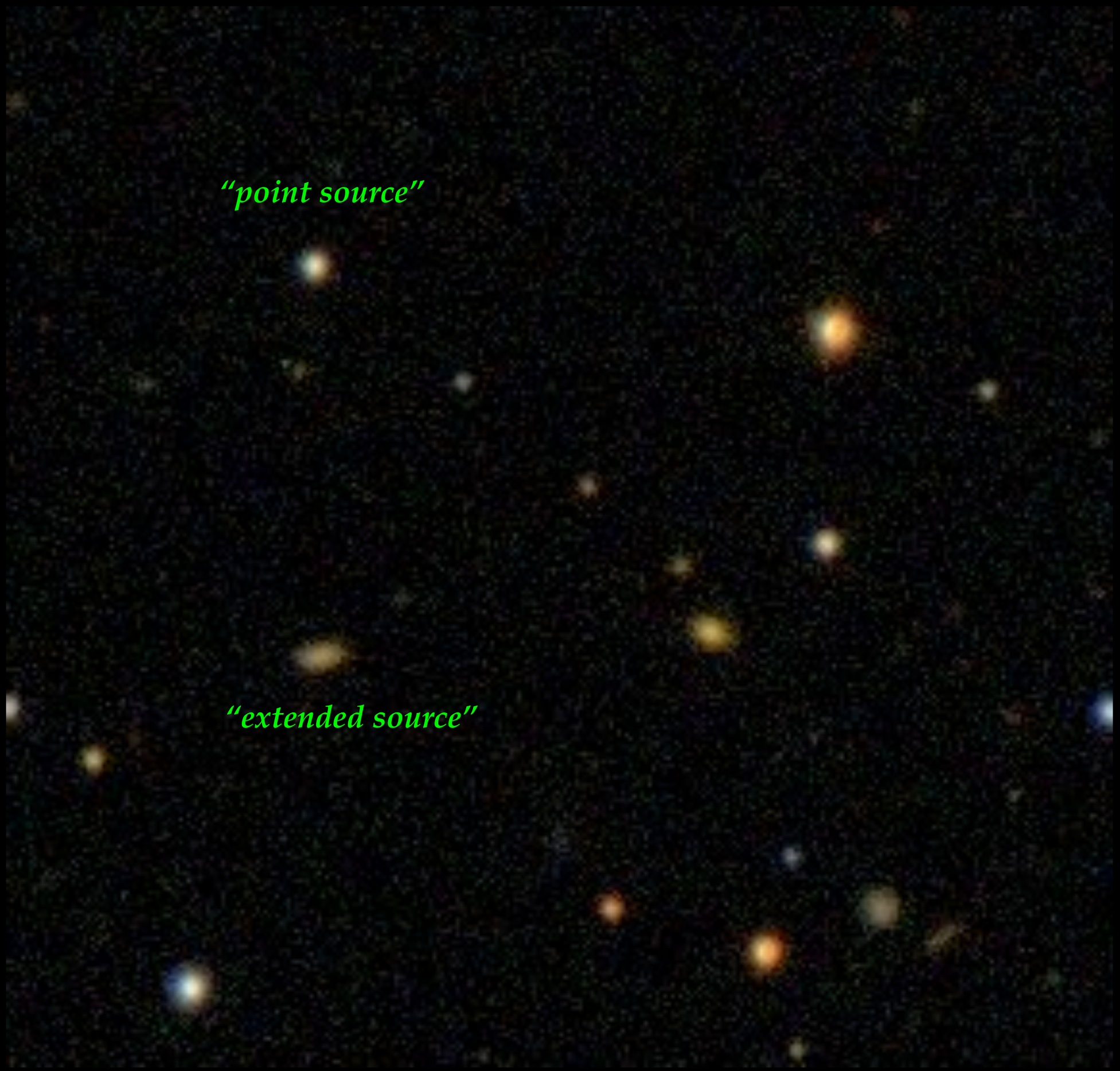
$$\sigma_N^2 = N_{\rm sky} + N_{\rm dark} + R^2$$

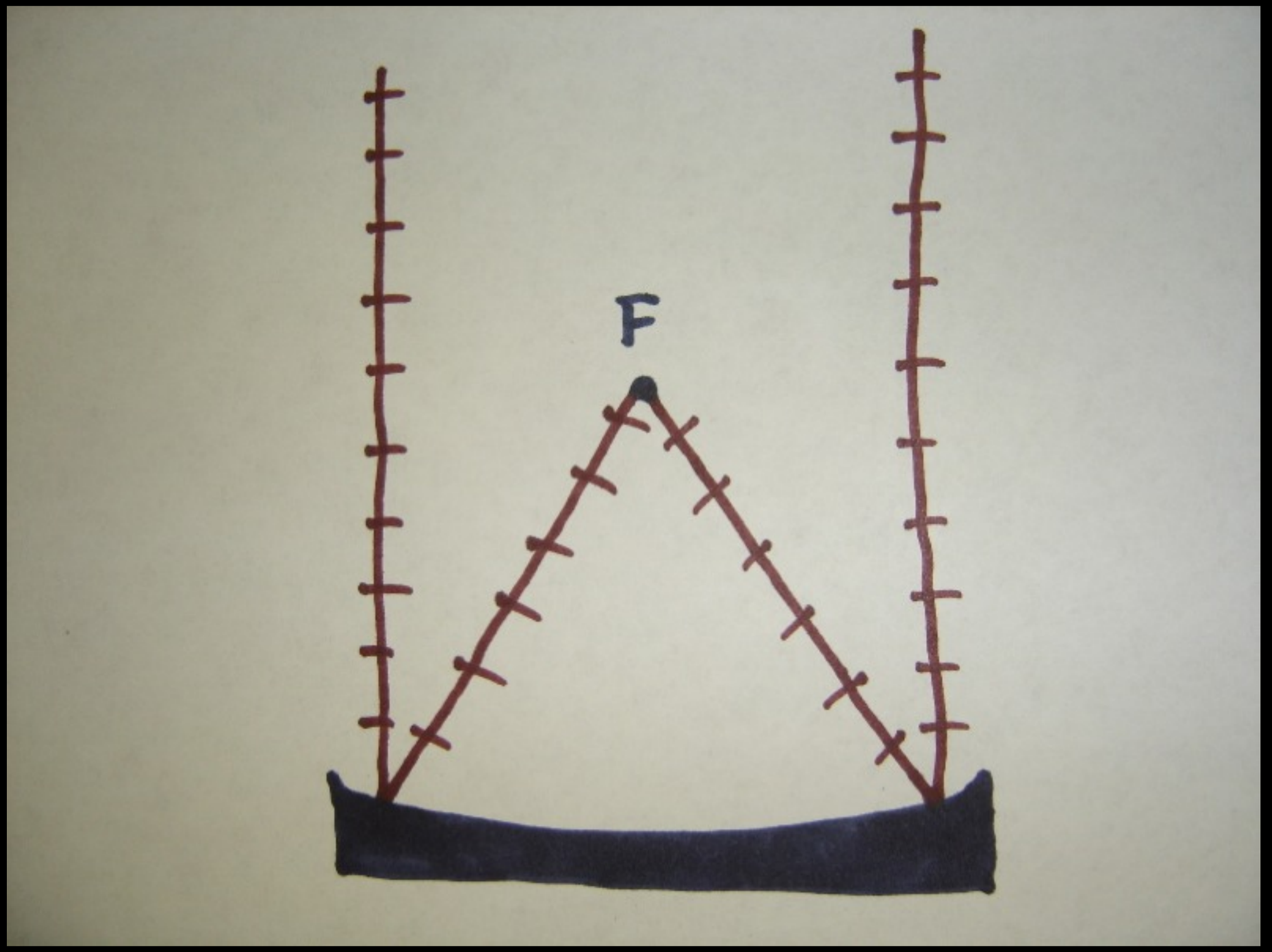
- Point spread function: response of system to point source (delta function)
- In ground-based astronomy, cores well modeled with a double Gaussian
- There are features further out (diffraction spikes, power-law wings)

