Lab #3 – Astronomical Spectroscopy

AST326 – December 06, 2018

Ayush Pandhi (ayush.pandhi@mail.utoronto.ca)

Group E: Ayush Pandhi, Hansen Jiang and Keslen Murdock

1 Abstract

A USB 2000 spectrometer is used to gather spectra of an incandescent lamp, fluorescent strip, gas discharge lamp and sunlight. A wavelength calibration is done using Ocean Optics Calibration Guide that was provided with the experiment; the wavelength solution is fit to a quadratic solution. The spectra are then analyzed for continuum shapes (i.e. peak wavelength and general shape) as well an absorption effects corresponding to specific elements. The SBIG instrument is then used on a 16" telescope to measure the astronomical sources: Moon, Mars, Vega, Capella and Enif. A neon spectrum is again used to calibrate this data as well and darks and flat fields are also used to calibrate the data correctly to remove any external noise. These spectra are also analyzed for chemical compositions and physical conditions.

2 Introduction

Fundamentally, spectroscopy is the study of the interactions between matter and electromagnetic radiation. Specifically, one can observe a continuous spectrum, emission lines and absorption lines by using a prism or diffraction grating. Continuous spectra are produced by objects that radiate heat under ideal conditions (blackbody assumption); absorption lines are often found in these continuous spectra and originate from certain particles absorbing specific wavelengths of light. An emission spectrum on the other hand occurs when a hot gas emits light at certain wavelengths (corresponding to the particles in the gas). In the context of astronomical studies, this allows researchers to measure the spectrum of electromagnetic radiation from various astronomical sources, as well as obtain information about gas clouds and atmospheric conditions between the source and observer. Specifically, the continuum shape and peak wavelength of the spectrum is related to the temperature as seen from Planck's law of black-body radiation:

$$E(\lambda, T) = \frac{2hc^2}{\lambda^5}\left(\frac{1}{e^{\frac{hc}{\lambda kT}} - 1}\right) \tag{1}$$

$$\lambda_{peak} T = 2.898 \ x \ 10^{-3} \ \ (mK) \tag{2}$$

Where, h is Planck's constant, k is the Boltzmann constant, c is the speed of light, λ is the wavelength (m) and T is the temperature (K). The chemical composition of the source and the physical conditions can also be discerned through analysis of the line emission/absorption. Furthermore, by comparing the absorption lines to the line emission of well-known elements, the velocity of the source can be found through a Doppler Shift.

The following experiment uses observations from both a USB 2000 Spectrometer and the SBIG (Santa Barbara Instrument Group) Self Guided Spectrograph for various different sources. Theoretically, this data provides a basis from which the wavelength-pixel relationship can be calibrated and thus allows for the analysis of astronomical sources in comparison to the lab data. The data is analyzed as a spectrum of electromagnetic radiation in the visible band (roughly 350–750nm).

3 Observation and Data

A USB 2000 spectrometer was used to obtain the spectra of an incandescent lamp, fluorescent strip light, a gas discharge lamp and sunlight. This device is a simple optical instrument that utilizes a diffraction grating and a one-dimensional CCD detector array (1 x 2048 pixels). The spectrograph is based on a Czerny-Turner optical design and achieves a spectral resolution of approximately 0.6nm between 370 – 700nm wavelengths. The device is connected via USB to a PC and the Ocean Optics' SpectraSuite software is used to operate the spectrometer through a serial interface. Figure 1 is taken from the instrument manual for the USB 2000 spectrometer and depicts the interior optical layout:

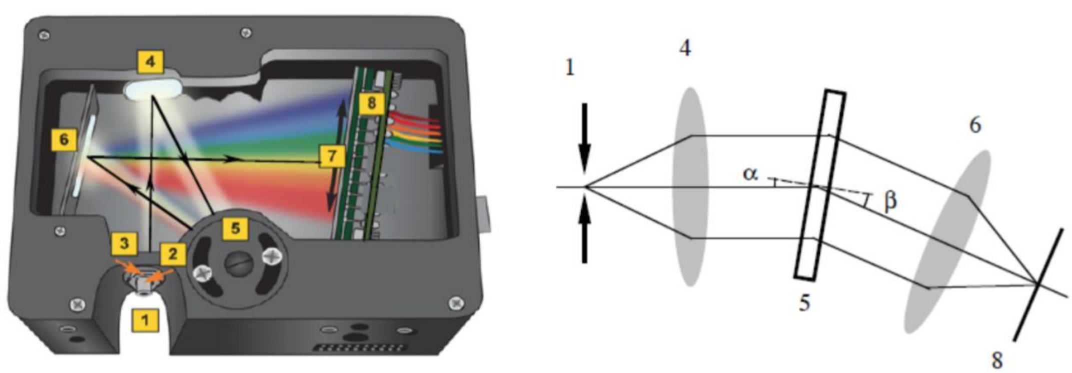

Figure 1: *The interior optical layout of a USB 2000 spectrometer. The key optical components are labelled: entrance aperture (1), collimating mirror (4), diffraction grating (5), camera mirror (6) and detector array (8).*

The USB 2000 spectra were collected over an integration time of 10 milliseconds with no processing options (only raw counts were taken from the CCD) and were saved as tab-delimited ASCII text files. The following table provides a log of the observations:

Date	Time [EST]	Source	Integration Time	Conditions
2018/11/20	12:30 pm	Incandescent Lamp	10.0 ms	Lights off and blinds down
2018/11/20	12:36 pm	Fluorescent Strip	10.0 ms	Lights off and blinds down
2018/11/20	12:41 pm	Gas Discharge Lamp	10.0 ms	Lights off and blinds down
2018/11/20	12:47 pm	Sunlight	10.0 ms	Blinds open, clear skies

Table 1: *USB 2000 spectrometer observation summary for various sources and conditions.*

In addition, the SBIG Self-Guiding Spectrograph was utilized on the 16" campus telescope to record the spectra of various astronomical sources (Moon, Mars, Vega, Capella and Enif). The spectra for neon and dark calibration were also taken. This instrument has two available gratings: 150 lines per mm (4.3 Angstroms per pixel) and 600 lines per mm (1.0 Angstroms per pixel). There are also two slit width options: 18 microns and 72 microns. For the purposes of this experiment, the narrow slit and 150 lines per mm grating were selected, which gives a resolution of 10 Angstroms (or equivalently 1 nm). The following table provides a log of the observations:

Date	Time [EST]	Source	Exposure Time	Conditions
2018/11/22	09:12 pm	Moon	120.0 s	Cloudy
2018/11/22	09:20 pm	Mars	120.0 s	Cloudy
2018/11/22	09:27 pm	Vega	120.0 s	Cloudy
2018/11/22	09:35 pm	Capella	120.0 s	Cloudy
2018/11/22	09:42 pm	Enif	120.0 s	Cloudy
2018/11/22	09:46 pm	Darks (x3)	120.0 s	Cloudy
2018/11/22	09:58 pm	Neon Light	120.0 s	Cloudy

Table 2: *SBIG Self-Guiding Spectrograph observation summary for various sources and conditions.*

It should be noted that although measures were taken to remove excess sources of electromagnetic radiation when using the USB 2000 spectrometer (the room lights were turned off and blinds were pulled down), it is possible that there could be small excess noise in the data. However, this noise deemed to be negligible when analyzing the overall structure of the spectra.

