

# Report on Winter2014 Production: Image Differencing

March 13, 2014

Summary

# Contents

<b>1 Production Scope and Goals</b>	<b>3</b>
1.1 Subtask 1: Reproduce W13 Results Using Current PhoSim . . . . .	3
1.2 Subtask 2: Simulate Starfield using Multiple SEDs at a Single Airmass . . . . .	3
1.3 Subtask 3: Simulate Starfield using Multiple SEDs at Multiple Airmasses . . . . .	3
1.4 Subtask 4: Include Realistic Mix of Stars and Galaxies . . . . .	3
<b>2 Review of Wavelength Dependent Refraction</b>	<b>3</b>
<b>3 Running PhoSim</b>	<b>3</b>
3.1 Reference SEDs . . . . .	3
<b>4 Analysis of Subtask 1</b>	<b>3</b>
<b>5 Analysis of Subtask 2</b>	<b>3</b>
<b>6 Analysis of Subtask 3</b>	<b>3</b>
<b>7 Possible Solutions to Dcr Issue</b>	<b>3</b>
<b>Appendices</b>	<b>19</b>
<b>A Use of PhoSim</b>	<b>19</b>
A.1 Setting up PhoSim . . . . .	19
A.2 Generation of Instance Catalogs for W14 . . . . .	19
A.2.1 General . . . . .	19
A.2.2 Task A . . . . .	19
A.2.3 Task B . . . . .	19
A.2.4 Task C . . . . .	19
A.3 Running phosim.py . . . . .	20
A.4 Creating Astrometry Index Files . . . . .	20
A.5 Running processEimage.py . . . . .	20
<b>B ImageDifferenceTask</b>	<b>20</b>

## **1 Production Scope and Goals**

- 1.1 Subtask 1: Reproduce W13 Results Using Current PhoSim
- 1.2 Subtask 2: Simulate Starfield using Multiple SEDs at a Single Airmass
- 1.3 Subtask 3: Simulate Starfield using Multiple SEDs at Multiple Airmasses
- 1.4 Subtask 4: Include Realistic Mix of Stars and Galaxies

Pushed back to S14.

## **2 Review of Wavelength Dependent Refraction**

## **3 Running PhoSim**

### **3.1 Reference SEDs**

## **4 Analysis of Subtask 1**

## **5 Analysis of Subtask 2**

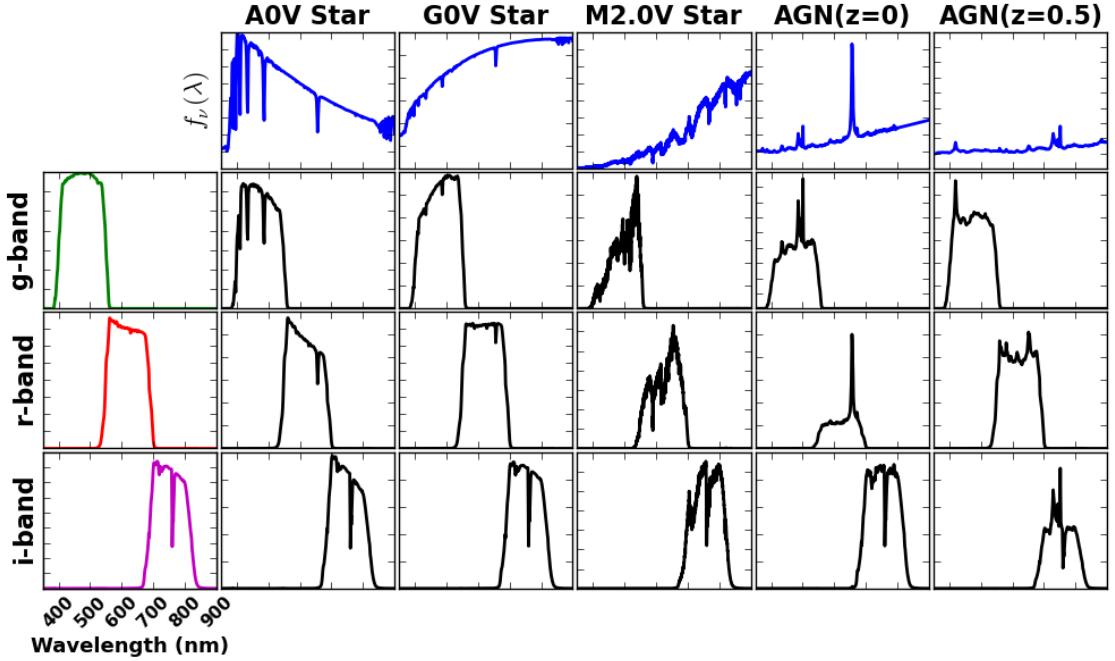
## **6 Analysis of Subtask 3**

## **7 Possible Solutions to Dcr Issue**

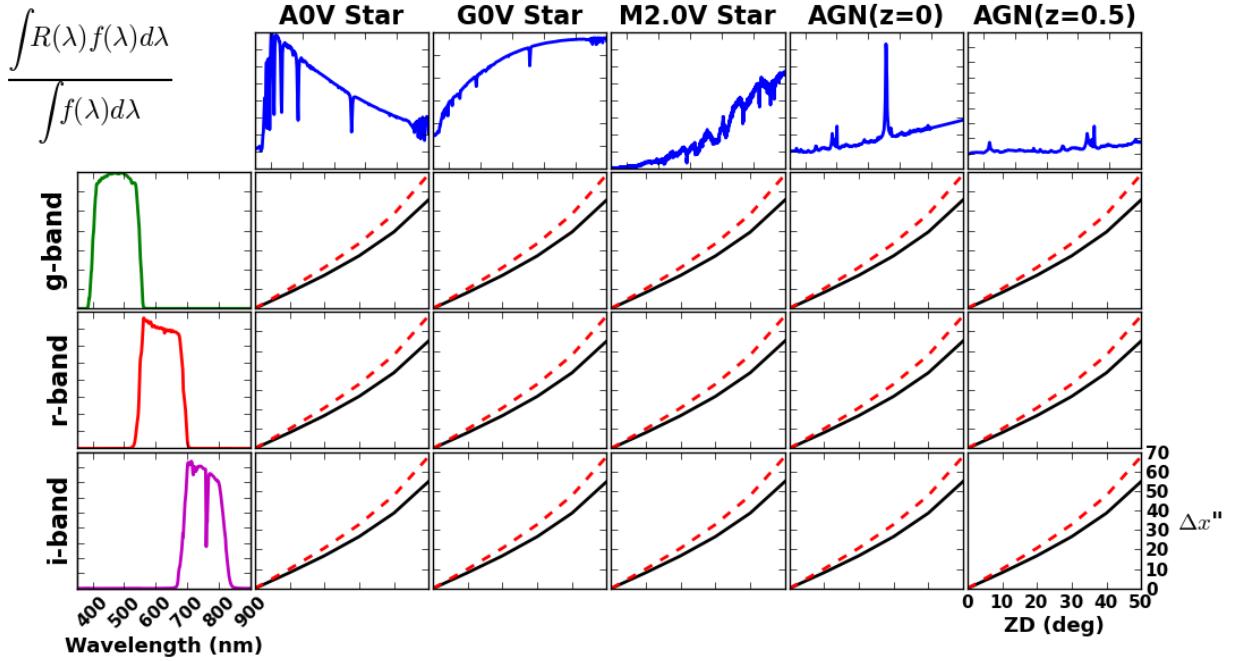
Remapping each template image pixel by pixel, depending on color