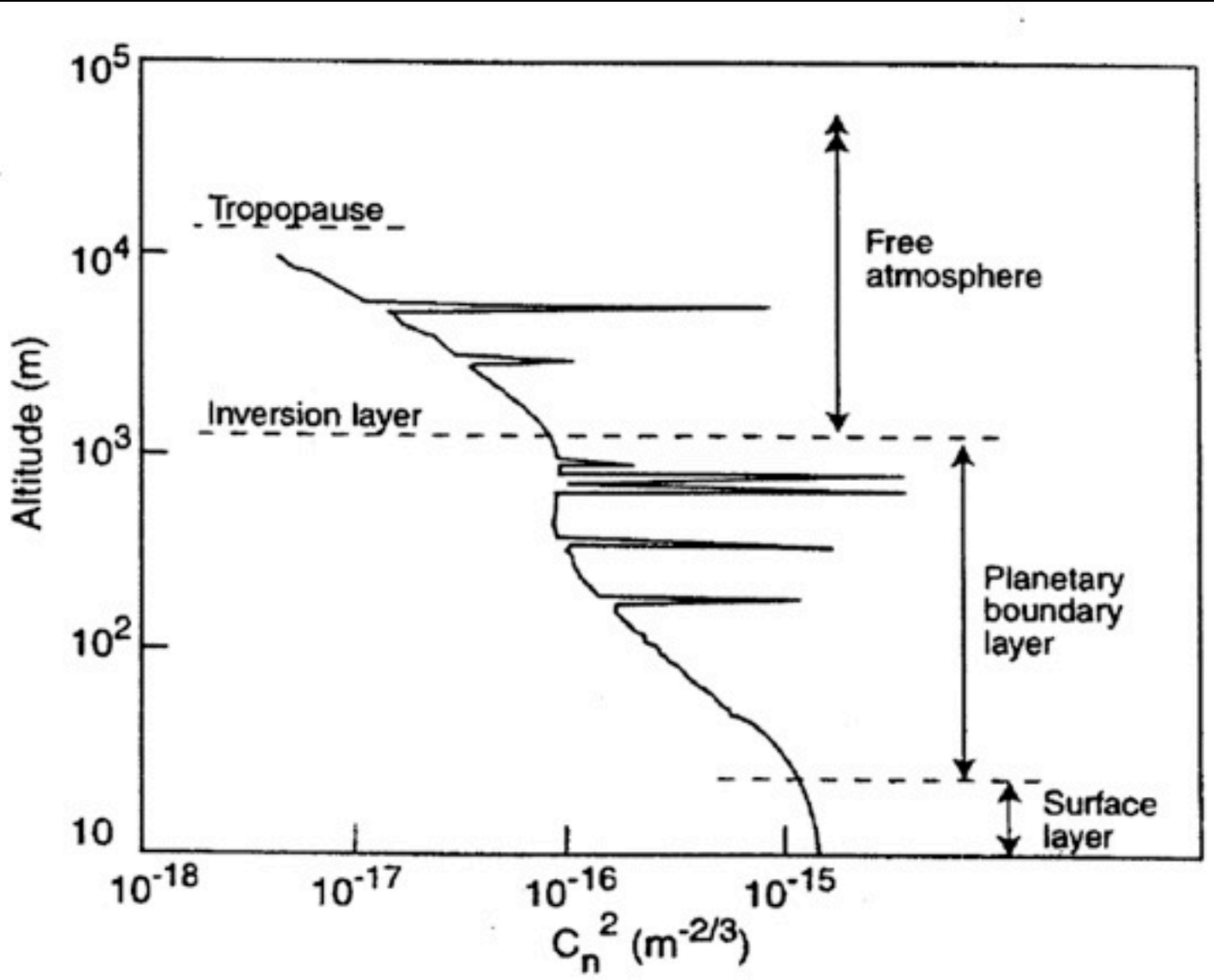


Example image from SDSS (combined into RGB from several bandpasses: i = R, r = G, g = B)




"point source"
"extended source"


