SN ZTF24aahgqwk in NGC 3443

Observation Notes

Typically a session has 60 30-second exposures in each of g' and r'.

Observation Log (https://brianhill.github.io/supernova-observation/ZTF24aahgwk_observation_log.html)

In [6]:

```
import os
import numpy as np
from astropy import units as u
from astropy.nddata import CCDData
# from astropy.io import fits
from ccdproc import ImageFileCollection, combine
# Combiner, combine, subtract_dark, flat_correct
import matplotlib.pyplot as plt
%matplotlib inline

# filters

filters = ['g', 'r']
filter_full_names = ["Sloan g'", "Sloan r'"]
```

Combine the Calibration Images into Masters

Calibration Images

The calibration images are in ~/2024 Sessions/2024-04-12/. In turn, ~/2024 Sessions is actually a soft link to /Volumes/Astronomy Data/2024 Sessions/2024 Sessions.

In [5]:

```
# calibration directory

calibration_date = '2024-04-12'

calibration_directory = os.path.join(os.path.expanduser('~'), '202

# subdirectory for the 30-second darks

dark_directory = os.path.join(calibration_directory, 'dark')

# subdirectories for the 0.1-second g and r flats
```

```

flat_directories_by_filter = {filter:os.path.join(calibration_dire
                              for filter in filters}

# subdirectory for the biases (TheSky Professional Edition may inc

bias_directory = os.path.join(calibration_directory, 'bias')

# Trimmed image reader utility (needed because our images have fou

def delete_last_columns(arr, columns_to_delete):
    column_count = np.shape(arr)[1]
    del_arr = np.delete(arr, slice(column_count - columns_to_delet
    return del_arr

def trimmed_image_reader(file):
    img = CCDData.read(file, unit=u.adu)
    data = img.data
    trimmed_data = delete_last_columns(data, 4)
    img.data = trimmed_data
    return img

# darks

dark_files = ImageFileCollection(dark_directory).files_filtered(in
darks = [trimmed_image_reader(file) for file in dark_files]

# flats by filter

flat_files_by_filter = {filter:ImageFileCollection(flat_directory)
                        for filter, flat_directory in flat_directo
flats_by_filter = {filter:[trimmed_image_reader(file) for file in
                   for filter, flat_files in flat_files_by_filter.

# biases

bias_files = ImageFileCollection(bias_directory).files_filtered(in
biases = [trimmed_image_reader(file) for file in bias_files]

# Combine darks, flats, and biases

method = 'median'  # alternatively, the method can be 'average'

master_dark = combine(darks, method=method)
master_flat_by_filter = {filter:combine(flats, method=method)
                         for filter, flats in flats_by_filter.item
master_bias = combine(biases, method=method)
```

...

Load and Align Lights

The lights we are examining are in ~/2024 Sessions/2024-04-17/.

In [9]:

```
observation_date = '2024-04-17'

observation_directory = os.path.join(os.path.expanduser('~'), '202

# subdirectories for the 30-second g and r lights

light_directories_by_filter = {filter:os.path.join(observation_dir

# lights by filter

light_files_by_filter = {filter:ImageFileCollection(light_director
                     for filter, light_directory in light_directories_by
lights_by_filter = [[trimmed_image_reader(file) for file in light_
                        for filter, light_files in light_files_by_filt

# the aligned directories are written to not read from

aligned_directories_by_filter = {filter:os.path.join(light_directo
                                       for filter, light_directory in li

for aligned_directory in aligned_directories_by_filter.values():
    if not os.path.exists(aligned_directory):
        os.makedirs(aligned_directory)

```

...

In [ ]: