MK CLASSIFICATION OF SPECTRA USING AN AUTOMATED CLASSIFICATION ALGORITHM

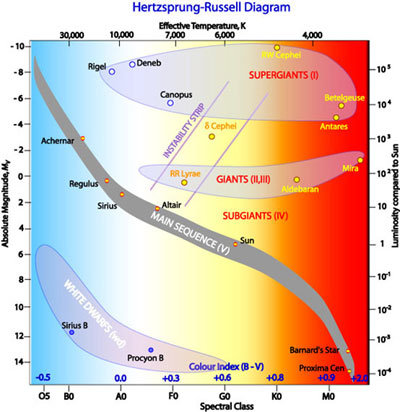
JULY 24, 2018

ROSIE FASULLO, DAVID WHELAN

ABSTRACT

This paper presents improved CCD data reduction and spectral extraction tools, nicknamed “Giraffe Butts”, and lays the groundwork for an automated spectral classification algorithm to be integrated into the program. Giraffe Butts is based off the data reduction and spectral classification tool written by D. G. Whelan to aid in the classification of spectra obtained with the 24” telescope spectrograph at the Adams Observatory, Austin College. Throughout the summer of 2018, optimizations were made to the data reduction algorithms that increased the user’s spectral typing efficiency by 57 %. The program can now perform automated wavelength calibration and will soon be able to perform automated continuum rectification. The automated spectral typing program will be completed in the next phase of this project during the fall semester of the 2018-2019 academic year. We discuss several methods of automated spectral classification, the existing spectral classification programs, *MKCLASS* and *The Cannon*.

1. INTRODUCTION

 Stars are grouped according to two different criteria, spectral type and luminosity class. The spectral type corresponds to the relative color of the light emitted by a star. The color of a star is an indication of what the surface temperature of the star is; the bluer the color, the hotter the star (Gray and Corbally 2009). There are 7 different spectral types used by astronomers today and they are, from bluest to reddest, O, B, A, F, G, K, and M (Gray and Corbally 2009). For reference, our sun is a spectral type G star. Stars above this type, the K and M type stars, are referred to as late-type stars. Stars that are below this type, the O, B, A and F type stars, are referred to as early-type stars (Gray and Corbally 2009). Within each type there is a numeric subtype ranging from 0 to 9 with 0 corresponding to the bluest subtype and 9 corresponding to the reddest subtype (Gray and Corbally 2009). Our sun has a spectral type of G2, meaning it is on the hotter end of a mid-range star (Gray and Corbally 2009). Luminosity class corresponds to the brightness of the star, which is an indication of the size of the star (Gray and Corbally 2009). Stars range from dwarf stars with luminosity class V to supergiant stars with luminosity class Ia (Gray and Corbally 2009). Our sun is a luminosity class V star, which is the most common luminosity class of the stars astronomers have observed (Gray and Corbally 2009). The distribution of spectral types and luminosity classes can be seen in what is called a Hertzsprung-Russell (HR) Diagram in Figure 1 below. The current observing project at the Adams Observatory is the spectral classifications of O, B and A type stars, with the goal of better understanding their fundamental properties and types of chemical peculiarities.

*Figure 1: Hertzsprung-Russell Diagram. Effective temperature is listed on the top the graph with the corresponding spectral type along the bottom. The luminosity is shown on the left with the corresponding absolute magnitude of the star along the right. The higher the magnitude of brightness the dimmer the star (“Hertzsprung-Russel Diagram”).*

In addition to spectral types and luminosity classes, a notation of any spectral peculiarities is also included in spectral classifications. There are several different types of spectral peculiarities. One form of spectral peculiarity is either full or partial emission in the hydrogen lines. Emission normally indicates there is a shell or disk around the star that is emitting light as the light from the star passes through it (Gray and Corbally 2009). A Chemically peculiar star shows unusual metal lines in the spectrum. Many chemically peculiar stars show unusually weak or unusually strong helium lines, indicating a particular abundance or lack of helium in the star (Gray and Corbally 2009). This can make these stars tricky to classify because helium lines are used in classifying O and B type star spectra (Gray and Corbally 2009). Additionally, there may also be unusually strong absorption of other metallic lines, indicating an unusually large abundance of that metal. Another common peculiarity is a binary star system. Depending on the physical relationship between the stars, the binary nature of the system may not be photometrically visible. Thus, the spectrum of the system may be the only way to see the spectral absorption features from both stars (Gray and Corbally 2009). A composite spectrum is distinguished from a normal spectrum by the presence of two spectra; although this can be quite difficult if the two stars are physically similar (Gray and Corbally 2009). Nebular spectra are characterized by the broadening and shortening of the metallic lines. What distinguishes nebular spectra from normal spectra is that although the depth and width of the lines are different than the standard, the strength (or total area of the line) is the same (Gray and Corbally 2009). A nebular spectrum indicates that the star is rotating at an unusually high rate and the widening of the lines is due to the Doppler effect (Gray and Corbally 2009).