F



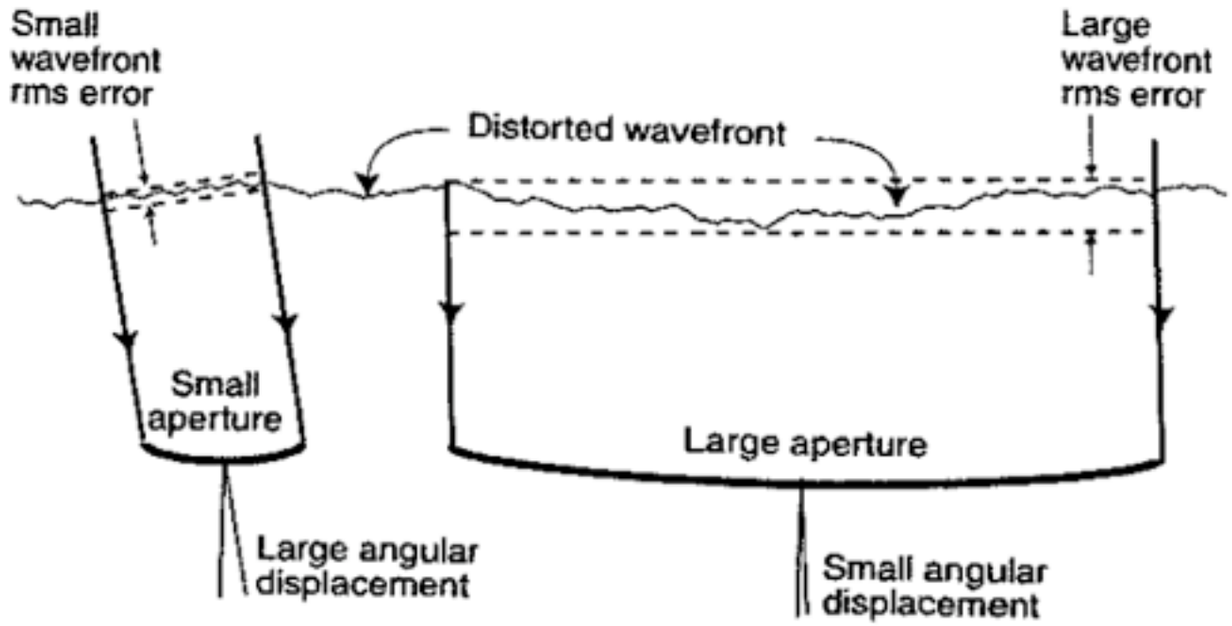


Fig. 1.9. Image motion decreases as telescope size increases, whereas the reverse is true for image blur. (Adapted from Ref. [8].)

- World Coordinate System (WCS) headers for astrometry
- Allow conversion between (x,y) and (ra,dec) for an image, dealing with:
  - *projection of sphere onto plane*
  - *optical distortions*
- SDSS has its own astrometry header format too (“asTrans”)

```
im= pyfits.open('frame-r-004797-4-0189.fits')
print im[0].header
```

```
RADECSYS= 'ICRS    '           / International Celestial Ref. System
CTYPE1  = 'RA---TAN'           /Coordinate type
CTYPE2  = 'DEC--TAN'           /Coordinate type
CUNIT1  = 'deg     '           /Units
CUNIT2  = 'deg     '           /Units
CRPIX1  =       1025.00000000 /X of reference pixel
CRPIX2  =       745.000000000 /Y of reference pixel
CRVAL1  =        334.167305597 /RA of reference pixel (deg)
CRVAL2  =       0.314301833490 /Dec of reference pixel (deg)
CD1_1   =    5.14450142856E-08 /RA deg per column pixel
CD1_2   =    0.000109986653735 /RA deg per row pixel
CD2_1   =    0.000110011043479 /Dec deg per column pixel
CD2_2   =   -7.93590273551E-09 /Dec deg per row pixel
```

```
from astropy import wcs
pixcrd = numpy.array([[1025.,745.], [0., 745.]], numpy.float_)
world = w.wcs_pix2world(pixcrd, 1)
print world
```

Catalog directory and file structure

```
http://data.sdss3.org/sas/dr9/boss/photoObj/[rerun]/[run]/
      photoField-[run]-[camcol].fits
      [camcol]/
         photoObj-[run]-[camcol]-[field].fits
```

photoField files contain info about each field

photoObj files contain each object in each field

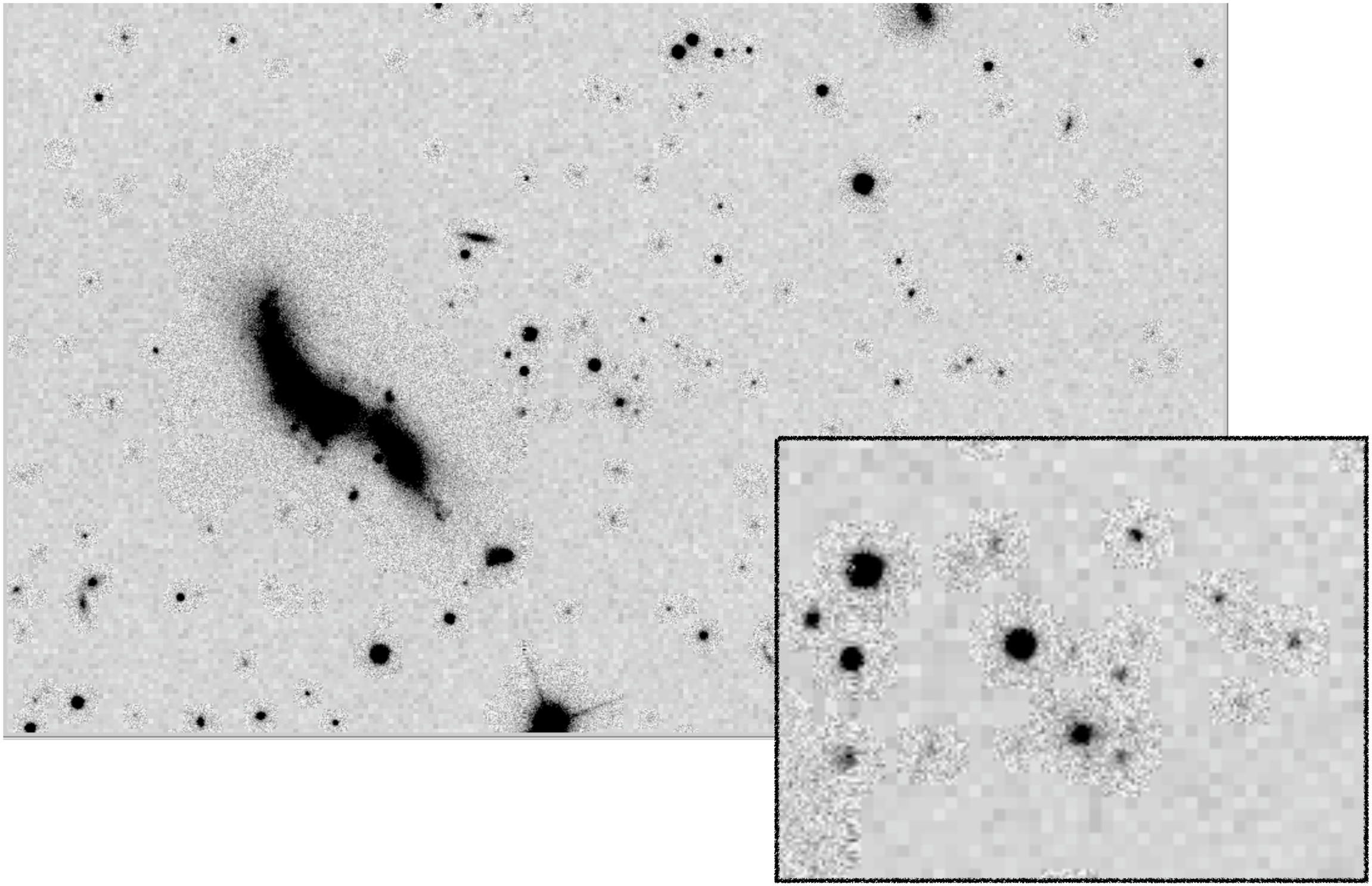
Note: all these catalogs total 3 Tbytes. There is an easier way to get at the information than downloading them all. Stay tuned.

