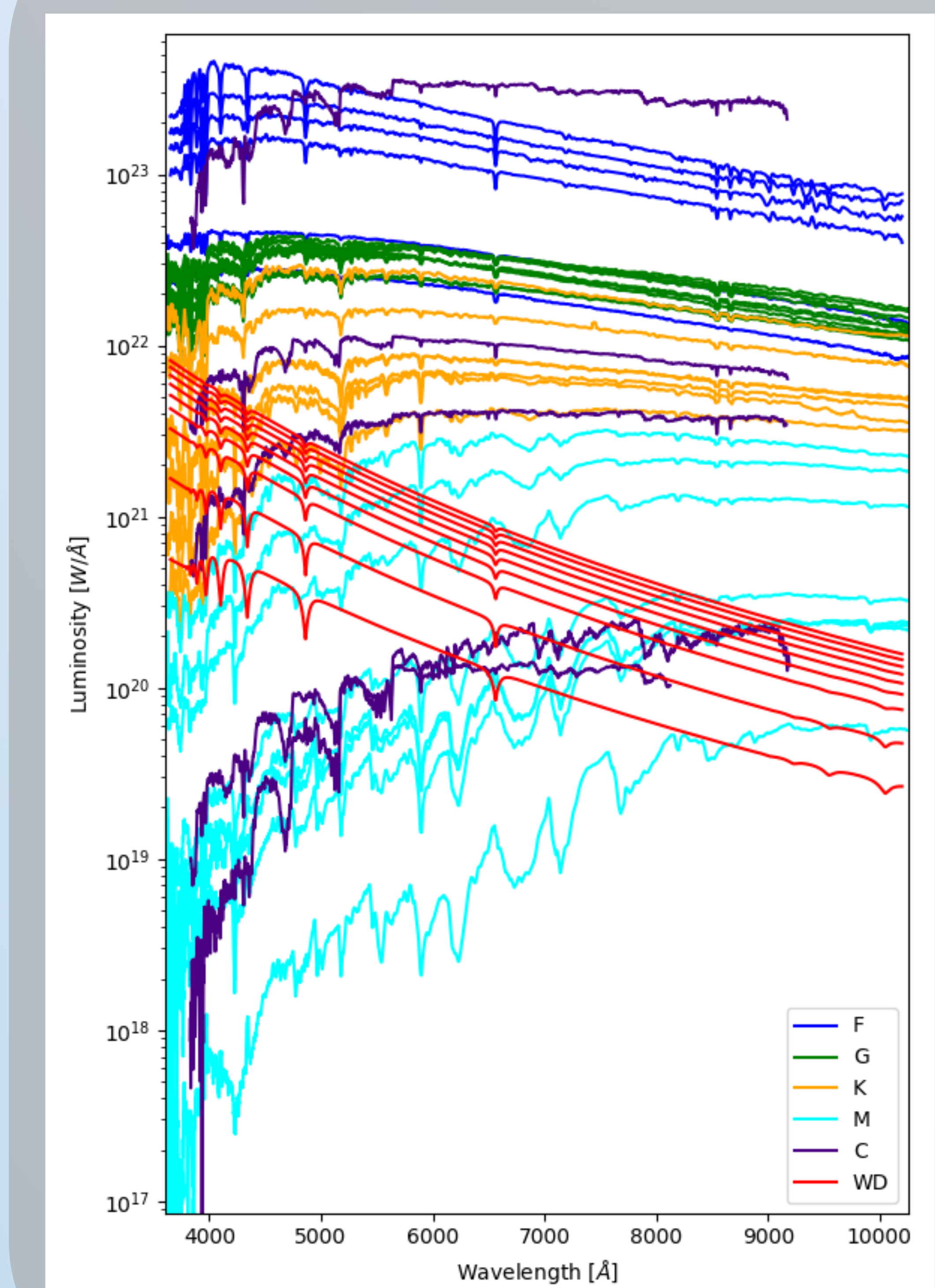
Detection and Spectral Typing of Binaries from Optical Spectra with PyHammerSB2

B.R. Roulston[1, 2], P. J. Green[1]

([1]CfA, [2]BU) – Contact: **broulston@cfa.harvard.edu**

New Luminosity Normalized Stellar Spectral Library



Spectral typing is a key component to understanding the physical nature of stars, stellar systems, and even exoplanets. Therefore, having a fast and accurate method for spectral typing is essential to large spectroscopic surveys.