There are several different systems used to classify stars. The system we employ is the MK classification system developed by W. W. Morgan and P.C. Keenan (Morgan and Kennan 1973). The MK system classifies stars by comparing spectrum of the star in question to a set of spectral standards and giving the star the label of whichever spectral standard it most resembles (Gray and Corbally 2009). This comparison is done using only spectra and does not admit the use of any extra information about the star such as photometric data (Gray and Corbally 2009). In this way, the MK system classifies stars according to their most natural groupings. The system does allow for the interpolation of spectral classifications if a star appears to lie between two spectra (Gray and Corbally 2009). The MK system allows spectral standards of three different types per classification: anchor points, primary standards, and secondary standards (Gray and Corbally 2009). An anchor point is a spectral standard that has not changed classification since development of the MK classification system began (Gray and Corbally 2009). Primary standards fill in the gaps of the anchor points and provide a reference for the best-known spectrum for each spectral classification (Gray and Corbally 2009). Secondary standards are the best-known stars that are accessible from both the northern and southern hemisphere (Gray and Corbally 2009).

Although a few spectral classification algorithms have been created, the most reliable method is still classifying stars by eye. The MK classification system often relies on the astronomer’s intuition. Although computers are good at many things, they are unable to make decisions based on intuition and simulating intuition is very complex. Thus, it is usually easier and more accurate for spectral typing to be done by eye. However, given the number of stars in the sky and the time it takes to classify each star, an accurate and efficient spectral typing program would be hugely beneficial. What a team of astronomers could do over the course of several days, a reliable spectral typing program could do in a few minutes.

There has been an attempt at an expert classification program by R. Gary and C. Corbally. They developed a program called *MKCLASS* that can accurately classify normal and chemically peculiar stars within 0.6 of a spectral type and 0.5 of a luminosity class (Gray and Corbally 2014). *MKCLASS* uses a weighted least squares comparison combined with a detailed comparison of the metallic line strengths to determine the spectral classification (Gray and Corbally 2014). *MKCLASS* is very good at classifying stars that are normal or have chemical peculiarities (Gray and Corbally 2014). However, there are some drawbacks to the *MKCLASS* program. Unfortunately, when a spectrum has a peculiarity that is not expected by the program, it gives up at classifying the spectrum without an explanation as to why the classification failed (Gray and Corbally 2014). In addition, there is no way to numerically measure the accuracy of the spectral classification of the program (Gray and Corbally 2014). The program provides a file that lists the quality of the classification that ranges from “Very poor” to “Very good”, but this refers to the signal to noise ratio of the spectrum, not the accuracy of the classification (Gray 2015). While *MKCLASS* is a strong program there is certainly room for improvement.

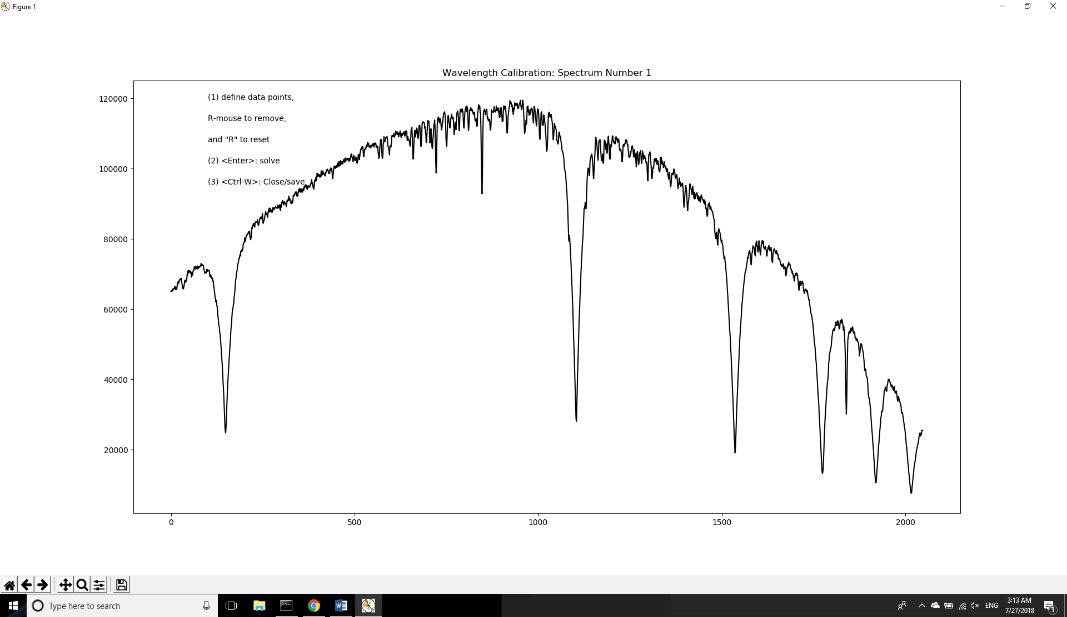
Although not a spectral classifying algorithm, *The Cannon* a program written by a group of astronomers at MIT, presents another perspective on the problem (Ness et al. 2015). *The Cannon* was developed to find a way to efficiently determine spectral labels of large sets of data (Ness et al. 2015). It is important to mention that the term “spectral labels” refers to any label applied to the spectra at all, not specifically spectral classification (Ness et al. 2015). *The Cannon* uses a data-driven approach to determine spectral labels (Ness et al. 2015). This method involves training a classifier on a reference set of spectra, and then using the trained classifier to determine the spectral labels of unknown spectra (Ness et al. 2015). The classifier learns the characteristics of the stars’ spectra with certain labels and looks for those characteristics in the spectra of yet unlabeled stars (Ness et al. 2015). This is a very computationally efficient method of classifying spectra because it does not involve storage of large lists of metallic line strengths or predetermined line characteristics (Ness et al. 2015). *The Cannon* is very effective at determining the spectral labels of a large set of spectra with a high degree of accuracy (Ness et al. 2015). The main weakness of this approach is that it requires a sufficiently large data set to train on for it to provide accurate classifications (Ness et al. 2015). The Cannon also requires extremely precise continuum rectification to be effective, which is extremely difficult to do (Ness et al. 2015). Although the program is not immediately applicable to determining classifications, there is a strong potential for adapting *The Cannon* method for use as an automated spectral classification algorithm.