http://sdss3.org/dr9/algorithms/image_quality.php

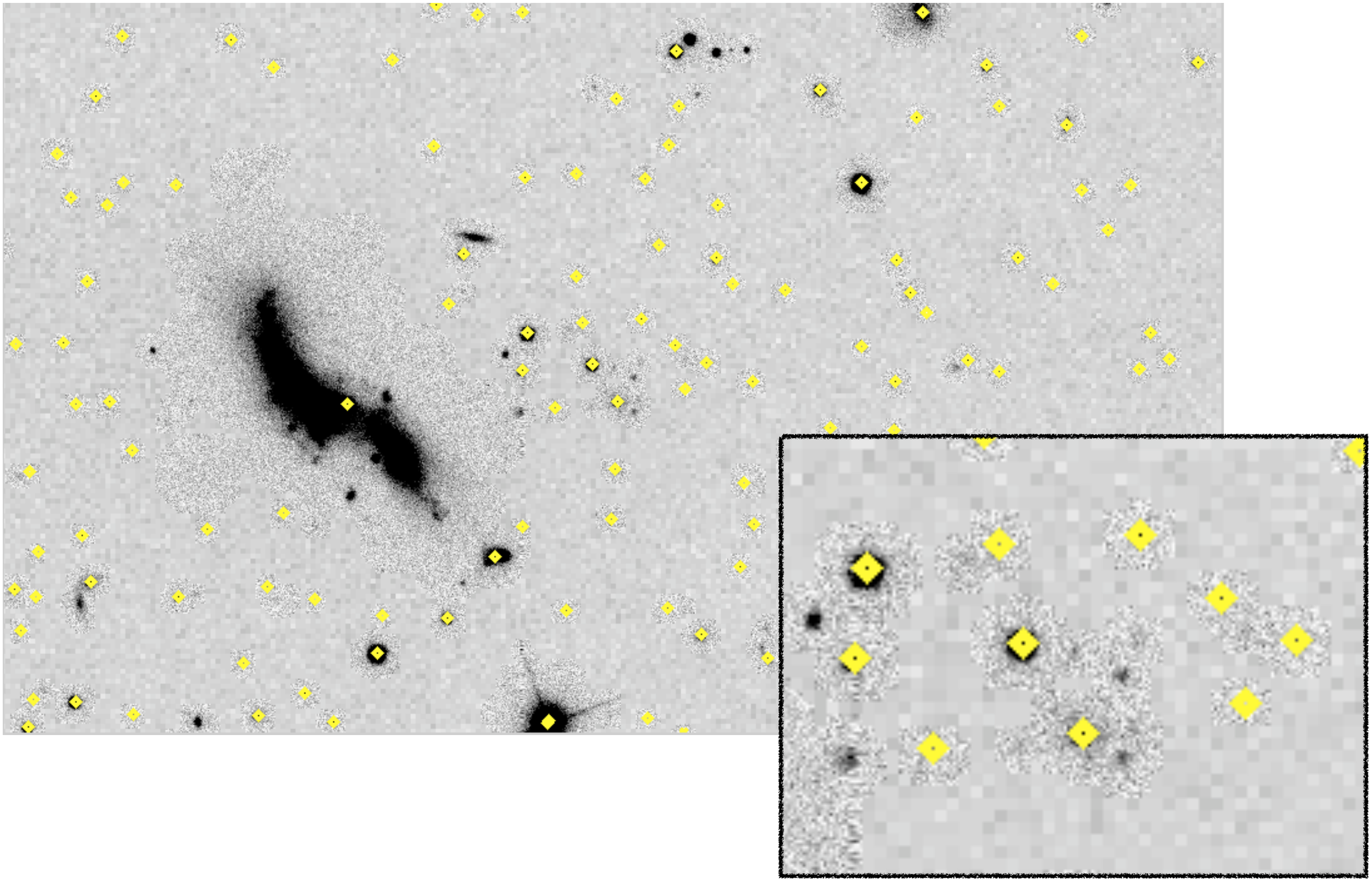
- Quality flags for images
  - *hiccup in observing?* `image_status`
  - *well-calibrated?* `calib_status`
  - *trouble fitting PSF?* `psp_status`
  - *photometric analysis failed?* `photo_status`
- Other important indicators of quality:
  - *point spread function FWHM* `psfwidth`
  - *sky brightness* `skyflux`