Here we provide a new sample of spectroscopic binary (SB2) templates constructed from luminosity normalized SDSS spectra (left). We also preview a new version of the code "PyHammer" that can type these SB2 stars from optical spectra.

Figure 1. (left) – Examples of the spectral types included in the new SB2 templates. Fluxes were converted to luminosity units with Gaia DR2 distances. Note how the dwarf C stars cover spectral range from late M all the way to G types.

PyHammer

PyHammer is a Python spectral typing code based on the "Hammer" which was originally written in IDL by Covey+2007.

PyHammer was originally developed at BU by Kesseli+2017. They constructed a set of 324 single star template spectra from SDSS. These templates cover spectral types: O, B, A, F, G, K, M, L. They developed a GUI to visually inspect the resulting spectral classification and provide users with the ability assign spectral types by eye as well.

Our new version, PyHammer SB2, adds 363 new SB2 templates and new fitting features.

The classification algorithm is as follows:

1. User inputs a single spectrum, or a list of spectra to type. User can choose S/N cuts, and output file options.
2. For each input spectrum, PyHammer measures 43 spectral line indices, and compares them to the list of templates via a χ^2 test
3. The spectral type is chosen as the best match template, i.e. the template with the lowest χ^2
4. If the template match is in {O, B, A, F, WD} then PyHammer measures the width of Hα. Then compares this width to the templates to verify if the input spectrum is a WD or not (i.e. WDs have wider Balmer lines).
5. After all input spectra have been measured, the GUI is called allowing the user to visually inspect the matched types and choose their own classification if desired.
6. Once the visual check is complete and the user exits the GUI, the output files are written.

PyHammer SB2 (PHSB)