Color-dependent Psf matching kernel (does not fix template issue)

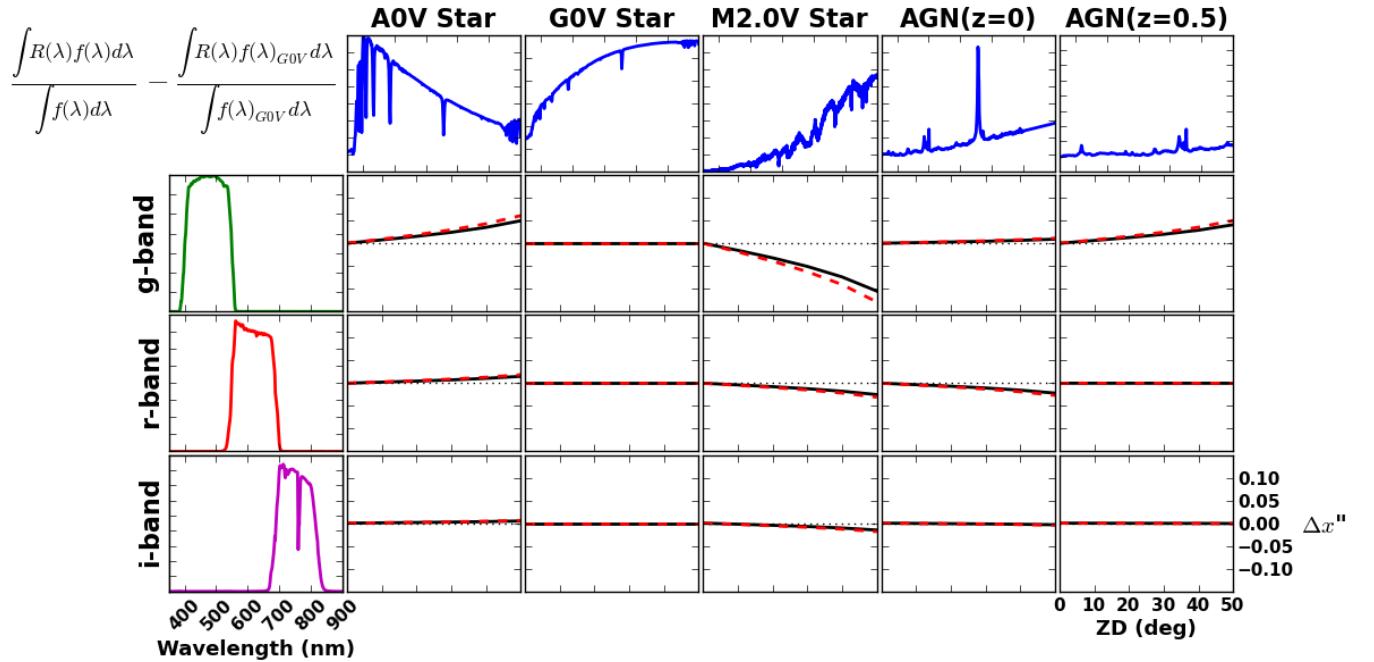
Dipole fit that restricts the angle of separation to be along Dcr vector



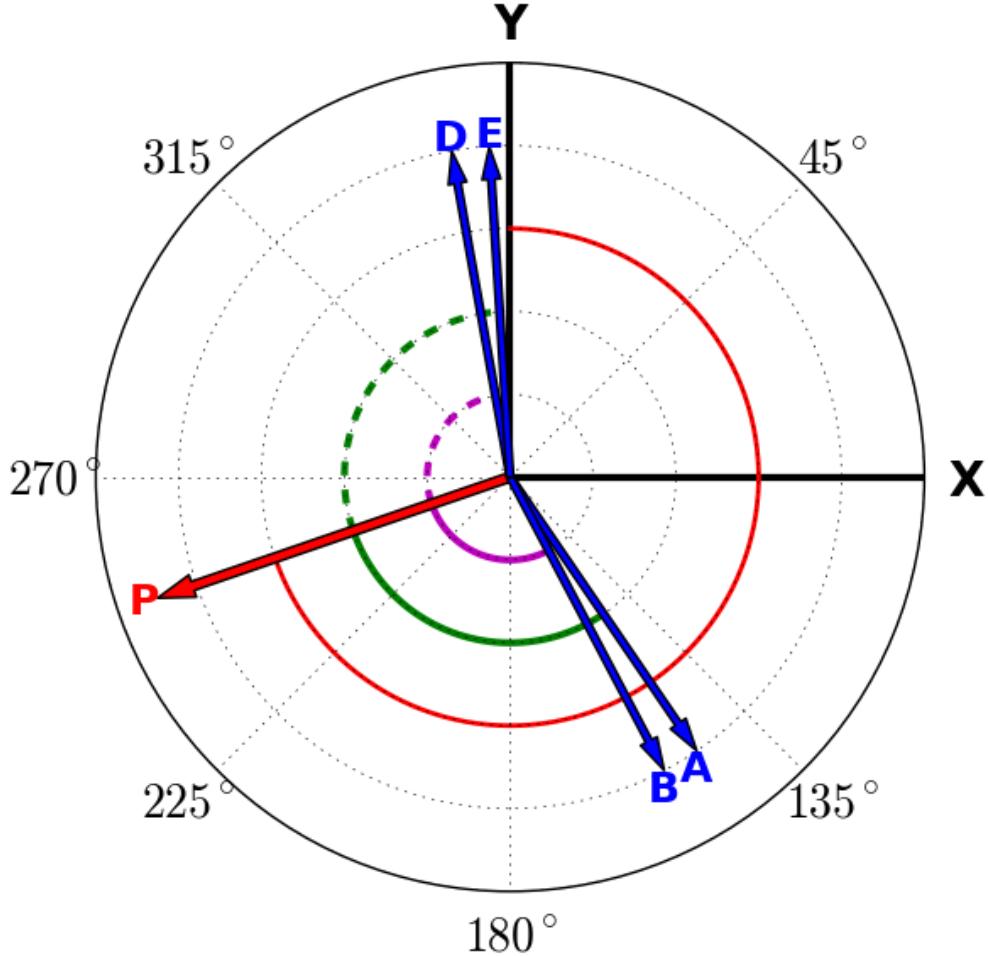
**Figure 1: Effective Spectral Energy Distributions :** The effective spectra of 5 reference objects – a “blue” A0V, a reference G0V, and a “red” M2.0V star, along with a QSO at redshift  $z = 0$  and  $z = 0.5$  – filtered through 3 transmission profiles corresponding to the LSST  $g$ ,  $r$ , and  $i$ -bands. The top row shows the input spectral energy distribution  $f_\nu(\lambda)$ , while the leftmost column shows the LSST filter transmission profile in units of the normalized system response  $\phi$ . The inner row/column figures show the effective spectrum of each SED (along columns) when multiplied through the respective filter (along rows). In all subpanels, the x-axis is wavelength. The A0V, G0V, and M2.0V spectra correspond to CAT\_SHARE\_DATA files `kp01_9750.fits_g45_9830.gz`, `km20_6000.fits_g30_6020.gz`, and `m2.0Full.dat.gz` respectively, and were used as the SEDs of the stars in the W14 image simulations. This figure may be recreated using the script `python/DCR.py`.