- Overall score for each field (0.5 to 1.0 "good", and 0.0 to 0.5 "bad") `score`

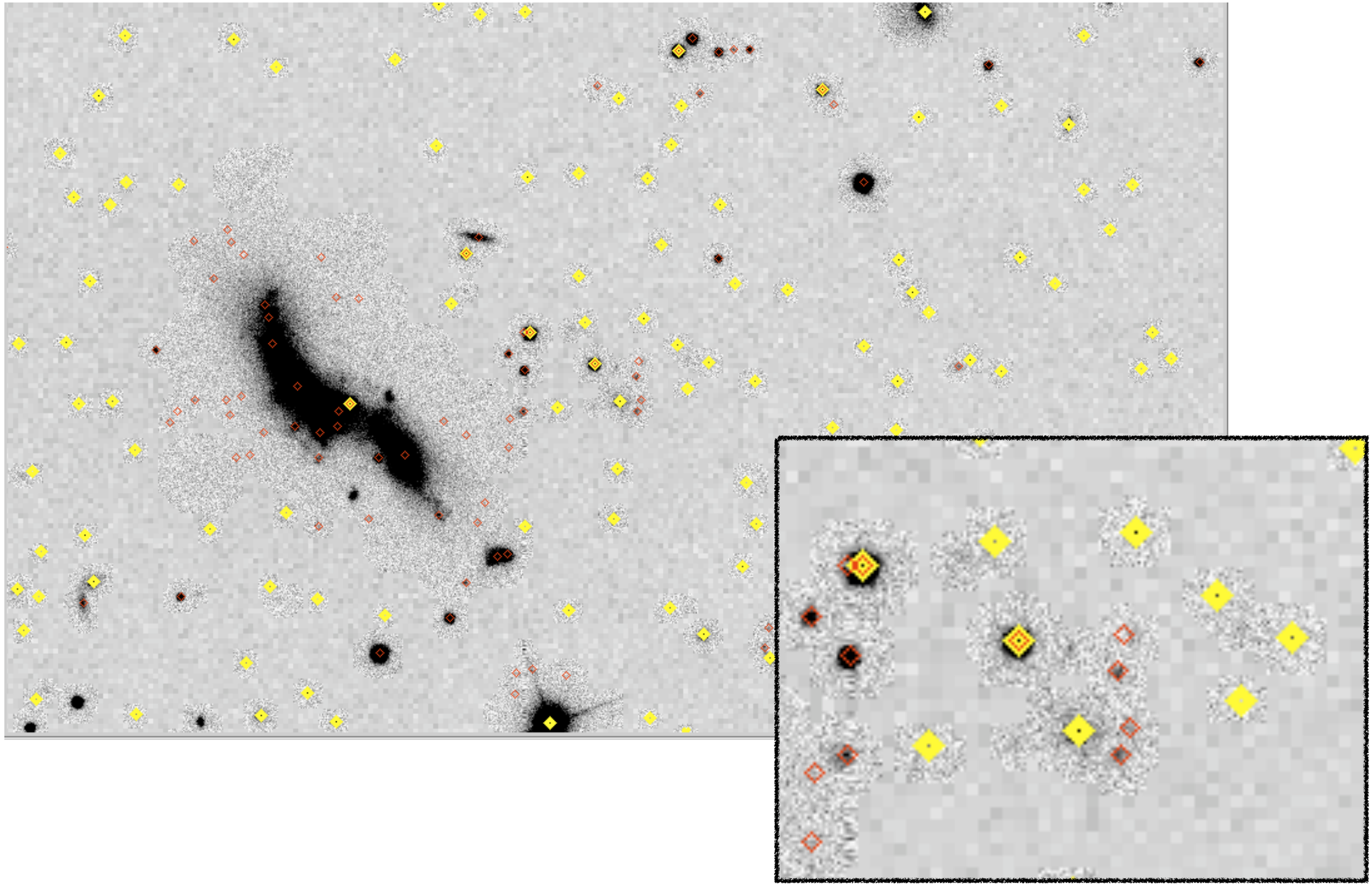
Detected Pixels



Parent objects



Parents and children



photoField quantities

http://data.sdss3.org/datamodel/files/BOSS_PHOTOOBJ/RERUN/RUN/photoField.html

Field identifier tags:

fieldID (string): Field identifier composed of [run, camcol, field, rerun, skyversion].
run (int16): Run number
rerun (int16): Rerun number
camcol (int8): Camera column
field (int16): Field number

Some other very useful tags:

mjd[5] (float64): MJD(TAI) when row 0 of each band was read
raMin (float64): Minimum RA of field (deg)
raMax (float64): Maximum RA of field (deg)
decMin (float64): Minimum Dec of field (deg)
decMax (float64): Maximum Dec of field (deg)

photoObj quantities

http://data.sdss3.org/datamodel/files/BOSS_PHOTOOBJ/RERUN/RUN/CAMCOL/photoObj.html

Object identifier tags:

objID (string): Unique SDSS identifier composed from [skyVersion,rerun,run,camcol,field,obj]
fieldID (string): Field identifier composed of [run, camcol, field, rerun, skyversion].
parentID (string): Object identifier of parent (if object deblended) or BRIGHT detection (if object has one), else 0
run (int32): Run number
rerun (string): Rerun number
camcol (int32): Camera column
field (int32): Field number
id (int32): The object id within a field. Usually changes between reruns of the same field
parent (int32): parent ID, if there is one

Position tags

ra (float64): J2000 Right Ascension (r-band) (deg)
dec (float64): J2000 Declination (r-band) (deg)
objc_rowc (float32): Row center position (r-band coordinates) (pix)
objc_rowcerr (float32): Row center position error (r-band coordinates) (pix)
objc_colc (float32): Column center position (r-band coordinates) (pix)
objc_colcerr (float32): Column center position error (r-band coordinates) (pix)
TAI[5] (float64): time of observation (TAI) in each filter (sec)

photoObj quantities

http://data.sdss3.org/datamodel/files/BOSS_PHOTOOBJ/RERUN/RUN/CAMCOL/photoObj.html

Type & quality tags:

objc_type (int16): Type classification of the object (star, galaxy, cosmic ray, etc.)
- 0 (OBJ_UNK): An object of unknown type.
- 1 (OBJ_CR): Not used.
- 2 (OBJ_DEFECT): Not used.
- **3 (OBJ_GALAXY): Object is classified as a galaxy**
- 4 (OBJ_GHOST): Not used.
- 5 (OBJ_KNOWNOBJ): Not used.
- **6 (OBJ_STAR): Object is classified as a star**
- 7 (OBJ_TRAIL): Not used.
- 8 (OBJ_SKY): Empty part of the image designated for sky.