4 Data Reduction and Methods

Firstly, the noise properties of the spectrograph are investigated by computing the mean and variance for each pixel in a time sequence. The relationship between variance, s_{ADU}^2, and mean pixel value, x_{ADU}, scales linearly as:

$$s_{ADU}^2 = s_0^2 + kx_{ADU} \tag{3}$$

Where s_0 and k represent the read noise and gain respectively. A focal point of this experiment involves calibrating the pixel to wavelength conversion of both the spectrometer and the SBIG instrument. This is achieved by computing the centroids of the neon spectrum observed by both instruments respectively and applying a linear least squares method with the expected wavelength of the peaks to determine a polynomial fit to the data sets. The theoretical wavelength for neon peaks is obtained from the Ocean Optics Calibration Guide (see appendix for details). In general, a least squares fit attempts to minimize the quantity:

$$\chi^2 = \sum_i [y_i - f(x_i)]^2 \tag{4}$$

Where y is the measured value (in this case it is the expected wavelength of the neon peaks) which depends on x (the centroids observed in the experimental neon spectrum) and $f(x)$ is a polynomial function to which the data is fit. For this experiment, a quadratic polynomial fit was used (i.e. $f(x) = ax^2 + bx + c$). The best values for the parameters a, b and c are thus found by simultaneously solving for when the partial derivatives of χ^2 with respect to each parameter is zero. This leads to an equation of the form:

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \Sigma x^4 & \Sigma x^3 & \Sigma x^2 \\ \Sigma x^3 & \Sigma x^2 & \Sigma x \\ \Sigma x^2 & \Sigma x & N \end{pmatrix}^{-1} \begin{pmatrix} \Sigma x^2 y \\ \Sigma xy \\ \Sigma y \end{pmatrix} \tag{5}$$

Here, N is the number of data points. Furthermore, the uncertainties for each of the above parameters can be derived from the standard deviation, which is given as:

$$\sigma^2 = \frac{1}{N-2} \sum_i [y_i - f(x_i)]^2 \tag{6}$$

Another step must be applied to the SBIG data after calibration however; the data must be corrected using darks and flat field calibration. Assuming that the flat field radiates like a black body with temperature equal to the colour temperature, the flux for each pixel can be computed as:

$$P_i = \frac{R_i - D_i}{L_i - D_i} B(v_i, T) \tag{7}$$

Where R_i is the raw signal from the source, D_i is the dark count, L_i is the flat field count and $B(v_i, T)$ is the Planck function, with $v_i = c/\lambda_i$:

$$B(v_i, T) = \frac{2hv^3}{c^2} \frac{1}{e^{\frac{hv}{kT}} - 1} \tag{8}$$

With these calibrations applied to the data, the spectra are than analyzed with respect to well-known element line emissions and physical characteristics of the astronomical source. One common example of this is the Hydrogen emission lines, which are defined by:

$$\frac{1}{\lambda} = R\left(\frac{1}{n_1^2} - \frac{1}{n_2^2}\right) \tag{9}$$

Where R is the Rydberg constant and n_1/n_2 are integer values corresponding to the principal quantum number for the orbitals occupied before and after the absorption/emission.

5 Data Analysis and Modeling

The detailed python code for generating all figures and data analysis in this section can be found in the Appendix. To start, the mean and variance for each pixel was computed from the time sequence. This allows for the computation of read noise and gain as per Equation 3. An example of a time series for pixel 1000 is given in Figure 2 below, the change between samples is due to noise and the AC power supply. However, it can be concluded that the variance is not dominated by external factors such as varying illumination.

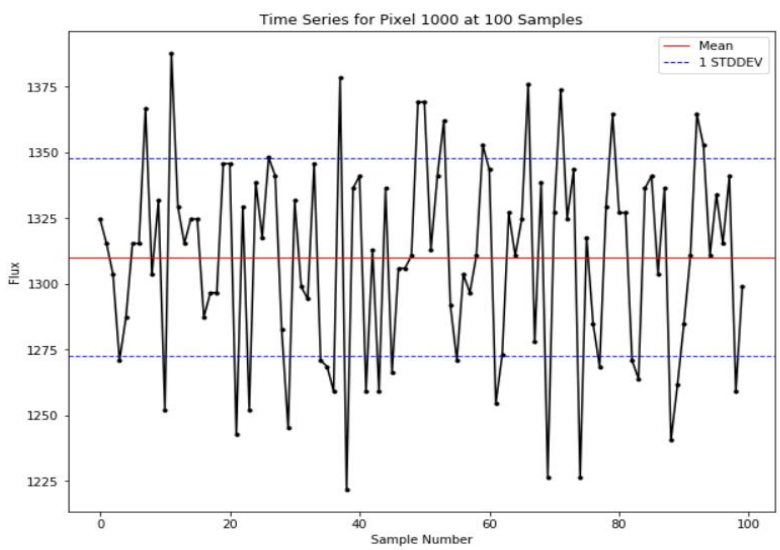

Figure 2: *Time Series for pixel 1000 at 100 Samples with mean and one standard deviation labelled.*

Using the USB 2000 spectrometer and SpectralSuite software, the spectra for an incandescent lamp, fluorescent strip, gas discharge lamp and sunlight were obtained. The raw data shows the flux in relation to the corresponding physical pixel that it was observed at (Figure 3).