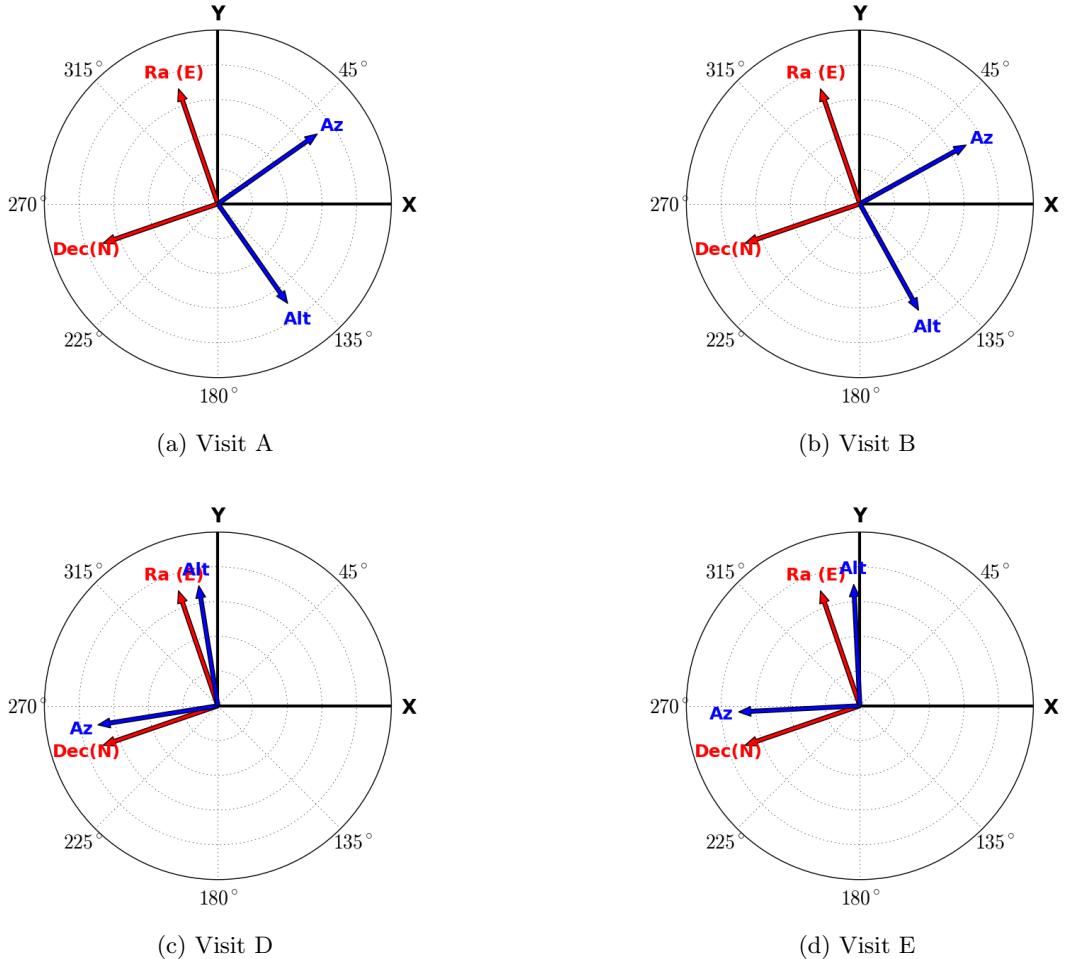
**Figure 2: Refraction Amplitude vs. Filter and Spectral Energy Distribution:** The flux-weighted amplitude of refraction (in arcseconds) for each of the filtered SEDs in Figure 1, as a function of zenith distance in degrees along the x-axis. The solid black line is the nominal result from Eqn ??, while the dashed red line ignores the corrections for temperature and pressure. Note the maximum amplitude of refraction reaches nearly 1 arcminute. This figure may be recreated using the script `python/DCR.py`.



**Figure 3: Differential Chromatic Refraction vs. Filter and Spectral Energy Distribution:** The differential chromatic refraction of all sources from Figure 1 with respect to the reference G0V star, with respect to zenith distance. The maximum amplitude of DCR reaches  $0.1''$ , or approximately half an LSST pixel. This figure may be recreated using the script `python/DCR.py`.



**Figure 4: Designed Orientation of Dcr in W14 Phosim Runs:** This figure represents the anticipated orientations of Dcr in the W14 phoSim data. The x,y coordinate system is depicted, as well as the convention that angles (`rotTelPos`, `rotSkyPos`) are clockwise with respect to the positive y-axis in the image coordinate system (counterclockwise in the camera coordinate system). The `rotTelPos` of 251 degrees specified for all simulations, which reflects the direction to the pole, is shown with the red vector Pand the red arc at  $y=0.6$ . The derived `rotSkyPos` for visits A,B,D,E are shown with the blue vectors, and reflect the angle towards zenith (the angle of increasing altitude). Dcr is expected to happen along these vectors. The angles PA,PE are similar, and represented by the green arcs; the angles PB,PD are also similar, and represented by the purple arcs. This is expected as observations A and E are taken at airmass 1.55 (zenith distance of 50 degrees) but at opposite sides of the meridian crossing of the star field; a similar situation was designed for observations B and D, which are taken at airmass 1.15 (zenith distance of 30 degrees). Visit C is not depicted as it was taken at zenith. This figure was created using the script `python/dcrSchematic.py`.