objc_flags (int32): photo object attribute flags (see http://www.astro.princeton.edu/~rhl/flags.html)
objc_flags2 (int32): Second set of photo object attribute flags
fracDeV[5] (float32): Weight of deV component in deV+Exp model
psf_fwhm[5] (float32): PSF FWHM (arcsec)
resolve_status (int32): says whether it is unique within a run

Flux tags:

psfMag[5] (float32): PSF magnitude (mag)
psfMagErr[5] (float32): PSF magnitude error (mag)
psfFlux[5] (float32): PSF flux (nanomaggies)
psfFluxIvar[5] (float32): PSF flux inverse variance (nanomaggies)

best #s for point sources

petroMag[5] (float32): Petrosian magnitude (mag)
petroMagErr[5] (float32): Petrosian magnitude error (mag)
petroFlux[5] (float32): Petrosian flux (nanomaggies)
petroFlux_Ivar[5] (float32): Petrosian flux inverse variance (nanomaggies)

model-independent galaxy fluxes (good for r < 19 or so)

cModelMag[5] (float32): DeV+Exp magnitude (mag)
cModelMagErr[5] (float32): DeV+Exp magnitude error (mag)
cModelFlux[5] (float32): better of DeV+Exp flux (nanomaggies)
cModelFlux_Ivar[5] (float32): Inverse variance in DeV+Exp flux fit (nanomaggies)

best #s for faint galaxies

Using the type and quality flags

http://data.sdss3.org/datamodel/files/BOSS_PHOTOOBJ/RERUN/RUN/CAMCOL/photoObj.html

http://www.astro.princeton.edu/~rhl/flags.html

You will want to select unique objects in each run. This requirement means you don't want any parents and that you don't want any objects that are in the overlap areas between frames. In practice there is a flag for each object "clean" which is 1 if the object is clean and unique, 0 otherwise.

What does this do? Checks bits in flags variables to get child objects:

```
((objc_flags & BLEND) == 0  || (objc_flags & NODEBLEND) != 0) &&   not deblended further
(objc_flags & BRIGHT) == 0                                        not a "BRIGHT" detection
```

and make sure objects aren't in overlap region between frames:

```
(objc_rowc >= 64 && objc_rowc < 1489-64)                          in unique region of field
```

and finally ignore object types we don't care about:

```
(objc_type == 3 || objc_type == 6)                                is a star or galaxy
```

Finally, there are a number of other bits that are interesting to check, e.g.

SATUR SATUR_CENTER EDGE MAYBE_CR

Fluxes & units

http://data.sdss3.org/datamodel/files/BOSS_PHOTOOBJ/RERUN/RUN/CAMCOL/photoObj.html

Notice that for each type of flux there are two versions, e.g.

psfFlux[5] and psfMag[5]

The "fluxes" are in units of AB nanomaggies. That is:

$$\mathrm{flux} = 10^9 \times \frac{f_\nu}{3631\ \mathrm{Jy}}$$

The magnitudes are approximately the standard AB magnitude:

$$\mathrm{mag} = 22.5 - 2.5 \log_{10} \mathrm{flux}$$

But in fact are actually "asinh" magnitudes:

$$\mathrm{mag} = -\frac{2.5}{\ln 10}\left[\sinh^{-1}(2b \times \mathrm{flux}) + \ln(b)\right].$$

asinh Softening Parameters (*b* coefficients)

Band	*b*	Zero-Flux Magnitude [$m(f/f_0 = 0)$]	$m(f/f_0 = 10b)$
u	1.4×10^{-10}	24.63	22.12
g	0.9×10^{-10}	25.11	22.60
r	1.2×10^{-10}	24.80	22.29
i	1.8×10^{-10}	24.36	21.85
z	7.4×10^{-10}	22.83	20.32

brighter than this differences are less than 1%

Fluxes & units

http://data.sdss3.org/datamodel/files/BOSS_PHOTOOBJ/RERUN/RUN/CAMCOL/photoObj.html

Note also that for flux we specify inverse variance, but for magnitude we specify errors!

psfFlux_Ivar[5] and psfMag_Err[5]

Obviously inverse variance is just the inverse squared error.

Note, however, that for objects with bad measurements, the error is set to -9999, and the inverse variance is set to zero.

You'll get used to the inverse variance way of doing things, and I dare say learn to love it.

- Fiber fluxes
- Close to simple aperture magnitudes corresponding to 2 or 3 arcsec diameter spectroscopic fiber:
  - `FIBER2FLUX`
  - `FIBERFLUX`

- PSF fluxes
  - `PSFFLUX`
- Right choice for point sources.
- Uses PSF modeled within field for each band and fits amplitude of that model to get flux

- Model fluxes
  - `DEVFLUX`
  - `EXPFLUX`
  - `MODELFLUX`
- de Vaucouleurs & exponential 2D models
- Also yields
  - *half-light radius (pretty good)*
  - *position angle (eh)*
  - *axis ratio (not bad)*
- Accounts for PSF
- "Model" version is better fit of two.

- "Cmodel" fluxes
  - `FRACDEV`
  - `CMODELFLUX`
- Combine best-fit exp. + best-fit de Vauc without refitting shapes, just fluxes
- Probably the best default flux; works OK for biggest and smallest objects.

- Petrosian fluxes
  - `PETROFLUX`
- Model-independent flux
- Adaptive aperture flux, with an aperture that doesn't depend on surface brightness

- Good for reasonably bright (but not too bright!) sources

Petrosian ratio

$$\mathscr{R}_{\rm P}(r) \equiv \frac{\int_{0.8r}^{1.25r} dr'\, 2\pi r' I(r') / [\pi(1.25^2 - 0.8^2) r^2]}{\int_0^r dr'\, 2\pi r' I(r') / (\pi r^2)} \ ,$$