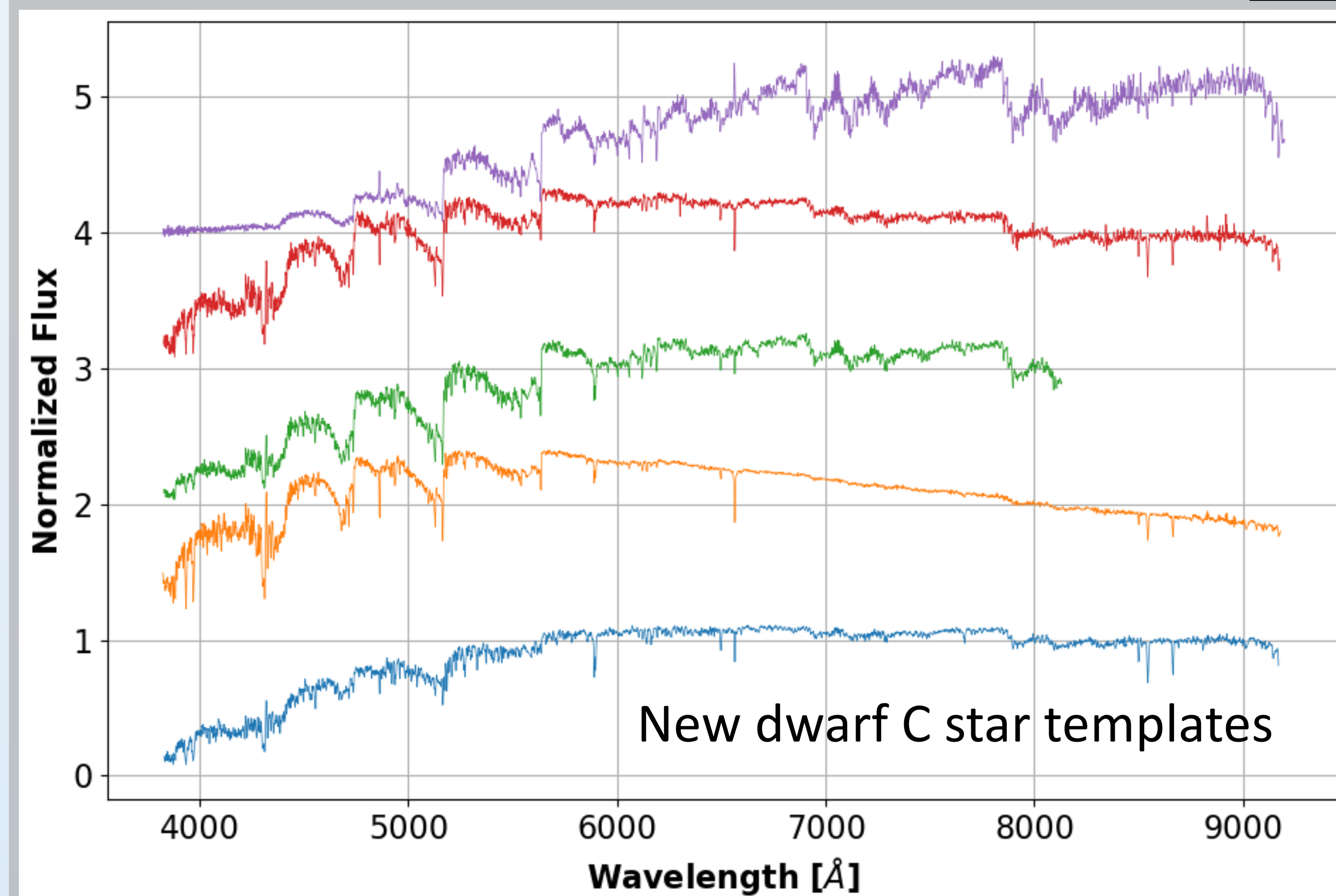


Figure 2. (left) – New single carbon (C) star templates that have been added to PyHammer SB2. There are 5 C star templates, . These templates are produced in the same method as those in the original PyHammer by Kesseli+2017, i.e. the coaddition of many spectra to make one particular template. For these C star templates the individual spectra were selected from the Green+2013 sample, which itself is a sample of carbon stars from the SDSS. We selected all the dwarf carbon stars from the Green sample, removed any that clearly showed the WD companion, and visually inspected, removing any spectra with bad flux regions. During the visual inspection, the spectra were grouped into one of 5 bins based on spectral shape, line strengths and color.