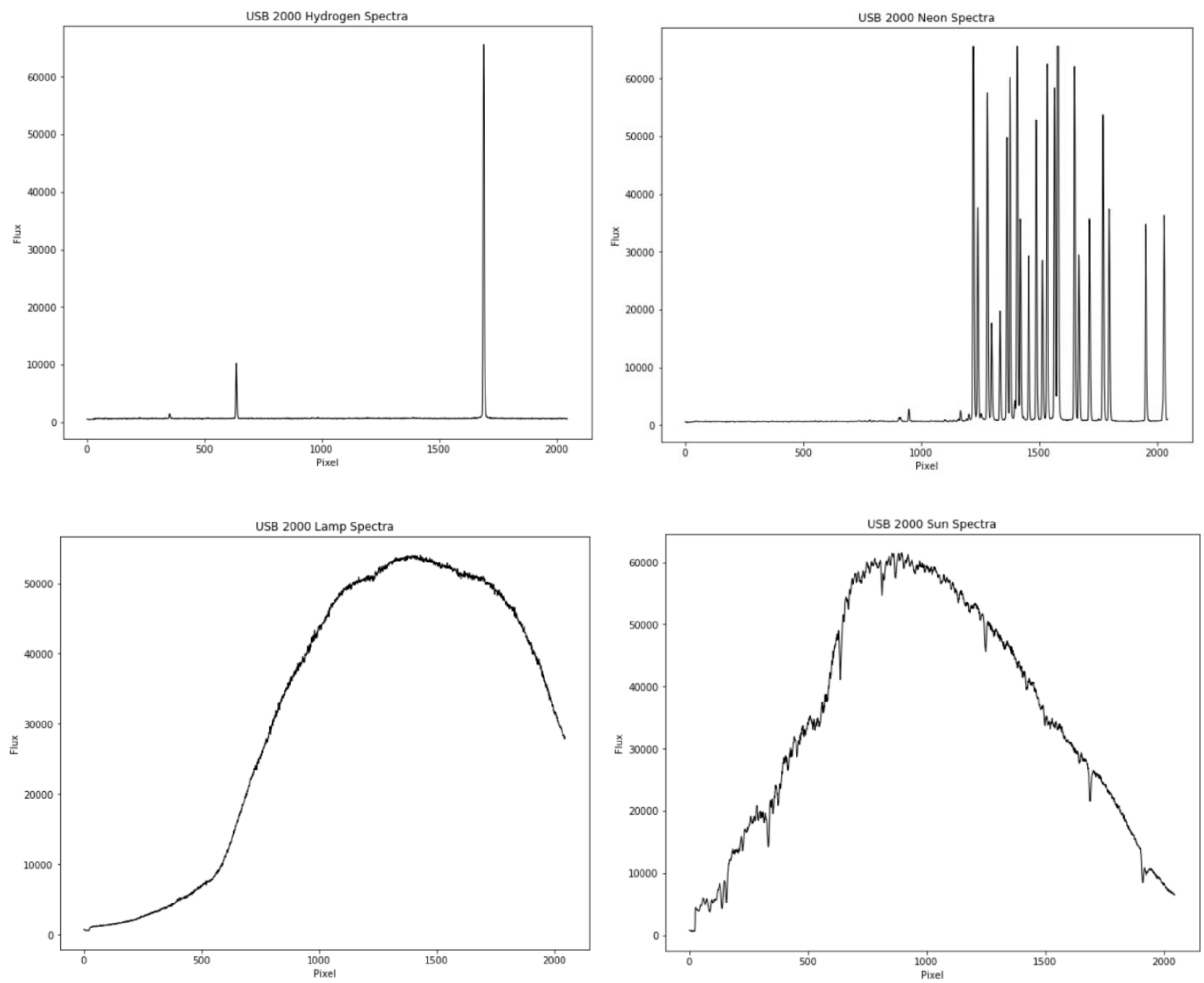

Figure 3: *All subplots are of Flux vs. Pixel Number from USB 2000 observations for different sources: (Top-Left) Gas Discharge Lamp, (Top-Right) Fluorescent strip, (Bottom-Left) Incandescent lamp and (Bottom-Right) Sunlight.*

The centroids for the neon spectrum's peaks were computed with a lower threshold of 10 000 on the Flux scale. The centroids corresponding to theoretical neon peaks from the Ocean Optics Calibration Guide were then used to estimate the parameters for a quadratic fit to calibrate the pixels (P) to wavelength (λ). The resulting polynomial fit for the USB 2000 spectrometer was approximately:

$$\lambda = (-1.66\, x\, 10^{-5})P^2 + (2.0\, x\, 10^{-1})P + (3.66\, x\, 10^{2}) \tag{10}$$

This wavelength solution was used to calibrate all four spectra seen in Figure 3 to their corresponding wavelength (nm). Figure 4 shows the centroids corresponding to neon emission lines that were used for the calibration and Figure 5 is the same plot as Figure 3 except with the wavelength solution applied and over-plotted with the theoretical Balmer series Hydrogen lines which are computed using Equation 9.

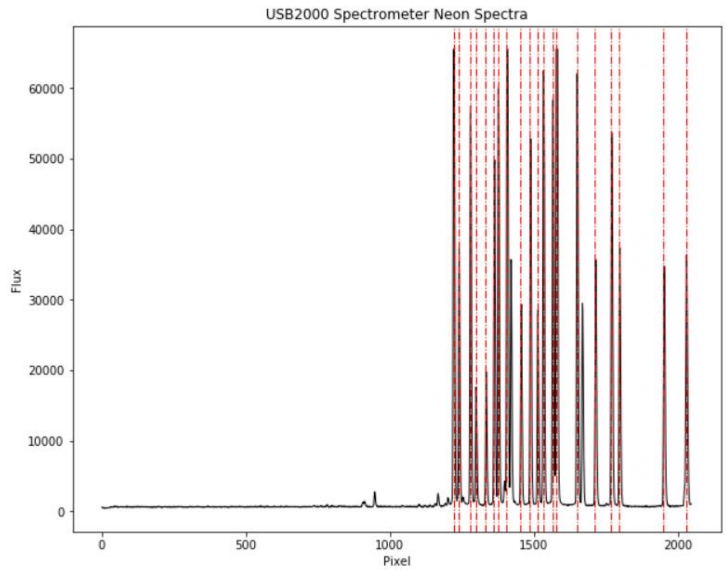

Figure 4: *USB 2000 Neon spectra as seen in figure 3, now over-plotted with centroid positions which correspond to neon line emissions from the Ocean Optics Calibration Guide (dotted red lines).*