$$F_{\rm P} \equiv \int_0^{N_{\rm P} r_{\rm P}} 2\pi r'\, dr'\, I(r') \ .$$

$N_P = 2$

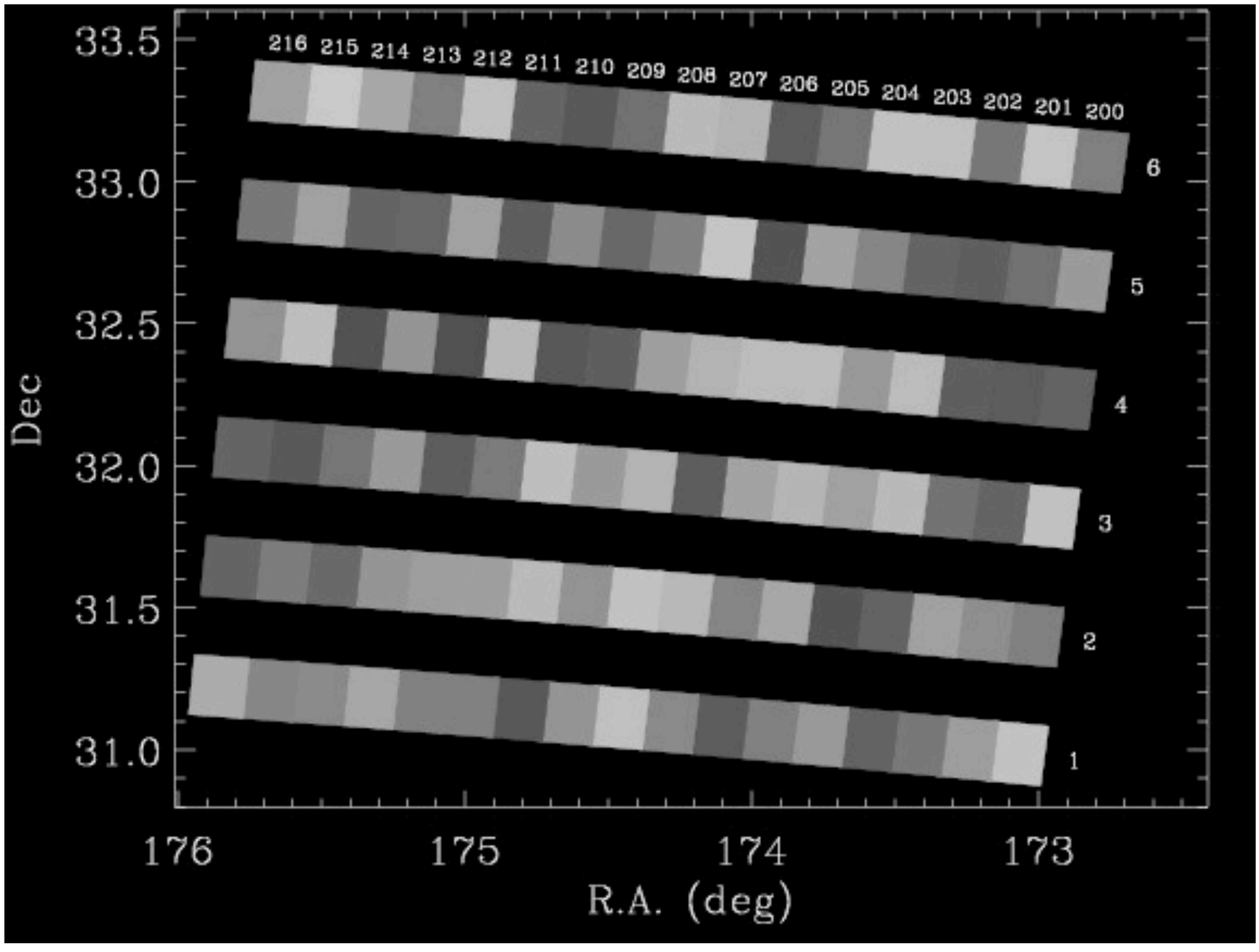
r_P is where Petrosian ratio is 0.2

- Note: for big objects (half-light radii > 10 arcsec) some issues with sky subtraction cause underestimated flux
- Redressed partially at `http://nsatlas.org`

- Star/galaxy separation
- Based on comparison of PSFFLUX with MODELFLUX
- If model magnitude is more than ~ 0.15 mag brighter, call it a galaxy
  - Works quite well

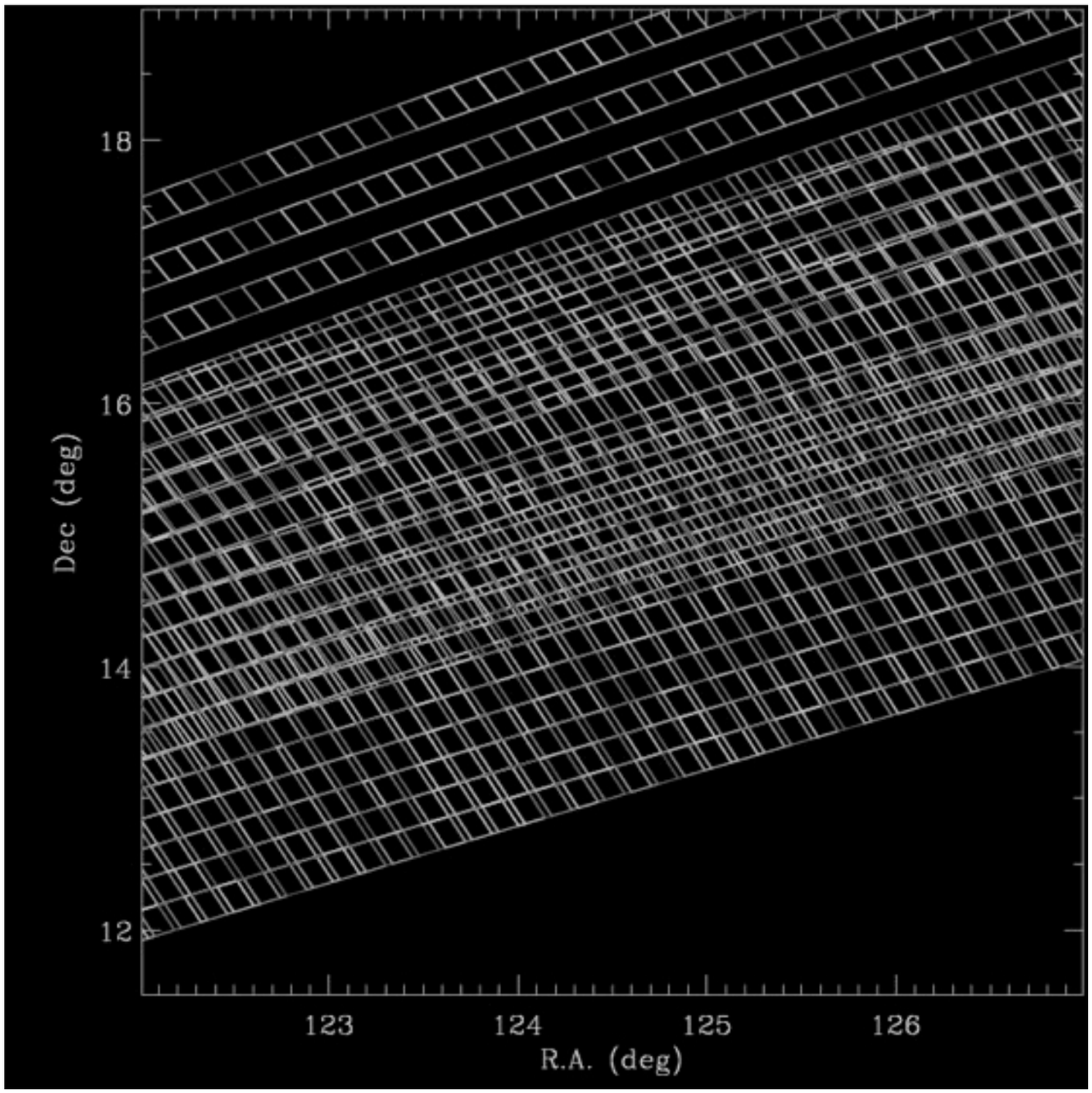
```
plt.plot(dd[1].data.field('cmodelmag')[:,2],
 dd[1].data.field('psfmag')[:,2]-dd[1].data.field('modelmag')[:,2],
',')
```