So far, we have introduced spectral classification, the MK classification system and the current attempts at automated spectral classification. In the following section, the original spectral typing tool used by astronomers at the Adams Observatory will be introduced as well as the changes made over the course of the summer to improve the program. Statistics on the efficiency improvement of the program from the beginning of the program to the current version of the program will be presented in the Results section. Finally, a brief discussion of these statistics and a discussion of the future for the spectral classification algorithm and the program as a whole will be detailed in the Discussion section.

2. RESEARCH METHODS

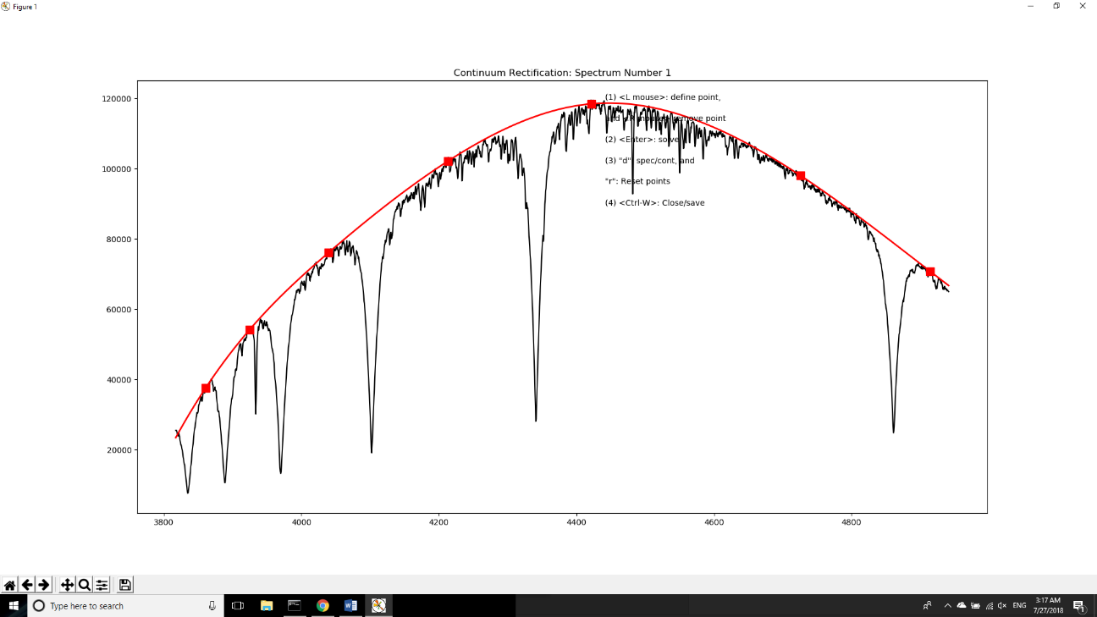
At the beginning of the summer, spectral extraction, data reduction and spectral classification at Austin College was done using a program written by Dr. David Whelan called ao\_redux. The program is written in the programming language Python and is just a tool to aid the astronomers working at the Adams Observatory with reducing spectroscopic data and classifying stars. Overall, the program is a bit cumbersome, but reasonably effective as a tool used for spectral classification.

The program begins by reducing the science images using the flatfields, biases and dark current images. Then, it shows the uncalibrated spectrum of the first star in the set to be wavelength calibrated in that run (Figure 2). The user then manually selects the hydrogen absorption lines for each star (Figure 3). The lines selected are then used for wavelength calibration. Wavelength calibration is the process of coverting the x-axis of the data from pixels to wavelength (Howell 2009). The program is provided with the known wavelength values of the hydrogren Balmer lines. Using the Balmer lines and the user selected lines, the x-axis is scaled to wavelength and the calibration is complete (Howell 2009).