**Figure 5: Wcs-Derived Orientations of Phosim Data:** These figures show the orientations of the Right Ascension and Declination axes (red), and Altitude and Azimuth axes (blue) of visits A,B,D,E. Arrows represent the directions of *increasing* coordinate value. The Ra,Decl axes are the same in all images since they were designed to have a common `rotTelPos`. Ideally, the directions of increasing Alt will correspond to the `rotSkyPos` depicted in Figure 4. All coordinate system orientations were derived from the fitted Wcs of the `calexp` of the  $g$ -band observation of seeing values 2, i.e. the worst seeing image. To determine the orientations empirically, small steps were taken in each coordinate starting at the center of the image, and the Wcs and topocentric corrections used to map these back into offsets in the pixel plane. This figure was created using the script `python/compareDcrFromSims.py.py`.

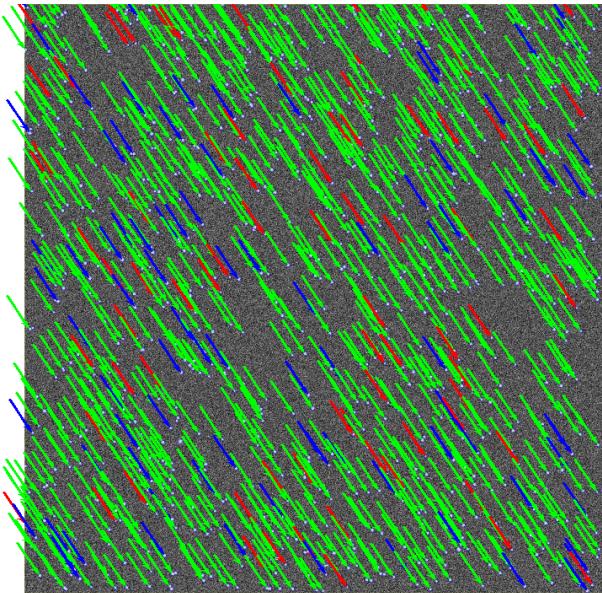
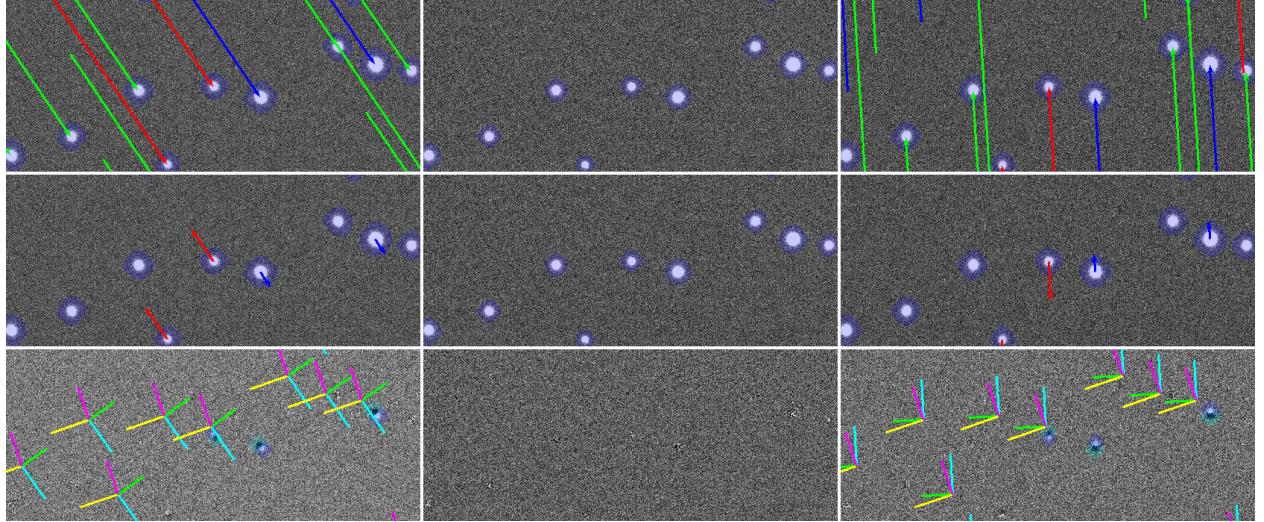
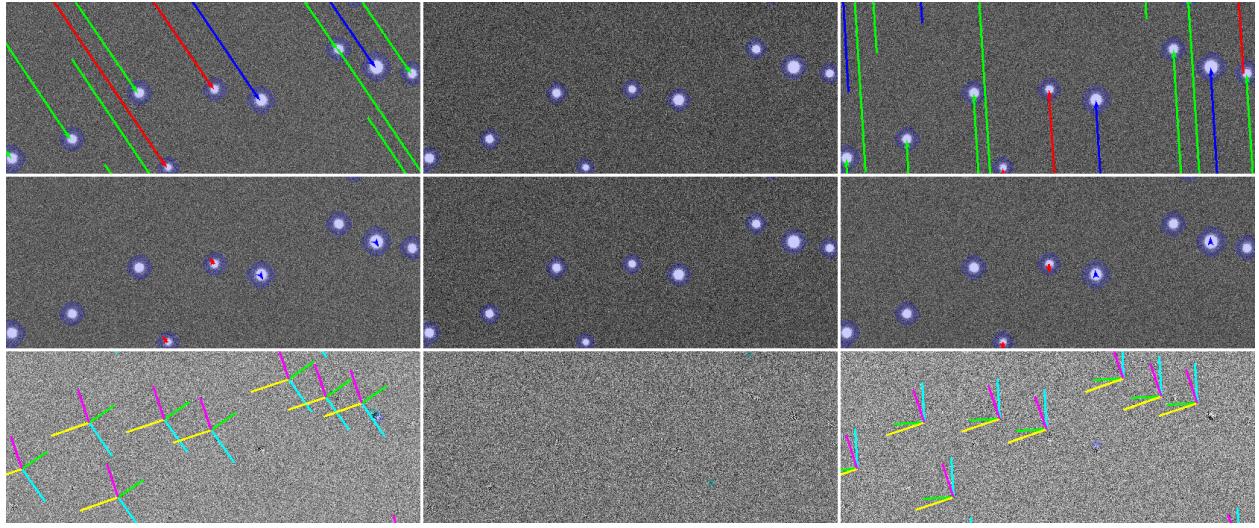


