The Accuracy of NASA's STELLA 1.1 Device for Spectroscopy-Based Climate Research

Maxx Menzies, Aidan Anderson, Utkarsh Mandavilli, Pranith Bhat, Maura Thomas

Abstract

The Climate Adaptation and Resilience Monitoring Alliance (CARMA) team from the University of Colorado at Boulder was founded this year with the goal to build two STELLA 1.1 Spectrometers, create an instructional video series on how to build them, conduct a preliminary field test on campus, conduct an environmental research study on a topic of choice, and work with a graduate team on their long term environmental monitoring pod project (EVA). Researchers suggest that there is a gap in the temporal, spatial, and spectral resolution currently available with satellite-only remote sensing. [1] Aerial measurements have been completed in some areas, but this is limited due to flight cost. This leaves ground-based measurements and drone coverage. CARMA has the potential to fill both of these gaps as STELLA devices can take point readings and can be mounted to a drone. The CARMA team faced many unexpected challenges with the construction of the first STELLA 1.1 Spectrometer and was ultimately unsuccessful in getting it operational before the symposium paper deadline. However, the team made strides towards developing a better assembly method for the video series, outlined a plan for a research study that aims to verify the accuracy of the STELLA 1.1 device for climate-related spectroscopy field research, and developed new ideas for how to improve the 1.1 device moving forwards.

1 Introduction

The CARMA project's work is important because of the variety of applications that it has. The CARMA team has chosen to work with the STELLA 1.1 for now, and the 12 spectral bands that it measures enable us to detect information such as evapotranspiration for implications in agricultural monitoring, irrigation method effectiveness studies, and detecting threats to a harvest before a visible inspection would reveal plant stress. Plant leaf stress as seen in the near infrared spectrum allows early detection of negative environmental effects because it can serve as an environmental indicator. When mounted to a drone, the STELLA device can detect snow and ice pack coverage, which has implications in measuring water tables and predicting drought. Since the Rocky Mountains serve as the watershed for the Colorado River, drought prediction this way can be vital for the millions that depend on the Colorado River in the western United States. This is especially crucial as NASA's Landsat project won't be able to detect snow and ice pack coverage until 2031 after the launch of the next generation of Landsat satellites, upgrading from 9 spectral bands to 31. [2]

Over the course of the semester, the CARMA team faced many challenges that set us back from our original goals, however, we are now more prepared going forward to deal with these issues and have outlined our intentions to further the CARMA project in this paper.

1.1 NASA STELLA Spectroscopy Devices

STELLA (Science and Technology Education for Land/Life Assessment) is a NASA program that uses affordable parts to make a portable spectroscopy tool. The device measures near-infrared and visible light spectra to analyze various environmental indicators such as plant stress. The STELLA devices are designed to mimic the capabilities of the NASA and US Geological Survey's Landsat satellites that fully cover the earth once every eight days, reading 9 spectral bands with up to 30m resolution. Although intended as an educational tool, CARMA is working independently on verifying the accuracy of the STELLA instruments for field research applications.

1.2 STELLA 1.1 Spectrometer

CARMA has chosen to work with the STELLA 1.1 for the near future. The STELLA 1.1 parts cost about $200 and once built comes equipped with an air temperature sensor, a thermal infrared sensor (TIR), a pressure and humidity sensor, a rechargeable battery, a display screen, and a near infrared spectral sensor (NIR) and visible spectra sensor (VIS) from SparkFun. The 1.1 takes measurements in 12 bands across the wavelength range 460- 860nm and can also be adapted to mount to a drone for use in hard-to-reach areas or to cover more ground at once.

2 Accomplishments

Over the course of the semester, the CARMA team was able to make significant accomplishments throughout the work on creating the STELLA 1.1 instruments and in the research done throughout the project. One of the accomplishments was that the CARMA team was able to successfully identify issues in the NASA STELLA 1.1 instructions and create a document that outlines those issues along with their proposed solutions. One of these proposed solutions is to create an edited building document with more accurate visuals, clearer instructions, and a different order of steps in order to make the STELLA 1.1 easier for classrooms to utilize. Another of these proposed solutions includes a video series, which, in addition to the building instructions, will walk the builder through the steps of creating their own STELLA 1.1. The series will guide viewers along as they build a STELLA device with the CARMA team. The recommended approach in the videos is different from the NASA approach, which recommends soldering on roughly 50 wires before attaching the components and then troubleshooting the completed device. However, from experience, the CARMA team has decided that first building the circuit on a breadboard to test all the components and then wiring them onto the protoboard one by one, running the NASA test codes throughout the process, is a much simpler method in the long run for debugging.

Through the troubleshooting of STELLA 1.1 issues during building, the CARMA team was able to determine an easier method for testing the various components and locating faulty ones, as the build instructions forced us to deconstruct much of the instrument in order to test defective components, which was time-consuming and did not easily pinpoint the issue. That we have determined an easier method than this is useful, as it is likely that users may encounter issues with faulty components (See Future Plans).

We determined that the best course of action for allowing us to utilize the STELLA 1.1 on other projects would be to prove its accuracy. This research helped us gain a better understanding of the limitations which could affect the accuracy of the STELLA 1.1 as well as introduce us to the possibility of creating an alternative STELLA 1.1 using an Arduino microcontroller, which would be more suited to our knowledge and to keeping down costs (See Future Plans).