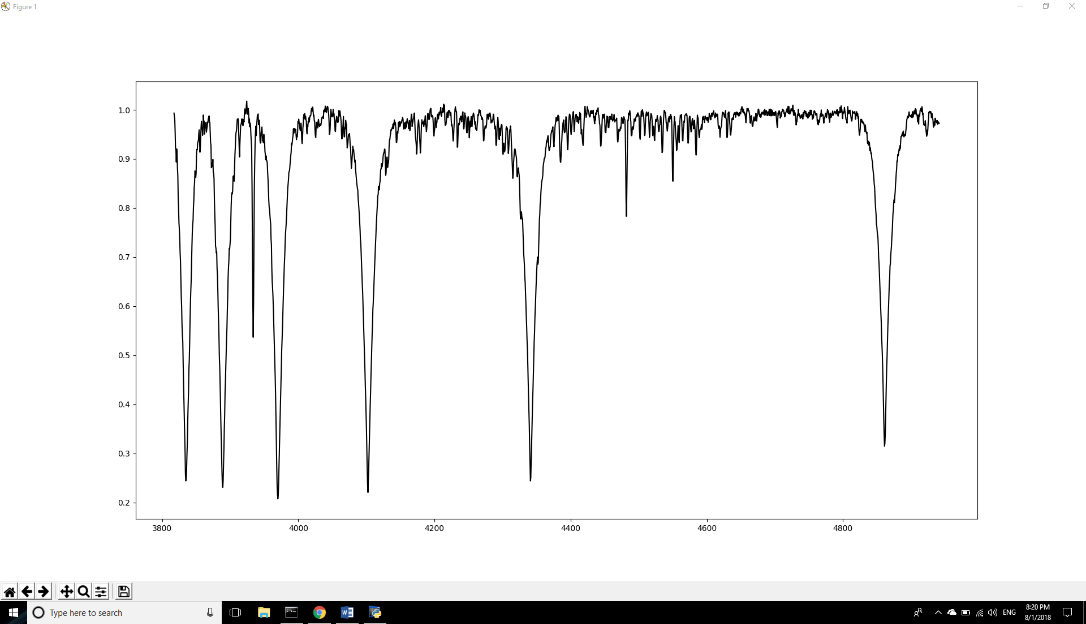
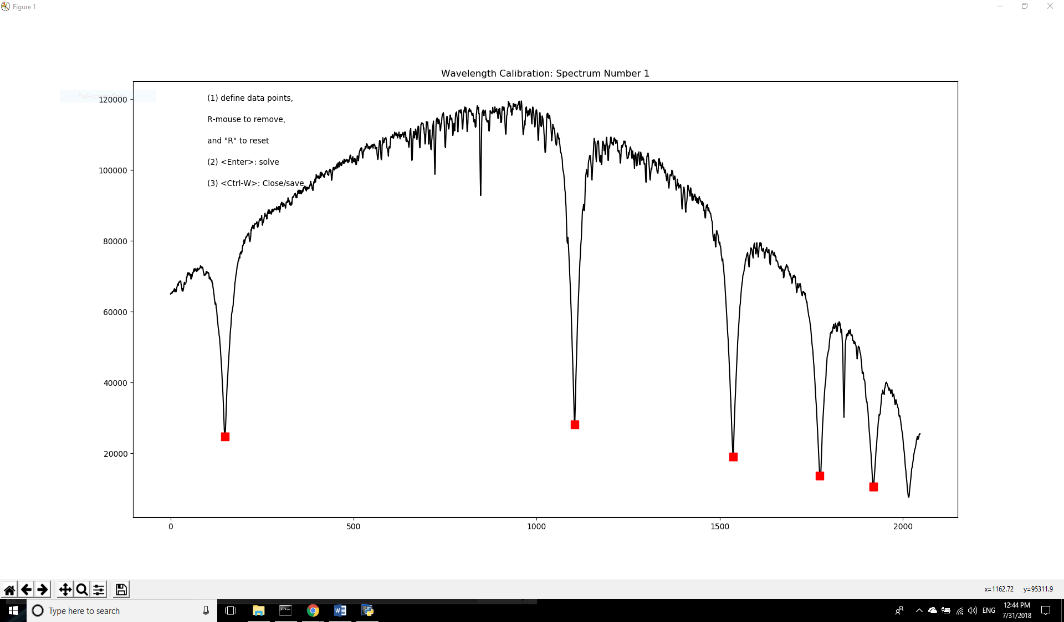
*Figure 2: Plot of the uncalibrated spectrum of a star as shown in ao\_redux.*

Once the user has completed wavelength calibration for each star, they begin continuum rectification. Continuum rectification is the process of taking a curve spectrum (Figure 4) and turning it into a flat spectrum (Figure 5). Continuum rectification involves the user selecting points representing each section of continuum manually and then the program solves for an approximate cubic spline fit to the curve created by the points (Figure 4). The curve will become the baseline of the rectified spectrum; thus, the curve needs to be smooth. The smoother the curve the straighter the rectified continuum. Once the user has gotten the fit to be as smooth as possible, they can view the rectified spectrum (Figure 5). If the spectrum is not rectified to their satisfaction, they must undo the rectification and start over. Once all the spectra in the set have been rectified, the user can continue to the spectral classification.



*Figure 4: Plot of the unrectified spectrum of the star with the cubic spline over-plotted. Usually one to two points per section of continuum are selected to create the cubic spline.*

*Figure 3: Plot of the uncalibrated spectrum of a star as shown in ao\_redux with the points selected for wavelength calibration.*