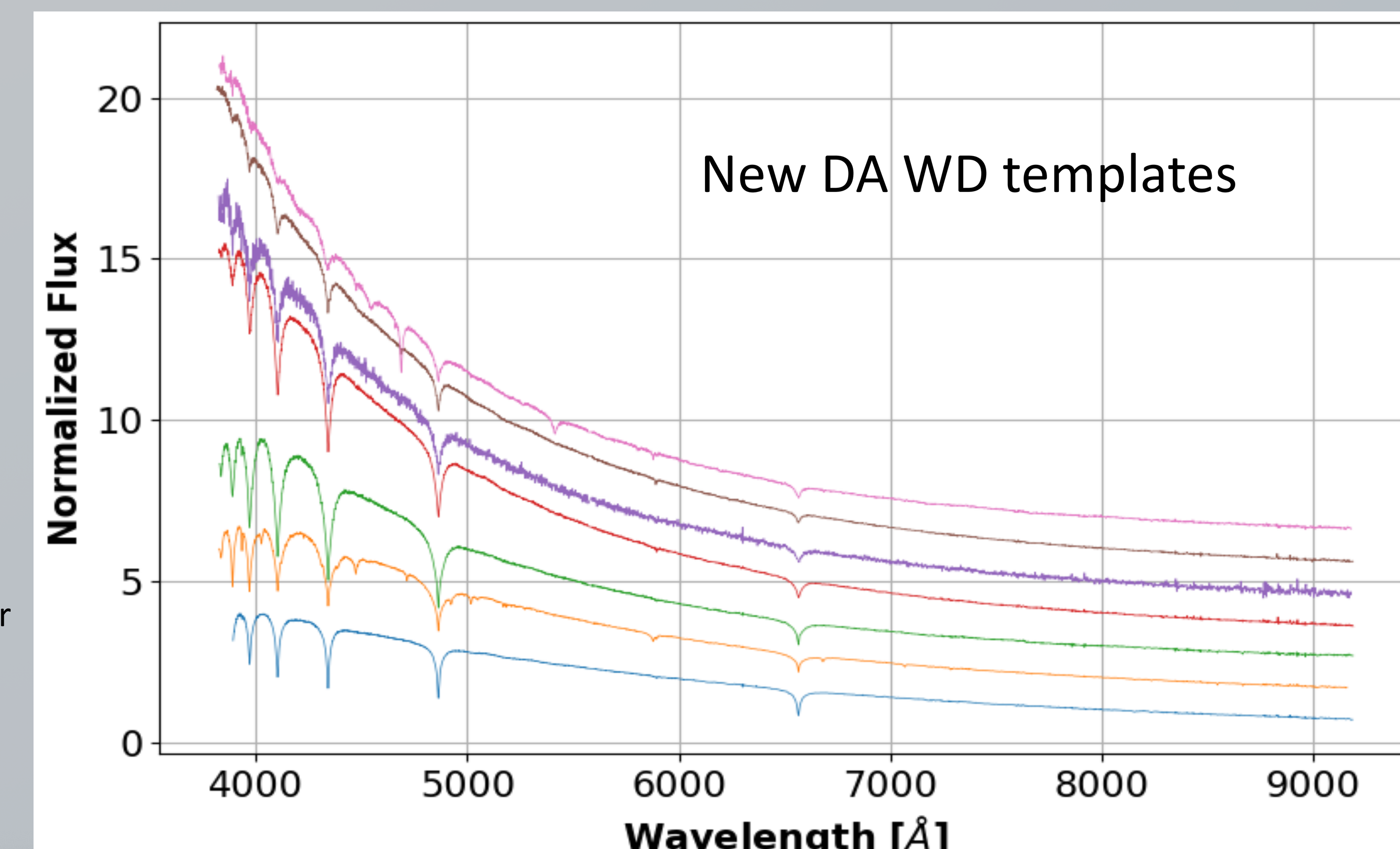


Figure 3. (right) – New single DA white dwarf (WD) templates that have been added to PyHammer SB2. There are 7 WD star templates that have been added. These WD templates were created from SDSS spectra from the Kleinman+2013 sample and were again inspected for bad flux regions and artifacts. These spectra were then binned in temperature bins ranging from 20,000K to 90,000K in bins of 10,000K. The spectra were then coadded to form the final 7 templates.