RESOLVE_STATUS



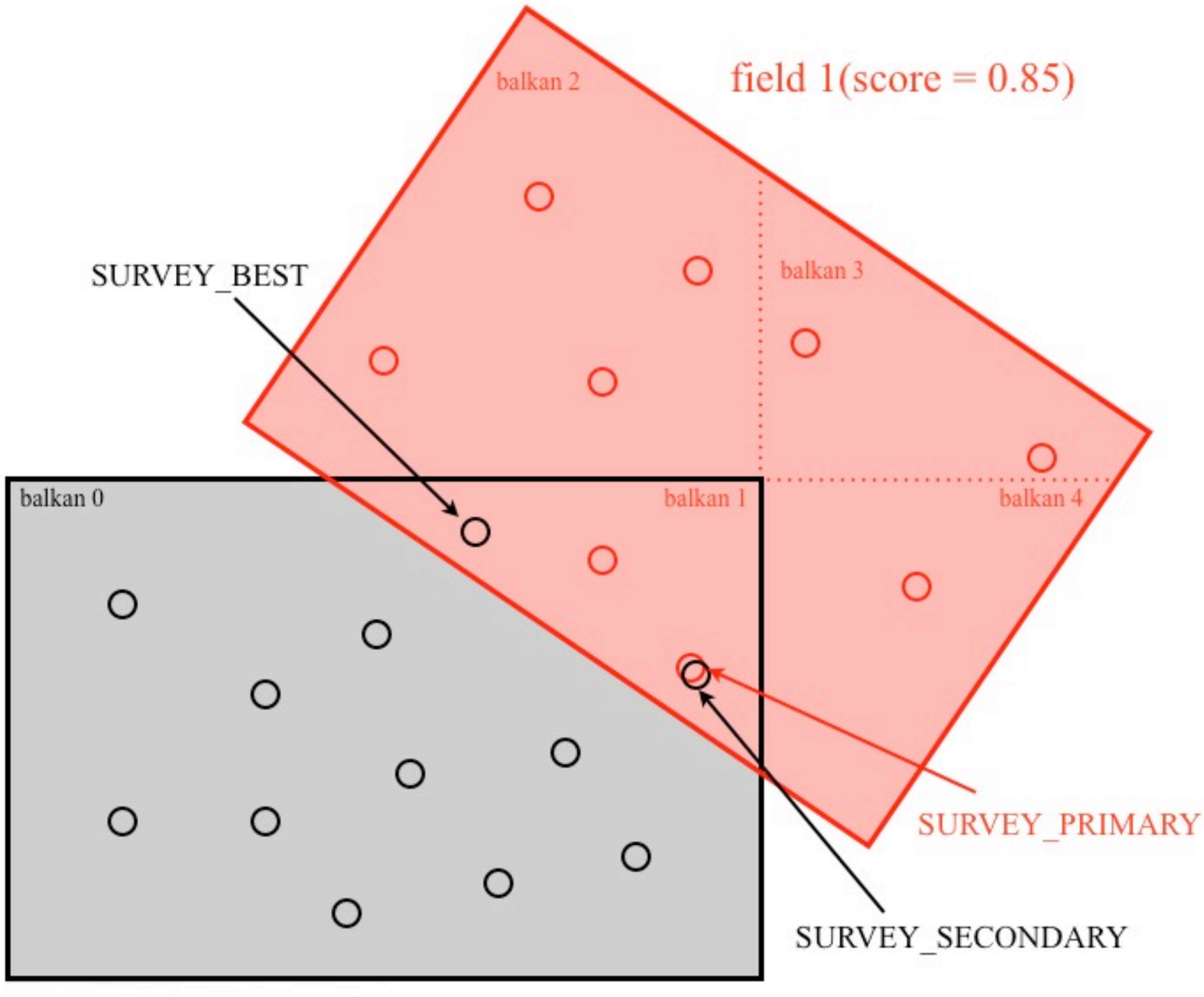
RUN_PRIMARY gives a unique set of objects across a run

RESOLVE_STATUS



SURVEY_PRIMARY gives a unique set of objects across the survey

RESOLVE_STATUS



- Data sweeps
- Compact form of photometric data in flat file form (300 Gb)
- Doesn't have all columns
- Organized by run (but there are some tricks if you need to search by RA/Dec)

http://data.sdss3.org/sas/dr9/boss/sweeps/dr9/

Afternoon Activity

- Searching for star and galaxy clusters
  - *write code to measure density of stars and galaxies within a given field*
  - *write code to check if there is a clump of stars (or of galaxies) on ~ arcmin scales in the field*
  - *run on several hundred fields*
  - *look at images for any detected cases*
  - *overplot star and galaxy positions on the image*
  - *check for correlation of star and galaxy densities with PSF width, sky brightness, Galactic coordinates*

hints:
- use RUN_PRIMARY
- you should pick a flux limit

do something simple!!

WCS

try to figure out what you should see