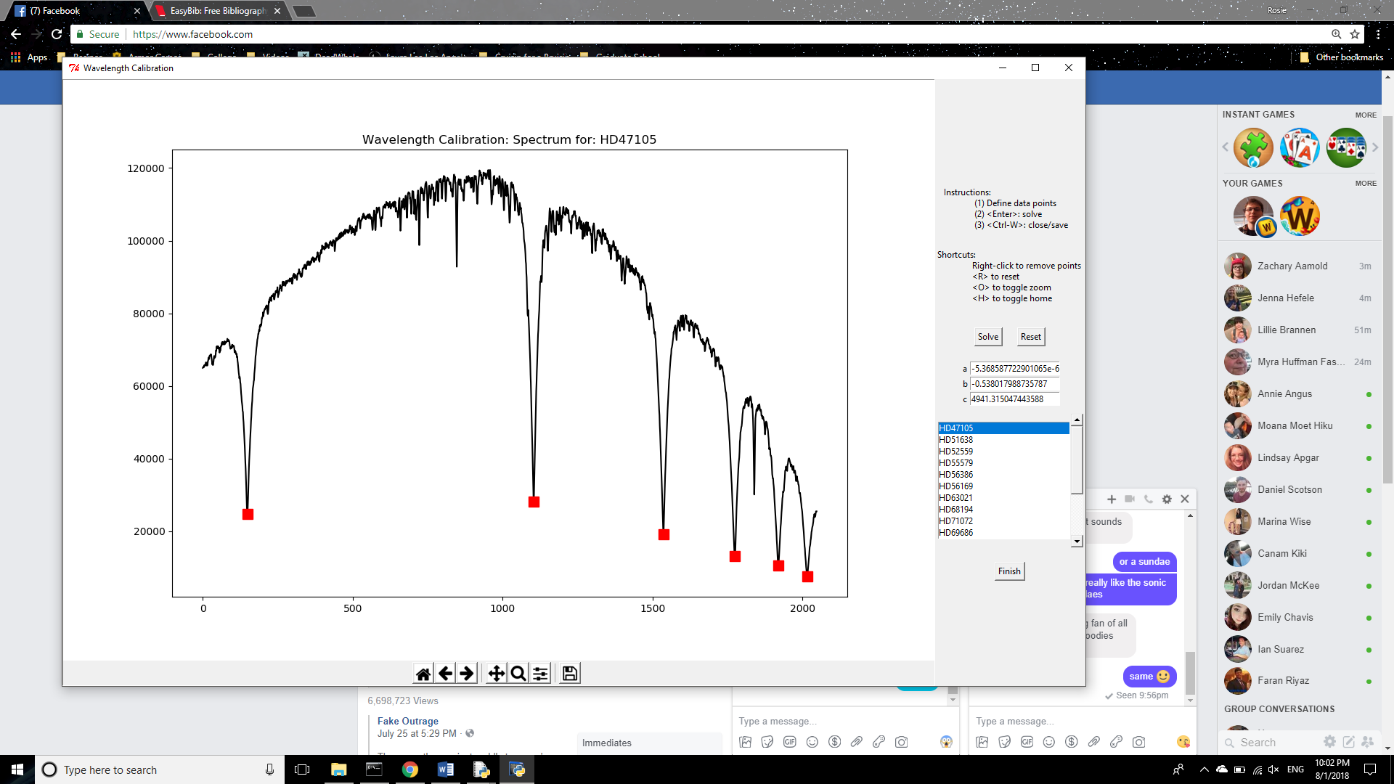


*Figure 5: Plot of the rectified spectrum of the star.*

During the classification phase of the program, the current spectral standard is plotted over the current unclassified spectrum in red, so the user can perform a direct visual comparison of the spectra. Spectral classification is completed using the MK classification system. We first look at the broad shape of the spectrum; depending on the shape of the continuum and the depth of the hydrogen lines, we make a general guess about whether the star belongs in the O, B, or A spectral types. O type stars are defined as having lines of ionized helium (He II) in their spectrum coupled with weak hydrogen lines compared to B and A type stars. For B type stars, the hydrogen lines gradually deepen and widen from B0 through B9 and the neutral helium lines (He I) become weaker. A type stars are distinguished from B type stars in that they are devoid of any helium lines but have increasingly strong iron and titanium lines. The hydrogen lines for A-type stars reach a maximum around A2 and gradually decrease through A9 (Gray and Corbally 2009). Once we have chosen the general type we think the spectrum fits best, we look analyze the hydrogen lines. The depth of the lines indicates the early or lateness of the subtype of the star and the width of the lines indicates the luminosity class. Once we believe we have found an approximate match, we look at the metal lines and compare them to the spectral standard to confirm our spectral classification. Finally, if the closest standard is not a strong match, we compare it to spectral standards with adjacent classifications and determine if an interpolated spectral classification is more accurate.