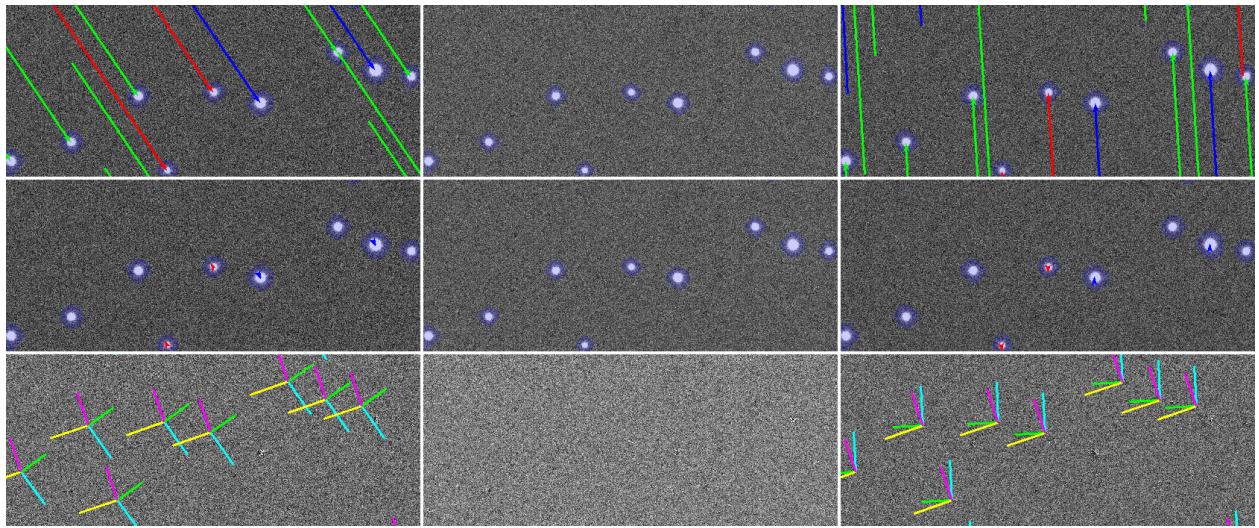
Figure 6: **Refraction:** This figure depicts the amplitude and orientation of the refraction vector in the visit A, `raft=2,2 sensor=1,1 filter=g`, seeing value 2 data. The vectors point from the unrefracted locations to the realized locations in the image. The SEDs of the sources are indicated with colors `blue`, `green` and `red` for AOV, GOV, and M2.0V respectively. This figure was created using the script `python/compareDcrFromSims.py`.



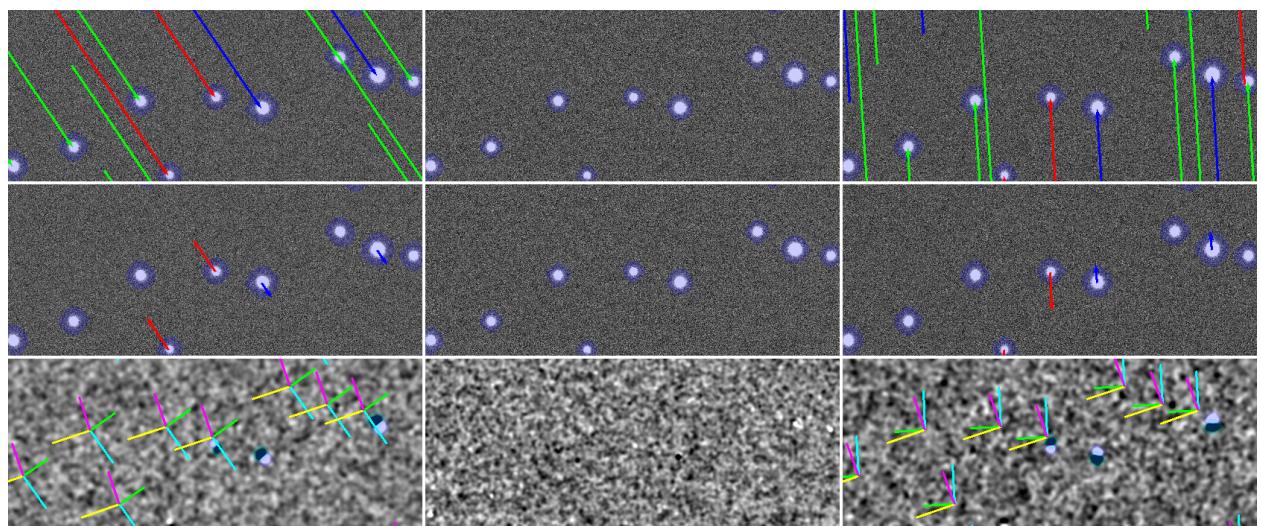
**Figure 7: Differential Chromatic Refraction and Difference Image Quality,  $g$ -band:** This set of panels demonstrates the amount of refraction (top row), differential refraction with respect to the GOV star (middle row), and the quality of the difference image (bottom row) for three sets of image differences. The first column uses science visit A, the middle C, and the third E. In all cases the template used was taken at zenith, i.e. visit C. The top rows effectively present the same information as Figure 6, but zoomed in on a particular cluster of stars. The second row subtracts off the green vector from all vectors. The residual refraction of the blue,red vectors represents differential chromatic refraction. These residual lengths have been multiplied by a factor of 100 for readability. Note that the blue vectors point along the vector to zenith, indicating the blue stars appear higher in the sky than their green counterparts, compared to an unrefracted observation. The red stars are not refracted as much and thus will appear lower in the sky. On the bottom row, we show the realized difference image quality. Note that the blue stars have positive lobes pointing along the direction to zenith, meaning the stars are “higher” in the sky w.r.t. the green stars when compared to the zenith template, while the dipoles of the red stars have the opposite polarity. For completeness, the Wcs-derived orientation of the Ra,Decl and Az,Alt coordinate axes are shown in the difference image (Ra,Decl,Az,Alt are magenta,yellow,green,cyan); see Figure 5 for more detail. This figure was created using the script `python/compareDcrFromSims.py`.



**Figure 8: Differential Chromatic Refraction and Difference Image Quality,  $r$ -band:** Same as Figure 7, but for  $r$ -band data. This figure was created using the script `python/compareDcrFromSims.py`.



**Figure 9: Differential Chromatic Refraction and Difference Image Quality,  $i$ -band:** Same as Figure 7, but for  $i$ -band data. This figure was created using the script `python/compareDcrFromSims.py`.