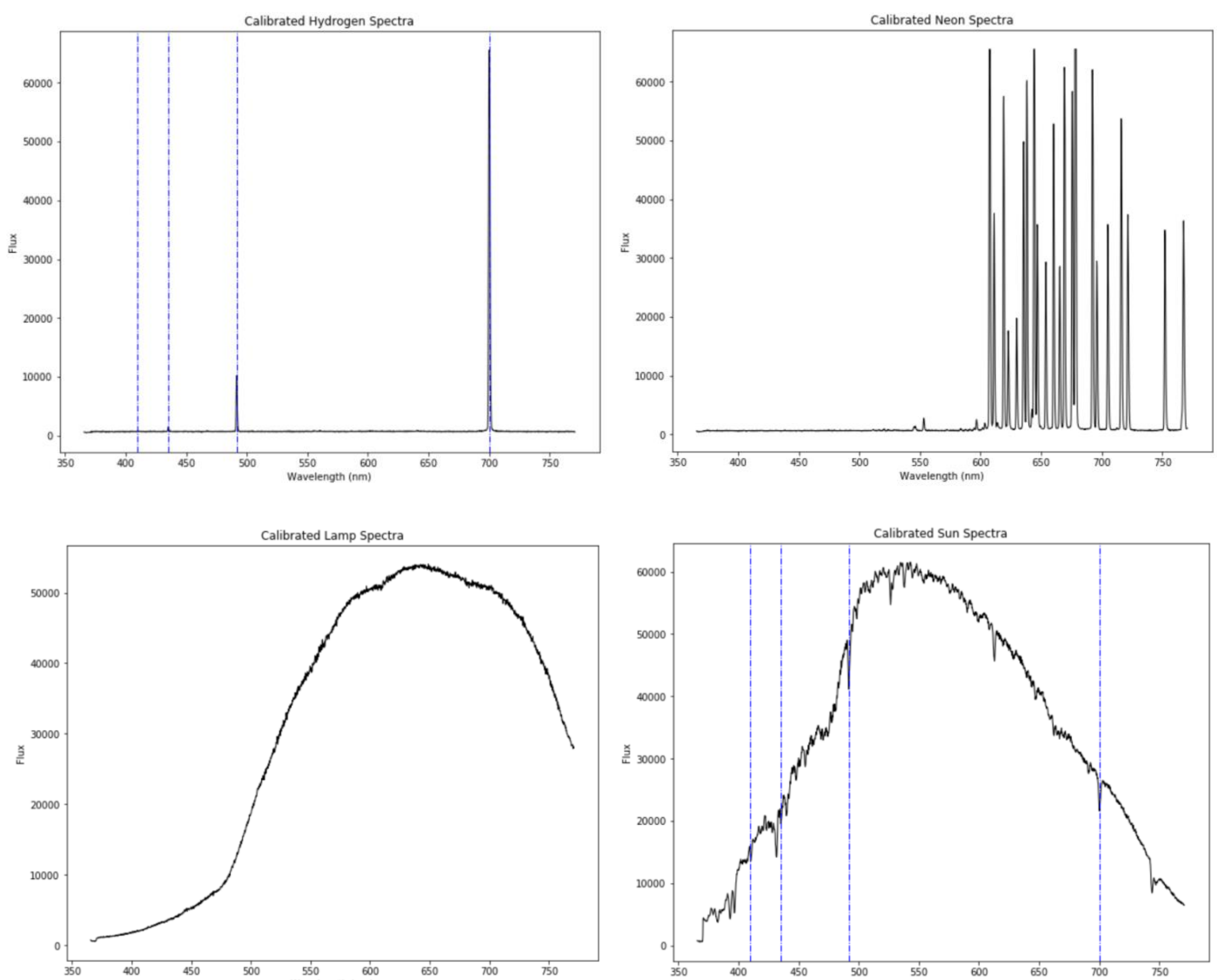

Figure 5: *Identical plot to Figure 3 but with the wavelength solution applied. Theoretical Balmer Hydrogen line emissions are over-plotted as dotted blue lines (correspond to Hydrogen alpha, beta, gamma and delta lines from right to left).*

The pixel to wavelength calibration for the SBIG instrument was done in an identical manner with the experimental centroids of its neon calibration spectra. The wavelength solution for the SBIG instrument was determined to be approximately:

$$\lambda = (-4.43\ x\ 10^{-5})P^2 + (4.43\ x\ 10^{-1})P + (3.85\ x\ 10^{2}) \tag{11}$$

A further calibration was applied to correct for the dark current as well as the flat field corresponding to the instrument. This calibration followed Equation 7 and was applied to every data set taken from this instrument. Figure 6 below shows the averaged darks and flat field used for this calibration.