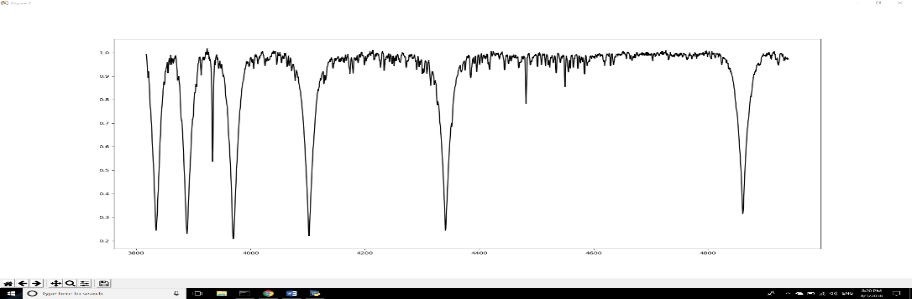
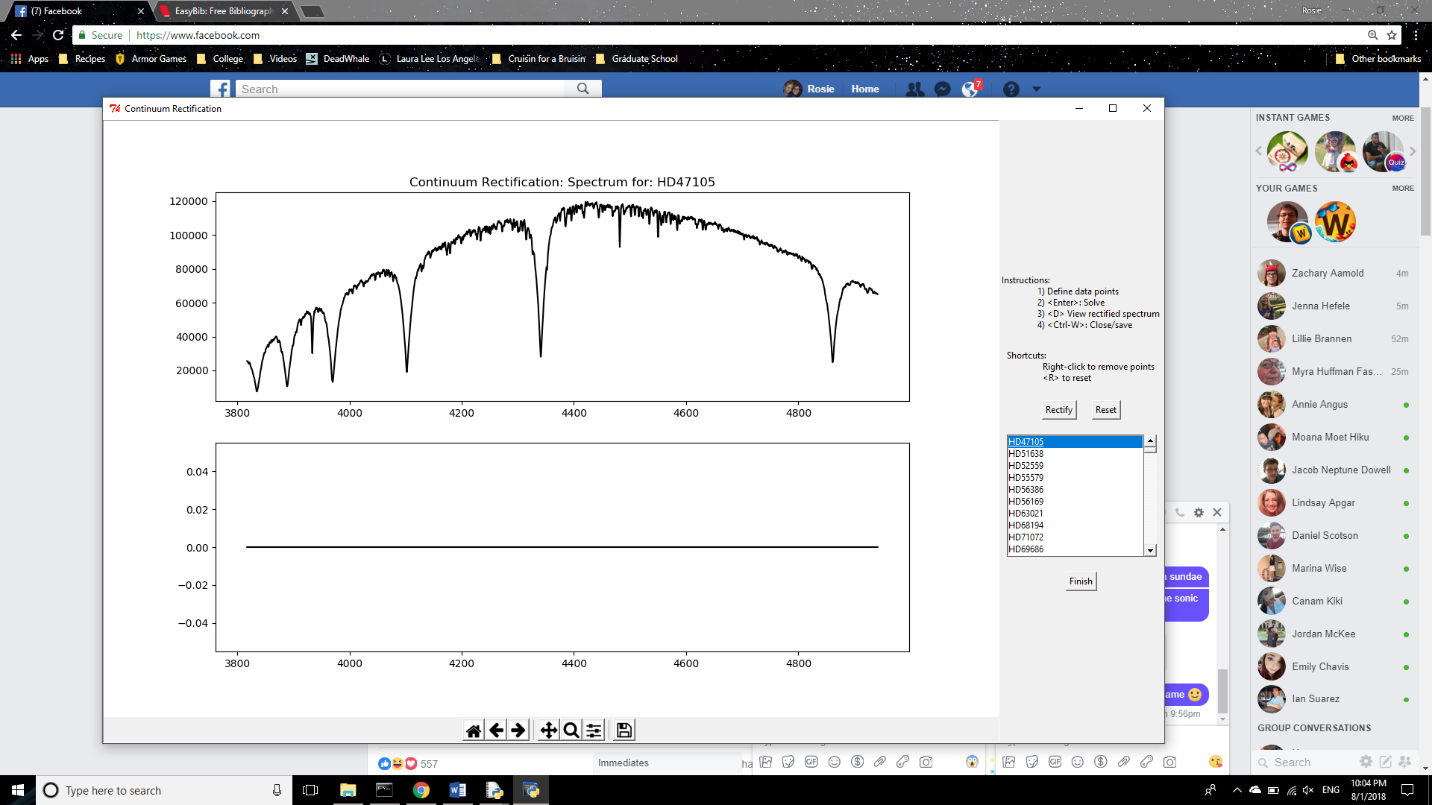
3. RESULTS

As the summer progressed, several changes were made in preparation for creating the automated program in the fall. The original program was weak in several areas, and throughout the summer these weaknesses were diminished or eliminated altogether. The first weakness was the wavelength calibration process. When given only a few stars to analyze the process is not too inconvenient, but when given many spectra, on the order of 50 or greater, the process was extremely time consuming. The user had to zoom in to the tip of each hydrogen point, select the point of the line, and then zoom out. To remedy this problem, an automatic wavelength calibration program was developed. Using a simple measure of the direction of the slope from one point on the spectrum to another, the program was able to successfully locate 6 out of 6 absorption lines for simple spectra. Once the locations of the absorption lines were found for one spectrum, the program then calculated the ratios between the absorption lines and used those as clues to look for absorption lines in all the other spectra. This increased the accuracy of the algorithm for more peculiar stars significantly, and the algorithm now always picks hydrogen absorption lines correctly for spectra that are not affected by emission. The program now can find the hydrogen lines automatically and greatly reduces the time the user must spend doing menial work (Figure 8).

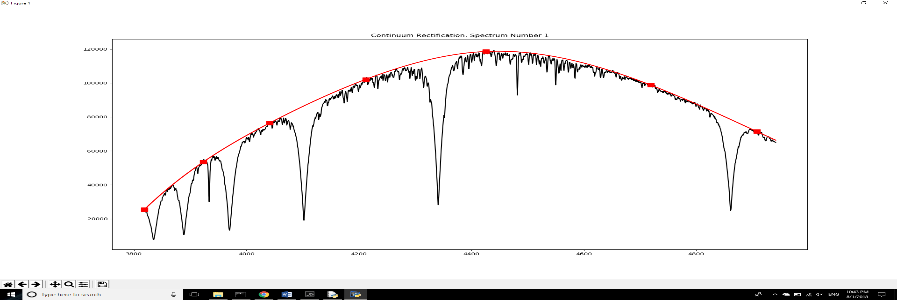


*Figure 8: Updated wavelength calibration. The points shown were automatically generated.*

The second weakness of the original program was with the continuum rectification process. The continuum rectification process is very important for the accurate classification of the star; the original method of continuum rectification did not aid the user very much in producing the most accurate continuum rectification possible. However, by simply changing the layout of the continuum rectification window, the user can create an accurately rectified continuum relatively quickly. The unrectified spectrum and the continuum rectified spectrum are now displayed side by side, so the user can see how the changes they make impact the rectified spectrum in real time. Additionally, the program now displays a horizontal line on the continuum rectified plot to provide a visual aid for the accuracy of the continuum rectification (Figure 9). There is work being done to produce an algorithm that will automatically rectify the continuum to further decrease the time the user must spend performing data reduction. However, at the time this paper is being written, this algorithm has not yet been completed.