**Figure 10: Differential Chromatic Refraction and Difference Image Quality, Prefiltering:**  
Same as Figure 7, but using prefiltering of the science image with its Psf. This figure was created using the script `python/compareDcrFromSims.py`.

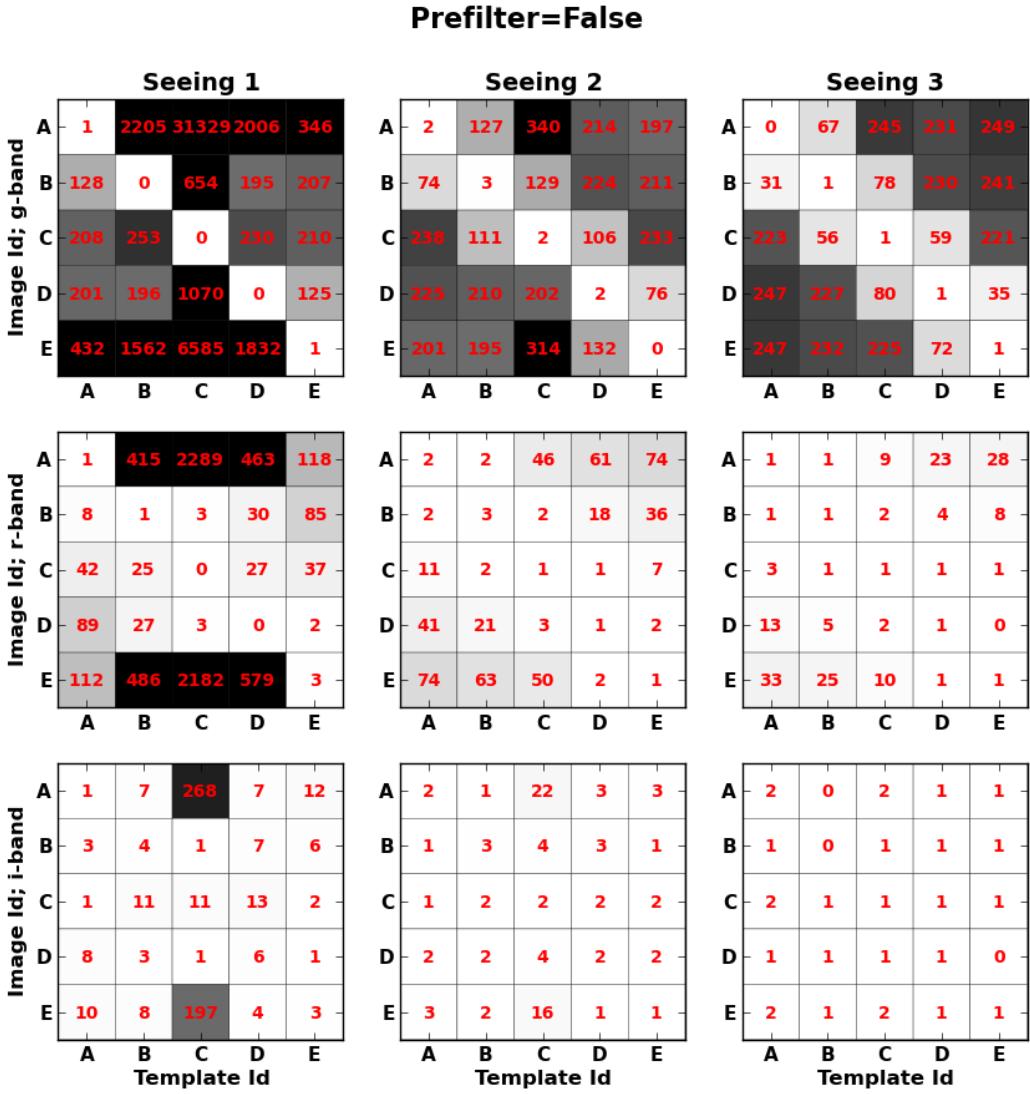


Figure 11: **Number of False Positives, Postfiltering:** This figure shows the median number of false positives across the 9 `raft=2,2` CCDs. The first row of information shows these “heat-maps” for *g*-band data, the second for *r*-band, and the third for *i*-band. The first column represents the good-seeing images, the second the medium-seeing images (same quality as template), and the third the poor-seeing images. Within each filter-seeing combination, the heat-map represents the median number of false positives as a function of the template airmass (visit ABCDE) along the x-axis, and image airmass (visit ABCDE) along the y-axis. The diagonal elements represent the situation where the template and science image are taken at the same airmass and have the same orientation w.r.t. zenith. The off-diagonal elements represent a mismatch between the template and science image in terms of airmass *and* parallactic angle. The general trend is that the numbers of false positives decrease with decreasing differences in the airmass,angle attributes of the template and science image, decrease with increasing seeing, and decrease with increasing wavelength. The *i*-band data in poor seeing do not appear sensitive to Dcr. This figure was created using the script `python/heatMap.py`.