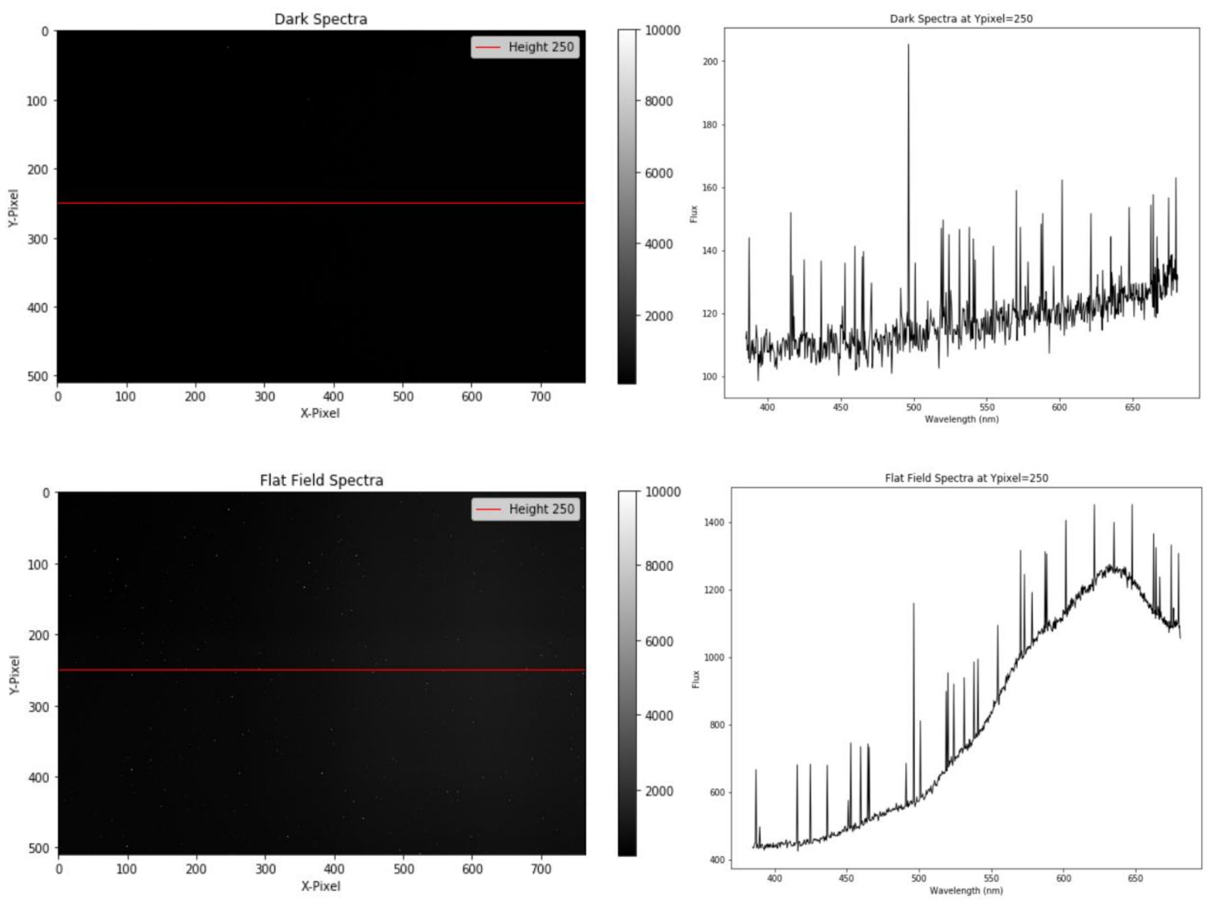

Figure 6: *(Top-Left) 2D pixel array plot with the colour bar representing flux for the averaged dark flats, (Top-Right) 1D spectra taken at Ypixel = 250 from the averaged dark spectra. (Bottom-Left) 2D pixel array plot of the flat field, (Bottom-Right) 1D spectra taken at Ypixel = 250 from the flat field.*

With these calibrations, the data for the observed astronomical sources was plotted (Moon, Mars, Vega, Capella and Enif) as both 2D and 1D spectra (Figures 7-11). These plots can be found in the appendix.

6 Discussion

In Figures 3 and 5 it can be seen that the Fluorescent strip and Gas Discharge lamp both produce a line emission corresponding to neon and hydrogen lines respectively. The incandescent light bulb acts as a blackbody and produces a smooth spectrum which does not have any clear absorption lines, which agrees with the expected result as there is physically very little in terms of an atmosphere or gas between the instrument and blub. Sunlight produces a spectrum with peak wavelength of approximately 520nm. This corresponds to a surface temperature of about 5573K which is fairly close to the known surface temperature of the sun (~5778K). Additionally, the hydrogen lines correlate to dips in the spectrum, which indicates that there exists Hydrogen near the surface of the Sun which is causing these absorption lines. The absorption line at roughly 590nm corresponds to Helium which, although to a lesser extent than Hydrogen, is also present in the Sun's atmosphere.

The spectrum of the Moon in Figure 7 is very similar to the Sun's spectrum, except its peak wavelength is shifted to approximately 580 nm (corresponds to a blackbody temperature of about 4996K). This is to be expected because the main source of flux from the Moon is due to it reflecting the Sun's light. Since the moon isn't a perfect reflecting surface; it must absorb and re-emit the Sun's radiation, causing the shift to a higher peak wavelength (and thus a lower blackbody temperature). The absorption lines also do correlate strongly with the hydrogen alpha and beta wavelengths (656nm and 486.1nm) and to a lesser extent with hydrogen gamma and delta (434nm and 410nm). This implies again, that the light from the moon is very similar to that from the Sun and that the Moon does not have a significant atmosphere as it did not create additional absorption effects.

In Figure 8, it can be seen that Mars has a peak wavelength of approximately 640nm (corresponds to a blackbody temperature of about 4527K). Again, there is some resemblance in the overall shape of the spectrum to that of the Moon and Sun. This is due to the fact that a lot of light observed from Mars is reflected from the Sun. Thus, as was the case of the Moon, the spectrum is slightly shifted to a higher peak wavelength and still somewhat retains the absorptions troughs corresponding to Hydrogen lines. Mars' atmosphere is primarily composed of carbon dioxide but the absorption lines for CO_2 cannot be seen clearly in the visible band.

Vega from Figure 9, seems to be peaked at approximately 575nm (corresponds to a blackbody temperature of about 5039K). This star is known to be a type A0 star (much hotter than the Sun which is a G2) but the experimental data suggests it is colder than the Sun. This is thought to be as a result of imperfect conditions at the time of observation as well as due to the 2D spectra not lining up perfectly horizontally, which could cause a shift in the flux data. The absorption spectrum is again dominated by Hydrogen emission lines, especially alpha, beta and gamma. Thus it can be inferred that there is a lot of Hydrogen near the surface of this star.

Capella (Figure 10) is thought to be a quadrupole star system; consisting of two binary systems. The binary system of Capella Aa and Ab is primary focal point when analyzing in the visual band as the other binary system consists of two very cool red dwarf stars. The peak wavelength seems to be at approximately 570nm (corresponding to a blackbody temperature of about 5083K), which agrees with the idea that the binary stars are both yellow giants (spectral type G3 and K0 respectively). There are clear hydrogen absorption lines; however they seem to be slightly shifted to a higher wavelength. This could imply that there is a redshift affect, meaning that Capella is moving away from Earth at a fairly significant velocity.

The final spectrum is of Enif (Figure 11), who's peak wavelength is approximately 650nm (corresponding to a blackbody temperature of about 4458K). This roughly agrees with the expected result, as Enif is a type K2 supergiant star. The spectrum is fairly noisy and it is difficult to discern any clear absorption lines; although hydrogen alpha, and to a lesser degree hydrogen beta, does line up with a fairly significant trough.

Throughout the SBIG observations, there was consistently an absorption trough at roughly 670nm. This corresponds to an H_2O absorption line and is seem in most spectra because it originates in Earth's atmosphere. Since this data was taken from a ground based telescope, Earth's atmosphere will also create absorption effects on the overall spectrum.