*Figure 10: Updated continuum rectification. The line shown is used for reference*



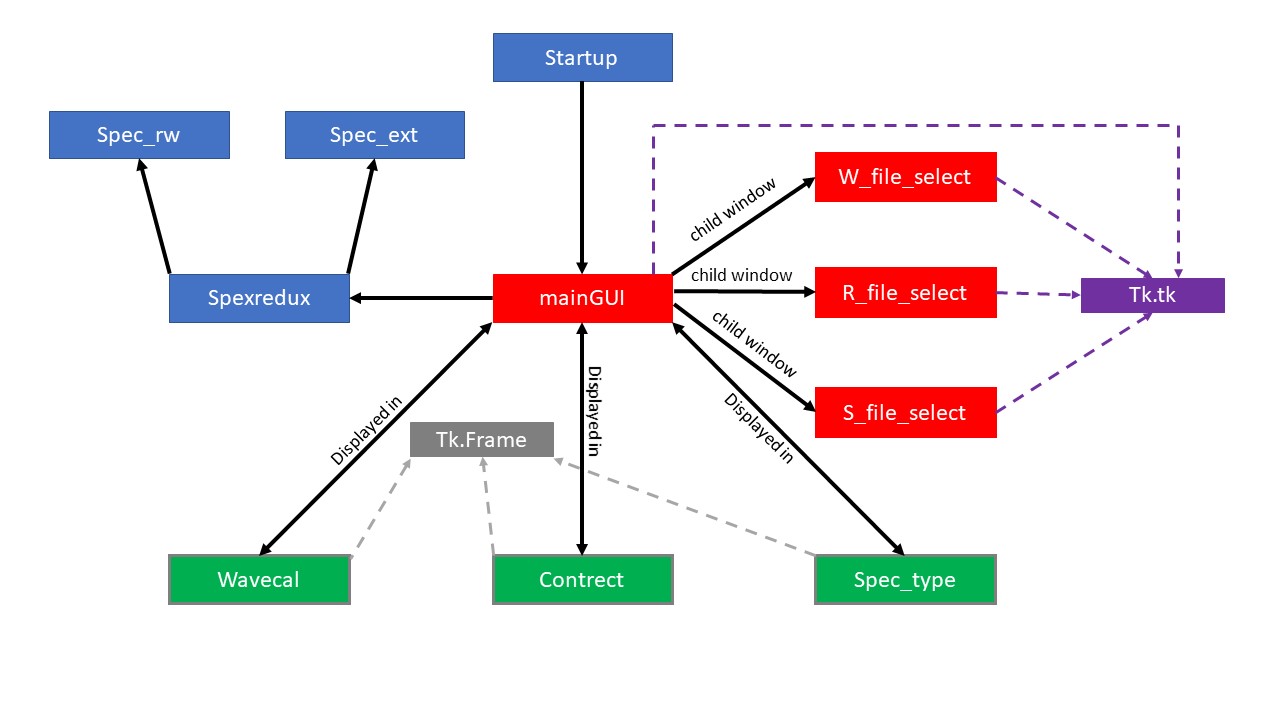
As of now, there have been no significant changes to the spectral typing process of the program, although the plans for how this process will be altered are outlined in the discussion section of this paper.

We now present our findings for the increased efficiency of the program. This was tested by performing wavelength calibration and continuum rectification on a set of data using the first version of the program, and then with the most recent version of the program. The spectral typing algorithm has not been altered as of yet, so it is unnecessary to test the time difference in spectral typing between the two programs. The data set was collected at the Adams Observatory on April 10th, 2018 and is a set of 16 spectra. Data reduction was performed in 38 minutes using the original version of the program, requiring 2 minutes for the extraction of the spectra, 14 minutes for the wavelength calibration, and 22 minutes for the continuum rectification. Using the most recent version of the program, the entire process took 22 minutes; requiring only 1.5 minutes for the extraction of the spectra, 30 seconds for the wavelength calibration and 20 minutes for the continuum rectification. This indicates an overall increase in efficiency by 57% due to changes made in the wavelength calibration and continuum rectification processes.

4. DISCUSSION

The results of the efficiency test clearly show that the changes made to the program dramatically decrease the amount of time the user must spend doing data reduction. Code optimization resulted in a 30 second decrease in the amount of time the program spends in spectrum extraction. The automation of the wavelength calibration means that very little time must be spent on that section. The user must simply check to see if the program has correctly located the hydrogen lines. There was not a significant decrease in the amount of time spent doing continuum rectification, but this is simply because the user must still perform the continuum rectification by hand.

The purpose of Giraffe Butts is to take the processes used in ao\_redux and make them much more convenient and efficient to use. Giraffe Butts will allow the user to easily select in what step of the data reduction and spectral classification process they would like to begin. In addition, aside from the file explorers used to locate the data to be used, the entire program will run in the same window. This increases efficiency and makes the program more aesthetically pleasing. As far as program design, ao\_redux was written as a group of standalone processes that are loosely connected to create one program. Giraffe Butts will make the connections more streamlined and make the program more cohesive, while still allowing for the possibility of performing each section of the program separately (Figure 10).