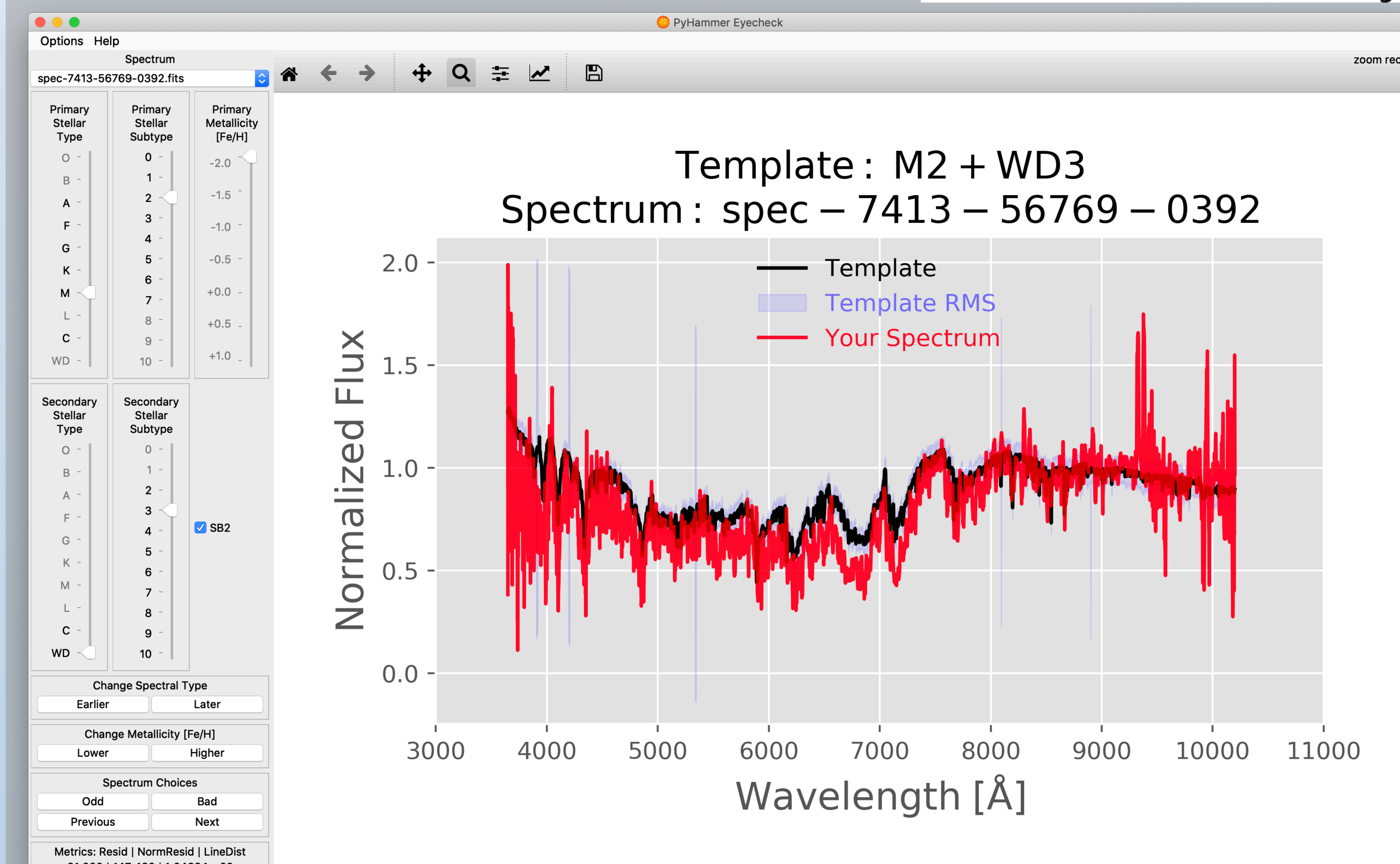


Figure 4. (above) – Example window of the GUI window of PyHammer SB2. After PyHammer has processed all of the users list of input spectra, the user has the option to inspect the classification of each spectrum. Our new version of PyHammer allows the option for specific combinations of spectral types to form SB2 composite spectra. The GUI provides sliders for the user to try the fit of different spectral types and subtypes for the SB2 template, while providing values of the goodness of fit

The output file saves the original PyHammer classification as well as the final user selected classification.

We only generated SB2 templates when the luminosity of each component spectrum had 20% of its pixels with in 20% of the luminosity values of the other component's spectrum. This ensures that the resulting SB2 has features from both spectra visible.

The resulting possible combinations we include are:

F+G
F+K
F+C
F+WD
G+K
G+C
G+WD
K+M
K+C
K+WD
M+C
M+WD
C+WD

and subtypes there in (e.g. M3+WD2).

References

Covey, K. R., Ivezic, Z., Schlegel, D., et al. 2007, AJ, 134, 2398

Green, P. 2013, ApJ, 765, 12

Kesseli, A. Y., West, A. A., Veyette, M., et al.2017, ApJS, 230, 16

Kleinman, S. J., Kepler, S. O., Koester, D., et al. 2013, ApJS, 204, 5

GitHub

https://github.com/broulston/pyhammer

SCAN ME