9 Appendix

9.1 SBIG 2D and 1D Spectra of Astronomical Sources

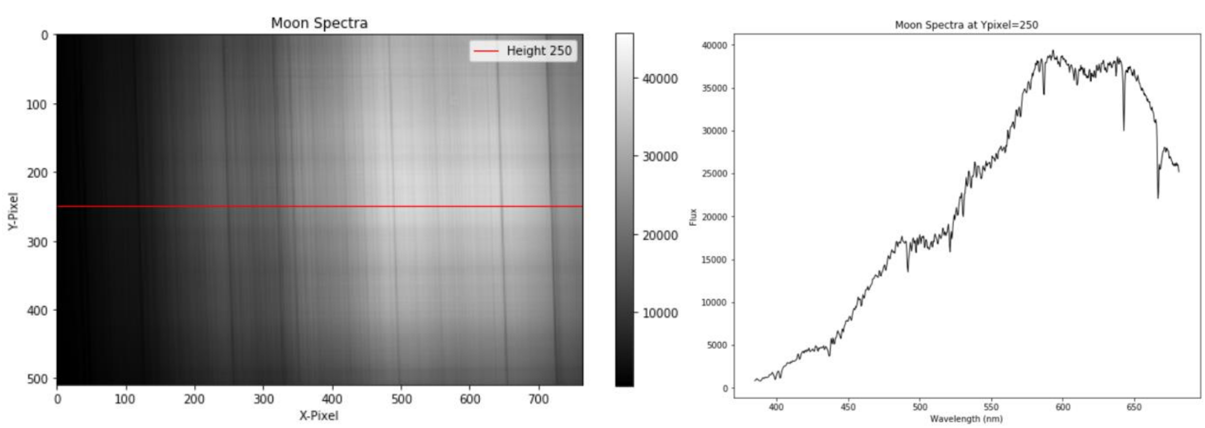

Figure 7: *A calibrated 2D and 1D spectra of the Moon.*

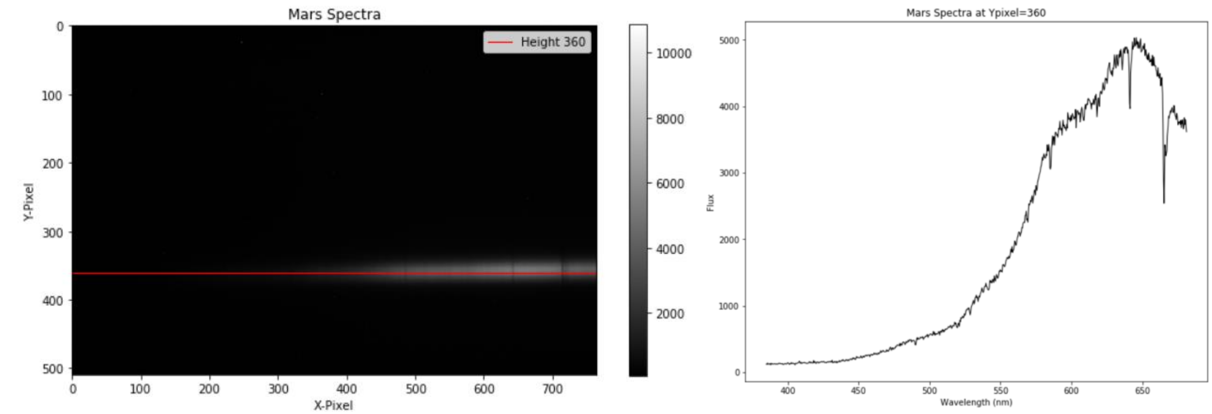

Figure 8: *A calibrated 2D and 1D spectra of Mars.*

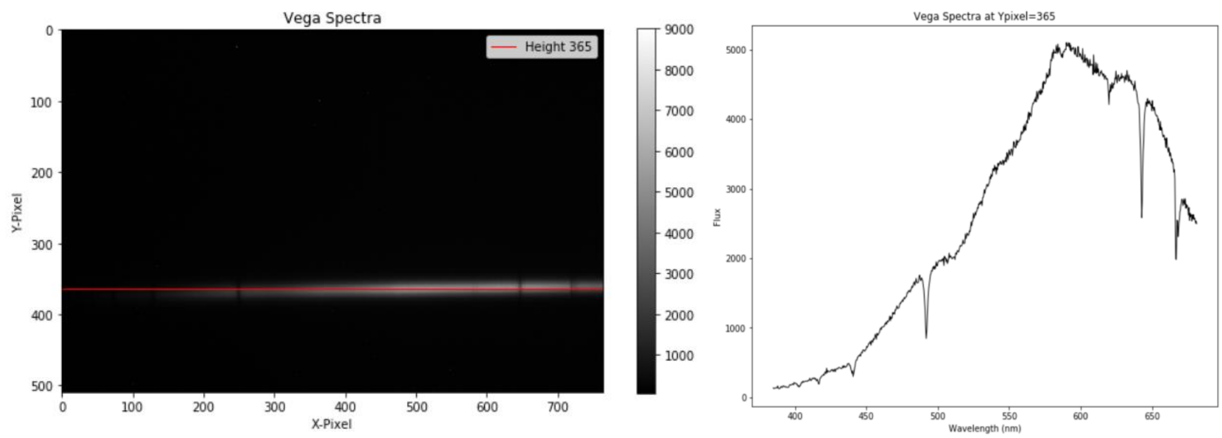

Figure 9: *A calibrated 2D and 1D spectra of Vega.*

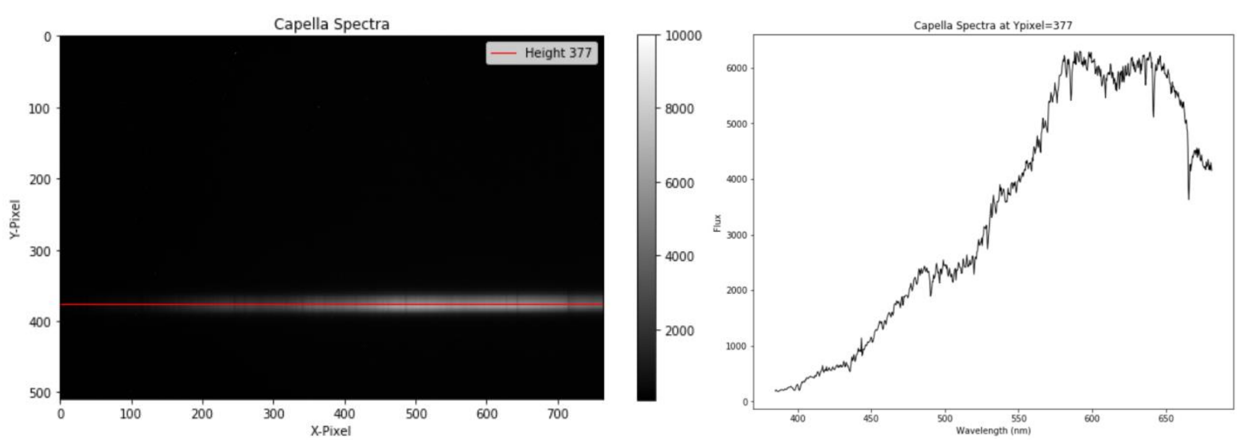

Figure 10: *A calibrated 2D and 1D spectra of Capella.*

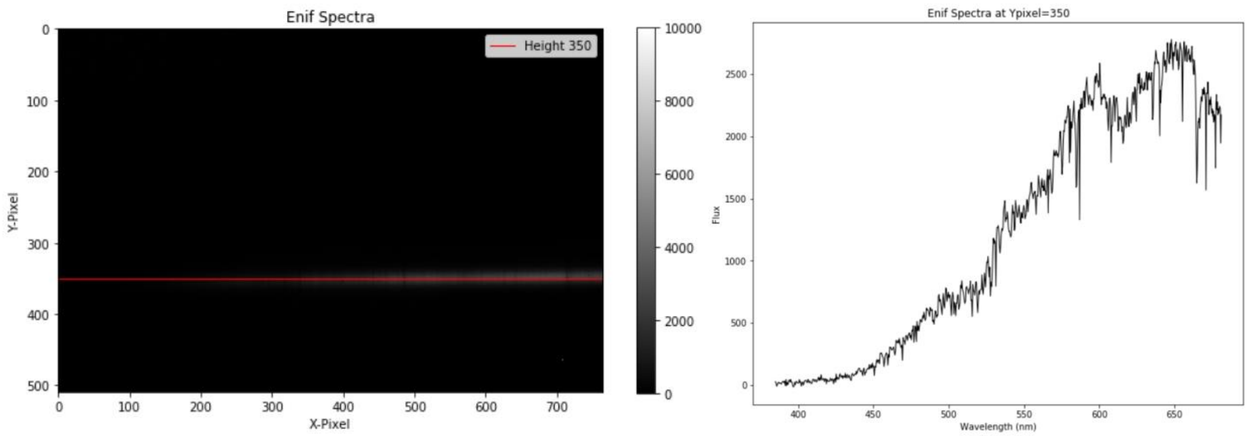

Figure 11: *A calibrated 2D and 1D spectra of Enif.*

9.2 Python Code for Figure 2

```
#Setting up a list to hold the loaded data

pixel = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99]

flux = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99]


#Loading all of the data in the 'to_combine' folder using a loop

for i in range(100):

  if i <= 9:

    pixel[i], flux[i] = np.genfromtxt('data2/gfdjgtm0000' + str(i) + '.txt', skip_header=17, skip_footer=1, unpack=True)

  else:

    pixel[i], flux[i] = np.genfromtxt('data2/gfdjgtm000' + str(i) + '.txt', skip_header=17, skip_footer=1, unpack=True)


#1000th pixel sample

P1000 = np.empty(100)
```

```
for i in range(100):
    P1000[i] = flux[i][1000]


#Plotting 1000th pixel time series for 100 samples
plt.figure(figsize=(10,8))
plt.plot(P1000, 'k.-')
plt.axhline(y=np.mean(P1000), color='r', label='Mean', linewidth=1)
plt.axhline(y=np.mean(P1000) - np.std(P1000), color='b', ls='--', linewidth=1, label='1 STDDEV')
plt.axhline(y=np.mean(P1000) + np.std(P1000), color='b', ls='--', linewidth=1)
plt.title('Time Series for Pixel 1000 at 100 Samples')
plt.xlabel('Sample Number')
plt.ylabel('Flux')
plt.legend()
plt.show()
```

9.3 Python Code for Finding Centroids

```
#Loading lab data
hydrogen_pixel, hydrogen_flux = np.genfromtxt('hydrogen.txt', skip_header=17, skip_footer=1, unpack=True)
neon_pixel, neon_flux = np.genfromtxt('neon.txt', skip_header=17, skip_footer=1, unpack=True)
lamp_pixel, lamp_flux = np.genfromtxt('lamp.txt', skip_header=17, skip_footer=1, unpack=True)
sun_pixel, sun_flux = np.genfromtxt('sunny.txt', skip_header=17, skip_footer=1, unpack=True)


#Creating an array of pixels numbers that are much higher than the mean to identify emission lines
p1 = []
for i in range(len(neon_flux)):
    if neon_flux[i] > 10000:
```

```
        p1.append(neon_pixel[i])


#Putting it into an array

p1_array = np.empty(len(p1))

for i in range(len(p1_array)):

   p1_array[i] = p1[i]

print(p1_array)
```

#From here manually compute the mean of each grouping of pixels in p1 and use that as the centroids of the peaks. Remove any peaks that do not correspond to Ocean Optics calibration data.

```
centroids = np.array([1221, 1239, 1279, 1298, 1333, 1362, 1376, 1406, 1454, 1485, 1487, 1512, 1532, 1565, 1578, 1649, 1713, 1769, 1796, 1951, 2028], dtype='int64')
```

9.4 Python Code for Figures 3, 4 and 5