*Figure 10: Diagram of Giraffe Butts. The class mainGUI will serve as the central hub of the program. The blue branch represents classes that are used solely for spectral extraction. The purple dotted line indicates a class houses a tkinter Window. The red branch indicates that each class is a child of the mainGUI class. The gray dotted line indicates a class is an instance of a tkinter Frame. The green branch indicates the class is displayed by the mainGUI.*

The next step for this project is to develop an effective spectral typing algorithm. We are considering two possible options for the methodology we could use for spectral typing. The first option is to use a method like *MKCLASS* but specialized for O, B and A type stars. The second option considered is adapting *The Cannon* for use with spectral classification. Both options have strengths and weaknesses, and each option will be discussed in greater detail in the following paragraphs.

Specializing the algorithm employed by *MKCLASS* would involve two major changes. The first change would be the implementation of a classifying forest used to classify the spectra. The forest would consist of unique trees using a variety of criteria including a X2 comparison and the strength of metallic lines to determine a spectral type. Each tree would then submit a “vote” for what they think the spectral classification of the spectrum is and the type with the most votes would be selected as the spectral type for that spectrum. This creates an accessible and quantitative uncertainty value for the spectral classification of the star. The second change to MKCLASS would be the development of methods used to determine the spectral classification of spectra with peculiarities beyond chemical peculiarities. These methods would be implemented in a separate classifying forest from the forest used to classify normal stars. One of the advantages of specializing the *MKCLASS* methodology is that it still maintains a close connection with the MK classification that humans use to classify spectra. Another advantage is the lack of a need for a large volume of correctly pre-classified spectra for the program to work. Unfortunately, this method requires the maintenance of a large table of line strengths for use by the classifying forests. This table would require both a considerable amount of memory space, and the artificial definition of threshold values for line strengths.

Adapting *The Cannon* for use with spectral classification would require several steps for implementation. A relatively large collection of data would have to be compiled for use with *The Cannon*. The collection would have to be created by individually selecting the spectra that most closely resemble the spectral standards of a specific classification. The main classifying algorithm would also have to be redesigned to look for criteria specific to spectral standards. In addition, the program would have to be specialized detect spectral peculiarities of all types. The major advantage of this method is that it could easily be applied to any type of classification. The user would simply have to provide a selection of pre-classified spectra of the type the user wished to include for the classifying algorithm to train on. The other advantage of this method is that it does not require the maintenance of a large table of artificially created thresholds. However, there are disadvantages to using *The Cannon*, the first being that the Adams Observatory has only collected a set of 970 spectra at the time this paper is being written. Although our training set will likely be much smaller than the training set used in the paper about *The* Cannon, it is possible that the Adams Observatory may not have a sufficient number of spectra to successfully train the classifier employed by *The Cannon* (Ness et al. 2015). Additionally, *The Cannon* does not conform precisely to the MK classification philosophy. While it is based generally in the same idea, it does not utilize the same methods humans use when they perform spectral classifications and may produce results inconsistent with human classification. Assuming our goal remains to make a program that can emulate human spectral classifications, this could lead to inconsistent classifications.

Regardless of which algorithm is used for the spectral classification, both methods require the creation and maintenance of a database of reference spectra including spectral standards and stars confidently classified. This database will allow the program to become increasingly more accurate as it correctly identifies stars because it will have more known spectra to compare unknown spectra to. In addition, this database will allow for easy tracking of the changes of a single star over a period. Multiple spectra of a single star will be easily located and compared to identify any changes or trends in the spectra. Maintaining a database will also allow for spectral standards to be easily changed and for spectra that have been classified by hand to be easily added to the program for use by the classification algorithm.

This project will be continued during the 2018-2019 school year at Austin College. The following is a list of goals to be achieved during the fall semester of the upcoming year:

* Create a database of spectral standards and stars that are very confidently spectral typed by hand
* Create a basic algorithm that can correctly spectral type the stars kept in the database
* Expand the algorithm to be able to interpolate between spectral standards
* Implement functionality for the program to be able to identify stars that are not able to be accurately classified based on their spectra
* Expand the classification algorithm to cover spectra that are chemically peculiar
* Be able to detect and extrapolate individual spectra from a composite spectrum from a binary system

5. REFERENCES

Gray, R. O., and C. J. Corbally. “An Expert Computer Program For Classifying Stars On The Mk Spectral Classification System.” *The Astronomical Journal*, vol. 147, no. 4, 2014, p. 80., doi:10.1088/0004-6256/147/4/80.

Gray, Richard O., and Christopher J. Corbally. *Stellar Spectral Classification*. Princeton University Press, 2009.

Gray, R. O., et al. “Lamost Observations In Thekeplerfield: Spectral Classification With The Mkclass Code.” *The Astronomical Journal*, vol. 151, no. 1, 2015, p. 13., doi:10.3847/0004-6256/151/1/13.

“Hertzsprung-Russell Diagram.” *COSMOS - The SAO Encyclopedia of Astronomy | COSMOS*, Swinburne University, 2015, astronomy.swin.edu.au/cosmos/H/Hertzsprung-Russell Diagram.

Howell, Steve B. *Handbook of CCD Astronomy*. Cambridge University Press, 2006.

Morgan, W. W., and P. C. Keenan. “Spectral Classification.” *Annual Review of Astronomy and Astrophysics*, vol. 11, no. 1, 1973, pp. 29–50., doi:10.1146/annurev.aa.11.090173.000333.

Ness, M., et al. “The Cannon: A Data-Driven Approach To Stellar Label Determination.” *The Astrophysical Journal*, vol. 808, no. 1, 2015, p. 16., doi:10.1088/0004-637x/808/1/16.