### Prefilter=True

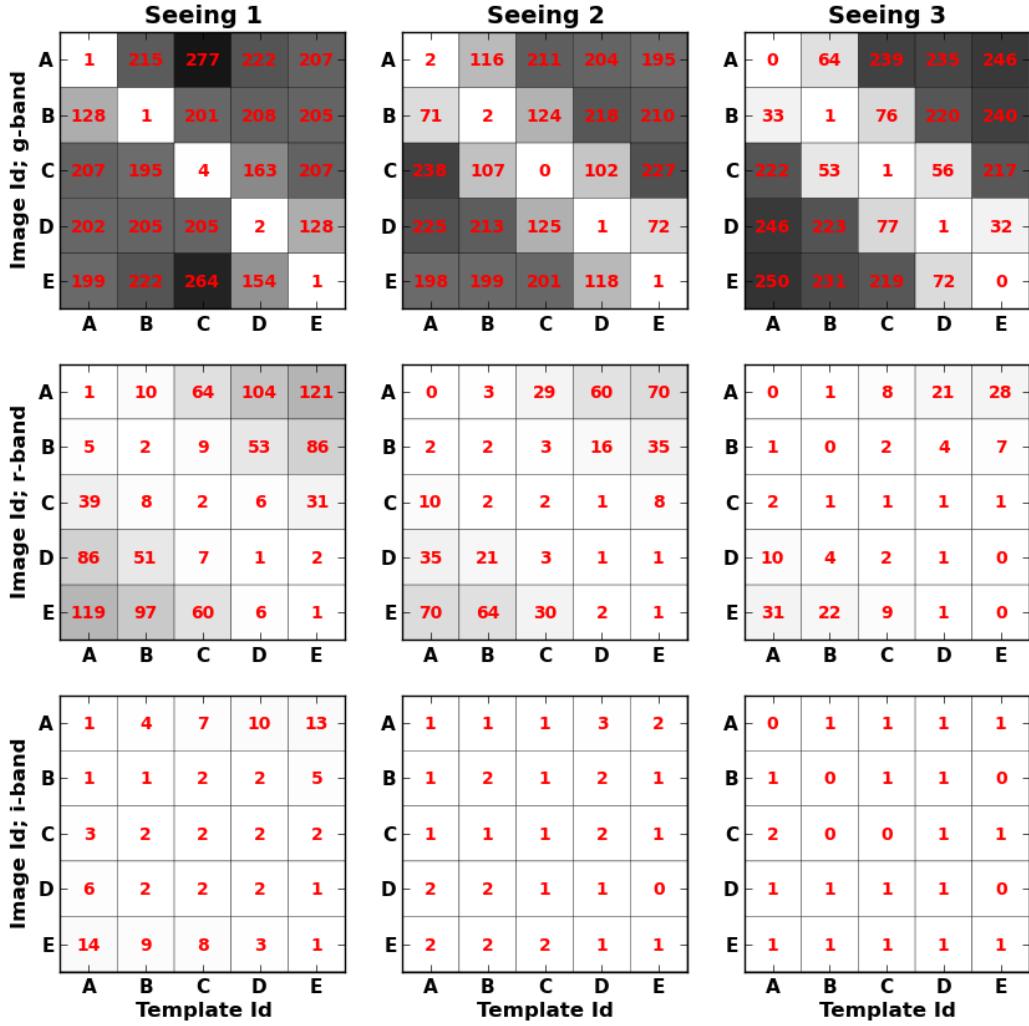


Figure 12: **Number of False Positives, Prefiltering:** Same as Figure 11, but using prefiltering of the science image with its Psf. The numbers of false positives is overall far lower than in the postfiltering case (Figure 11). This figure was created using the script `python/heatMap.py`.

PhoSim Visits					
Visit	MJD	rotSkyPos	rotTelPos	Altitude	Airmass
A	51130.111719	251.0689567	145.7139810	40.1	1.55
B	51130.176302	251.0689567	152.2708405	59.9	1.16
C	51130.272830	251.0689567	341.0975816	89.9	1.00
D	51130.272830	251.0689567	349.8619334	59.9	1.16
E	51130.433247	251.0689567	356.4181128	40.1	1.55

Table 1: Summary of the configuration parameters that define each phoSim visit ABCDE. Observations were designed to follow a single star field through zenith on a single night, and observed twice before and twice after crossing the meridian, as well as at zenith. Each visit was simulated using the same random seed (153555399), in 3 filters ( $g$ ,  $r$ ,  $i$ ), and under 4 observing conditions. These correspond to the template image (`Opsim_rawseeing` = 0.878") and `SIM_VISTIME` = 300) and then 3 bins in seeing for the science image corresponding to `Opsim_rawseeing` = 0.6", 0.878", and 1.2" for seeing bins 1, 2, 3, respectively. All science images were simulated with `SIM_VISTIME` = 15.

Overall, this yielded  $5 \times 3 \times 4 = 60$  runs of phoSim, which are encoded as `visitId=X00Y00Z` where X represents the photSim filterId  $gri=123$ , Y represents the visit ABCDE= 01234, and Z represents the seeing: 0 for the template and 123 for the science images. All 9 CCDs within `raft=2,2` were simulated. All trim files used to generate the sims for this analysis may be found in the `sims8/` directory.

Orientation of Dcr				
Visit	SED	rotTelPos	Wcs + Topo	Measured Dipole Orientation
A	AOV	145.7°	144.9°	145.7°
	M2.OV	325.7°	324.9°	325.6°
B	AOV	152.3°	151.1°	151.7°
	M2.OV	332.3°	331.1°	331.8°
D	AOV	349.9°	351.0°	349.3°
	M2.OV	169.9°	171.0°	170.0°
E	AOV	356.4°	357.1°	355.2°
	M2.OV	176.4°	177.1°	176.2°

Table 2: The expected and measured orientations of Dcr, using the coordinate conventions depicted in Figure 4. These numbers represent the orientation of the *positive* lobe of any dipole arising from the Dcr effect. Numbers are reported as a function of visit A–E, and the spectral energy distribution of the source. For the `rotTelPos` and `Wcs` columns, the numbers for the red M2.OV stars are simply 180° from those of the blue AOV stars, which should be in the direction of zenith. The numbers under `rotTelPos` reflect the designed orientation of Dcr that was input to phoSim. The numbers under `Wcs` reflect the empirically determined direction to zenith in each image, using the `calexp` fitted `Wcs` and topocentric corrections appropriate for each visit. Finally, we report the measured median orientation of DiaSources in each image difference, when differenced against the template taken at zenith (visit C). Here the red and blue populations are considered separately.

These numbers come from analysis of *g*-band data, seeing bin 1, which have the most dipoles. We only show results for 1 filter since the orientation should be filter-independent. Results in the *r*-band, in cases where there are more than 10 dipoles measured, show quantitatively similar results. In addition, we only report numbers from the postfiltered data, where dipole measurement is known to operate correctly. We do not see any dependence of these orientations on the seeing. Full results may be found in the file NOTES.

Amplitude of Dcr							
Visit	SED	Filter	Theory	Wcs Offsets	Dipole Amplitude Seeing Bin 1	Dipole Amplitude Seeing Bin 2	Dipole Amplitude Seeing Bin 3
A	AOV	<i>g</i>	0.049"	0.042"	0.167"	0.259"	0.208"
		<i>r</i>	0.015"	0.013"	0.049"	...	...
		<i>i</i>	0.005"	0.005"	...	...	...
	M2.OV	<i>g</i>	0.105"	0.077"	0.144"	0.204"	0.293"
		<i>r</i>	0.025"	0.020"	0.230"	...	...
		<i>i</i>	0.015"	0.012"	...	...	...
B	AOV	<i>g</i>	0.024"	0.020"	0.183"	0.243"	...
		<i>r</i>	0.007"	0.006"	...	...	...
		<i>i</i>	0.002"	0.002"	...	...	...
	M2.OV	<i>g</i>	0.051"	0.038"	0.222"	0.266"	...
		<i>r</i>	0.012"	0.010"	...	...	...
		<i>i</i>	0.007"	0.005"	...	...	...
D	AOV	<i>g</i>	0.024"	0.020"	0.201"	0.251"	...
		<i>r</i>	0.007"	0.006"	...	...	...
		<i>i</i>	0.002"	0.002"	...	...	...
	M2.OV	<i>g</i>	0.051"	0.038"	0.201"	0.254"	...
		<i>r</i>	0.012"	0.010"	...	...	...
		<i>i</i>	0.007"	0.006"	...	...	...
E	AOV	<i>g</i>	0.049"	0.042"	0.169"	0.235"	0.360"
		<i>r</i>	0.015"	0.013"	0.135"	...	...
		<i>i</i>	0.005"	0.005"	...	...	...
	M2.OV	<i>g</i>	0.105"	0.077"	0.158"	0.213"	0.267"
		<i>r</i>	0.025"	0.020"	0.262"	...	...
		<i>i</i>	0.015"	0.011"	...	...	...