```
#Plotting Hydrogen Spectra

plt.figure(figsize=(10,8))

plt.plot(hydrogen_pixel, hydrogen_flux, 'k', linewidth=1)

plt.title('USB 2000 Hydrogen Spectra')

plt.xlabel('Pixel')

plt.ylabel('Flux')

plt.show()


#Plotting Neon Spectra

plt.figure(figsize=(10,8))

plt.plot(neon_pixel, neon_flux, 'k', linewidth=1)

plt.title('USB 2000 Neon Spectra')

plt.xlabel('Pixel')
```

```
plt.ylabel('Flux')
plt.show()

#Plotting Lamp Spectra
plt.figure(figsize=(10,8))
plt.plot(lamp_pixel, lamp_flux, 'k', linewidth=1)
plt.title('USB 2000 Lamp Spectra')
plt.xlabel('Pixel')
plt.ylabel('Flux')
plt.show()

#Plotting Sun Spectra
plt.figure(figsize=(10,8))
plt.plot(sun_pixel, sun_flux, 'k', linewidth=1)
plt.title('USB 2000 Sun Spectra')
plt.xlabel('Pixel')
plt.ylabel('Flux')
plt.show()

#Plotting Neon Spectra with centroids used for calibration
plt.figure(figsize=(10,8))
plt.plot(neon_pixel, neon_flux, 'k', linewidth=1)
for i in centroids:
    plt.axvline(x=i, color='r', ls='-.', linewidth=1)
plt.title('USB2000 Spectrometer Neon Spectra')
plt.xlabel('Pixel')
plt.ylabel('Flux')
```

```
plt.show()


#Converting pixel to wavelength for USB2000

hydrogen_wavelength = ((-1.65784670*(10**-5)))*hydrogen_pixel**2 + (0.199775847)*hydrogen_pixel + 365.690737

neon_wavelength = ((-1.65784670*(10**-5)))*neon_pixel**2 + (0.199775847)*neon_pixel + 365.690737

lamp_wavelength = ((-1.65784670*(10**-5)))*lamp_pixel**2 + (0.199775847)*lamp_pixel + 365.690737

sun_wavelength = ((-1.65784670*(10**-5)))*sun_pixel**2 + (0.199775847)*sun_pixel + 365.690737


#Plotting Hydrogen Spectra

plt.figure(figsize=(10,8))

plt.plot(hydrogen_wavelength, hydrogen_flux, 'k', linewidth=1)

plt.title('Calibrated Hydrogen Spectra')

plt.axvline(x=410.174, color='b', ls='-.', linewidth=1)

plt.axvline(x=434.0462, color='b', ls='-.', linewidth=1)

plt.axvline(x=486.13615, color='b', ls='-.', linewidth=1)

plt.axvline(x=656.45377, color='b', ls='-.', linewidth=1)

plt.xlabel('Wavelength (nm)')

plt.ylabel('Flux')

plt.show()


#Plotting Neon Spectra

plt.figure(figsize=(10,8))

plt.plot(neon_wavelength, neon_flux, 'k', linewidth=1)

plt.title('Calibrated Neon Spectra')

plt.xlabel('Wavelength (nm)')
```

```
plt.ylabel('Flux')
plt.show()

#Plotting Lamp Spectra
plt.figure(figsize=(10,8))
plt.plot(lamp_wavelength, lamp_flux, 'k', linewidth=1)
plt.title('Calibrated Lamp Spectra')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Flux')
plt.show()

#Plotting Sun Spectra
plt.figure(figsize=(10,8))
plt.plot(sun_wavelength, sun_flux, 'k', linewidth=1)
plt.axvline(x=410.174, color='b', ls='-.', linewidth=1)
plt.axvline(x=434.0462, color='b', ls='-.', linewidth=1)
plt.axvline(x=486.13615, color='b', ls='-.', linewidth=1)
plt.axvline(x=656.45377, color='b', ls='-.', linewidth=1)
plt.title('Calibrated Sun Spectra')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Flux')
plt.show()
```

9.5 Python Code for Figure 6 (Wavelength calibration not shown, identical to above)

```
#Loading Dark data
dark1 = fits.open('CCD Dark 1.fit')
dark2 = fits.open('CCD Dark 2.fit')
```

```
dark3 = fits.open('CCD Dark 3.fit')
dark1data = dark1[0].data
dark2data = dark2[0].data
dark3data = dark3[0].data

#Mean of dark data
darkdata = np.empty([510, 765])
for i in range(0, 510):
   for j in range(0, 765):
       darkdata[i, j] = (dark1data[i, j] + dark2data[i, j] + dark3data[i, j])/3

#Flipping data to correct it
darkdata = np.flip(darkdata, axis=1)

#Plotting 2D Dark data
plt.figure(figsize=(10,8))
plt.imshow(darkdata, cmap='gist_gray', label='Dark Spectra', vmax=10000)
plt.axhline(y=250, color='r', label='Height 250', linewidth=1)
plt.title('Dark Spectra')
plt.xlabel('X-Pixel')
plt.ylabel('Y-Pixel')
plt.legend()
plt.colorbar(shrink=0.69)
plt.show()

#Plotting 1D Dark data at 250 height
plt.figure(figsize=(10,8))
```

```
plt.plot(darkdata[250], 'k', linewidth=1)
plt.title('Dark Spectra')
plt.xlabel('Pixel')
plt.ylabel('Flux')
plt.show()

#Plotting 1D Dark data at 250 height in wavelength
plt.figure(figsize=(10,8))
plt.plot(wavelength_sbig, darkdata[250], 'k', linewidth=1)
plt.title('Dark Spectra at Ypixel=250')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Flux')
plt.show()

#Loading Flat Field data
flat1 = fits.open('Flat-003F900s.fit')
flat2 = fits.open('Flat-004F900s.fit')
flat3 = fits.open('Flat-005F900s.fit')
flat4 = fits.open('Flat-006F900s.fit')
flat5 = fits.open('Flat-007F900s.fit')
flat6 = fits.open('Flat-008F900s.fit')
flat1data = flat1[0].data
flat2data = flat2[0].data
flat3data = flat3[0].data
flat4data = flat4[0].data
flat5data = flat5[0].data
flat6data = flat6[0].data
```

```
#Mean of flat data
flatdata = np.empty([510, 765])
for i in range(0, 510):
    for j in range(0, 765):
        flatdata[i, j] = (flat1data[i, j] + flat2data[i, j] + flat3data[i, j] + flat4data[i, j] + flat5data[i, j] + flat6data[i, j])/6

#Flipping data to correct it
flatdata = np.flip(flatdata, axis=1)

#Plotting 2D Flat data
plt.figure(figsize=(10,8))
plt.imshow(flatdata, cmap='gist_gray', label='Flat Field Spectra', vmax=10000)
plt.axhline(y=250, color='r', label='Height 250', linewidth=1)
plt.title('Flat Field Spectra')
plt.xlabel('X-Pixel')
plt.ylabel('Y-Pixel')
plt.legend()
plt.colorbar(shrink=0.69)
plt.show()

#Plotting 1D Flat data at 250 height
plt.figure(figsize=(10,8))
plt.plot(flatdata[250], 'k', linewidth=1)
plt.title('Flat Field Spectra')
plt.xlabel('Pixel')
```

```
plt.ylabel('Flux')
plt.show()

#Plotting 1D Flat data at 250 height in wavelength
plt.figure(figsize=(10,8))
plt.plot(wavelength_sbig, flatdata[250], 'k', linewidth=1)
plt.title('Flat Field Spectra at Ypixel=250')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Flux')
plt.show()
```

9.6 Python Code for Figure 7 (Figures 8-11 done identically)

```
#Loading moon data
moondata_final = ((moondata - darkdata)/(flatdata - darkdata))*(np.mean(flatdata))

#Plotting 2D moon data
plt.figure(figsize=(10,8))
plt.imshow(moondata_final, cmap='gist_gray', label='moon Spectra', vmin=0, vmax=10000)
plt.axhline(y=350, color='r', label='Height 350', linewidth=1)
plt.title('moon Spectra')
plt.xlabel('X-Pixel')
plt.ylabel('Y-Pixel')
plt.legend()
plt.colorbar(shrink=0.69)
plt.show()

#Plotting 1D moon data at 250 height
```

```
plt.figure(figsize=(10,8))
plt.plot(moondata_final[350], 'k', linewidth=1)
plt.title('moon Spectra')
plt.xlabel('Pixel')
plt.ylabel('Flux')
plt.show()

#Plotting 1D moon data at 250 height in wavelength
plt.figure(figsize=(10,8))
plt.plot(wavelength_sbig, moondata_final[350], 'k', linewidth=1)
plt.axvline(x=410.174, color='b', ls='-.', linewidth=1)
plt.axvline(x=434.0462, color='b', ls='-.', linewidth=1)
plt.axvline(x=486.13615, color='b', ls='-.', linewidth=1)
plt.axvline(x=656.45377, color='b', ls='-.', linewidth=1)
plt.title('moon Spectra at Ypixel=350')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Flux')
plt.show()
```