Table 3: The expected and measured amplitudes of Dcr, in arcseconds. These numbers represent the differential offset between positions of an object at zenith (visit C) and at the airmasses associated with visits ABDE, with respect to the positions of the reference GOV stars, which define the astrometric reference system in this study. The Theory column is the expected amplitude as described in Section 2, using the reference SEDs described in Section 3.1. The Wcs column represents the mean offsets between stars of the given SED and the astrometric reference solution, determined using calexp’s Wcs and Source products. Residuals of the reference GOV stars are smaller than 0.002" in all cases. The dipole amplitudes are determined using the same fits that yield the dipole orientations in Table 2, and represent the offset between the fitted positive and negative lobes of the dipole. These numbers are only reported for data sets that contain more than 10 measurements. Because we see a seeing dependence on these numbers, we report them for each of the seeing bins 1,2,3. Full results may be found in the file NOTES.

## References

# Appendices

## A Use of PhoSim

I describe here the end-to-end process of generating instance (or trim) files for `phoSim`, running these images to create output `eImages`, creating `astrometry.net` index files for astrometric calibration, and running these images through `processEimage.py`. This does *not* include the process of generating instance catalogs from a master base catalog, or the process of using calibration data to run the simulated images through `processCcd.py`, which does (amongst other operations) the assembly of amp images into CCD images and instrument signature removal. The process is described for `phoSim` version 3.3.2.

### A.1 Setting up PhoSim

### A.2 Generation of Instance Catalogs for W14

#### A.2.1 General

exposure times single snap control file header seeing

#### A.2.2 Task A

We describe our first step in the `phoSim` process, the generation of the input configuration. We start with the trim files used in W13 processing, which consist of a random star field populated by stars of SED `km50_5000.fits_g20_5140.gz`, and covering a range of magnitudes  $19 < r < 21$ . Observations were designed to be at a zenith distance of  $20.2^\circ$ , with a boresight pointing of (Ra, Decl) =  $(79.68926^\circ, -9.70229^\circ)$ . Observations were simulated in the *i*-band. These trim files are stored in the `sims3` directory in this repository.

#### A.2.3 Task B

We chose spectrum `km20_6000.fits_g30_6020.gz` for our reference GOV-star, and replaced 80% of the objects in the original trim file with this SED. We then selected the bluest object contained in the base catalog, `kp01_9750.fits_g45_9830.gz`, for every tenth object starting at object #0, and the most populous red object in the catalog `m2.0Full.dat.gz` for every tenth object starting at object #5. These correspond to spectral types AOV and M2.OV, respectively.

Because the magnitudes in the trim files represent a 500nm magnitude (approximate *g*-band), it was required to make color corrections to the requested magnitudes of the objects to preserve the relative brightness distribution of the as-simulated, multi-SED sources. This was done using their respective *i*-band magnitudes (i.e. a correction of  $-2.5 \log_{10} (f_i; \text{SED} / f_i; \text{GOV})$ ). Accordingly, the brightness distributions in the *gr*-bands were not optimal (this was fixed in Task C). The colors of each spectral type were calculated using `python/colors.py`. Script `makeNewTrims.py` was used to make these modifications to the `sims3/` trim files, yielding `sims5/` trim files (*i*-band). The only modifications made to the input trim files to simulate images in the *gr*-bands were to change the `Opsim_filter` field to indicate `gri = 123`; these are stored in the `sims5gr/` directory.

#### A.2.4 Task C

First, to have similar brightness distributions in all data, we make a SED-dependent *and* passband-dependent magnitude correction to the trim files used for this task. These corrections are made in

script `generateTrimFilesDcr.py`, which ran on the  $i$ -band trim files from `sims5`.

Second, the original observations do not pass through zenith on the given night of simulated observations. To design the observations to pass through zenith at the LSST site, which is at southern latitude  $-29.67^\circ$ , we first subtracted  $19.96437^\circ$  from all coordinates in the input trim files so that the field is centered at a declination equal to the Southern latitude of the site. This includes both the per-object coordinates and the boresight pointing `Unrefracted_Dec` in the trim file header.

We simulated this star field throughout the night of MJD 51130 (the night of the fiducial image simulations) to establish the times at which it was at an altitude of 40 and 60 degrees (zenith distance of 50 and 30 degrees; airmass of 1.55 and 1.15), and when it was closest to zenith. We chose 5 specific times at which to simulate the images: before meridian crossing at airmass 1.55; before meridian crossing at airmass 1.15; closest to zenith; after meridian crossing at airmass 1.15; after meridial crossing at airmass 1.55).

Because the instance catalog inputs to `phoSim` form an overcomplete set (e.g. the user specifies Ra, Decl, altitude, azimuth, and time of observation), it is possible to request a simulation configuration that is not physically possible. For this reason, we used

`lsst.sims.catalogs.measures.example_utils.makeObsParamsRaDecSky` subroutine  
`makeObsParamsRaDecSky` to generate a self-consistent set of observation parameters given the Ra  
and Decl of the field, and the times of observation established above.

In addition, this script synchronized the values of `Opsim_rotskypos` and `Opsim_rottelpos`. `Rotskypos` sets the orientation of Ra and Decl of the catalog, effectively a rotation of the Pole away from the Zenith. When `rotskypos=0` the Ra direction is “up”, along the y-axis. This value was kept the same for all images, so that the star field was always rendered in approximately the same orientation. `Rottelpos` is defined as (`rotskypos-180+parallactic angle`), such that the angle of DCR shoudl be along this axis. This will also be the angle of increasing Altitude in the Az,Alt coordinate system. This value is observation dependent, and was determined using `makeObsParamsRaDecSky`. Trim files for this run are found in the `sims8/` directory of this repository.

### A.3 Running phosim.py

#### A.4 Creating Astrometry Index Files

## A.5 Running processEimage.py

## B ImageDifferenceTask

We used `$PIPE_TASKS_DIR/bin/imageDifferenceWinter2013.py`, which has an additional flag that allows us to use a particular visit ID as the template, instead of extracting one from the template archive.