3 Research Methods

3.1 Research Plan Overview

Although we plan to use the STELLA 1.1 device for future research, NASA states that the 1.1 is a purely educational tool since it has not been independently verified for accuracy. Before more work can be done with these instruments, we must be able to support the findings with the knowledge that the readings are accurate. We aim to verify its accuracy over many ranges of reflectance, proving the device's validity over the full range of reflectance one might find during field research. To do this, we will compare the reflectance of Standard Reference Materials (SRMs) to the reflectance we measure. We will use the mathematical methods Root Mean Square Error (RMSE), Coefficient of Determination (R^2), and Mean % Error. These methods will compare the average deviation, measure correlation between, and check for over/underestimation against the SRM's expected values. Conducting multiple trials with various light sources will ensure the reproducibility of our results. This data will then be interpreted to determine if the instrument passes minimum standards to be considered accurate. We expect to find that the STELLA 1.1 instrument will be accurate within a mean percent error of +/- 10%, based on the claims made by the parts manufacturers, making them suitable for detection of broad environmental trends that the device is meant to analyze, but not accurate enough to compare to lab-grade analysis seen by spectrophotometers costing thousands of dollars. Ideally, we would want to find an error of +/- 5%, as we believe this is ideal for a field instrument such as the STELLA 1.1.

3.2 Purpose of Research Project

The goal of our research beyond enabling us to conduct novel climate research using a 1.1 device is to enable a wider audience of students to do the same. Our independent accuracy verification of the 1.1 will serve as a base that can allow high school and college students around the world to assemble a device and conduct their own studies. Just like how STELLA was started at NASA to educate students about remote sensing and the environmental monitoring technology of the Landsat satellites, by verifying the 1.1 for spectroscopy research purposes these students can take it one step further by starting their own research or contributing to citizen science projects.

3.3 Standard Reference Materials

The SRMs that CARMA has chosen for this project will allow our verification research to be easily replicated by the intended audience of the STELLA devices and our video instructional series: students. With this in mind, we chose photography cards. A set of photography white balance and color grading cards usually contains a black card (5% reflectance), an 18 grey card (18% reflectance), and a white card (90% reflectance). These are standardized quantities that are utilized in photography for their unique qualities of having a flat reflectance spectrum across the visible and near infrared ranges. For example, since we know that the 18 grey card should have a 18% reflectance regardless of the light source, we know that any deviation of our sensors from 18% reflectance is due to an error in that spectral band reading from the device. With the addition of the WhiBal G7 card the study can incorporate readings in the middle of the reflectance spectrum. Although the WhiBal card is not of uniform perfectly neutral reflectance like the rest, CARMA will be comparing our results to a study that used commonly accepted spectrometers to read spectral data for it.[3] The standard white, black, and grey photography cards can easily be found on Amazon for approximately $10, and the WhiBal card can be found for about the same, although we have elected to purchase a larger one for about $50.

3.4 Data Collection Methods

The CARMA team plans to conduct 3 trials on the reflectance of each SRM under 4 different light sources. The light sources will be indoors underneath the florescent lighting of standard commercial lights in the Colorado Space Grant Building, outside with direct sunlight, outside with overcast clouds, and with only the light produced by a phone flashlight. By varying the light sources, we hope to ensure that our findings will be applicable for future field studies done under non-regulated light of varying intensities and spectral makeups. Each trial will consist of one reading taken by the STELLA 1.1

spectrometer, held at a fixed distance for each trial that ensures the field of view of the sensors is entirely consumed by the SRM. Once collected the data will be analyzed according to the mathematical models described in the next section.

3.5 Data Analysis Methods

The 3 mathematical models that we plan to use to interpret the 1.1 data are Root Mean Square Error (RMSE), Coefficient of Determination (R^2), and Mean Percent Error. The combination of these methods will allow a comprehensive analysis of the data from multiple lenses, hopefully revealing the cause of the error in the device.

RMSE is a mathematical model that measures the average magnitude of the errors between the values the 1.1 will measure and the known reference values, in this case the percent reflectance of the different cards. This will tell us how far, on average, the measurements are from the expected values. To do an RMSE analysis, the CARMA team will find the difference between the expected and measured reflectance, square each error (this has the dual function of making the value positive and punishing a larger error more than a smaller error), take the mean, and then find the square root of the mean.

R^2 (or the coefficient of determination) will tell us how well the trend of the measured data matches the trend of the true values. To calculate R^2, first plot True vs. Measured reflectance, then compute the regression using excel, a custom python script, or by using existing tools online. R^2 is intended to show whether the 1.1 will preserve the relationship between measured values across a range of reflectance. If we are consistently overshooting the measurements by the same amount, this will be seen in the data by an R^2 value close to 1. If this is the case, the CARMA team plans to edit the code of our device to account for this error. If in the future we conduct this study again with multiple devices and find that the errors generally match this pattern, CARMA plans to release our updated code with a feature allowing other researchers to conduct a similar study and input their R^2 value to get their instrument adjusted. A script like this would use a simple linear regression model to find a value to add or a scalar multiple to shift the data and account for error.

Mean Percent Error measures bias. If our device is consistently overestimating or underestimating the reflectance values, then this analysis will find that. A positive bias indicates that the STELLA is overestimating, and a negative bias would indicate that it is underestimating. This is useful to find systematic flaws in our experimental methods, as well as helping to confirm the results seen by the R^2 method and a possible calibration script. To find the Mean Percent Error bias, the measured reflectance values will be subtracted by the true value, divided by the true value, then multiplied by 100 to give a percentage.

4 Current Challenges

In the construction of our own STELLA 1.1 devices, we have encountered many hinderances towards our success.

We were initially supplied with three semi-completed STELLA devices that were used by another research team prior to our involvement. These devices were a similar, but a distinct model older than the 1.1. The previous group was able to read data from the devices, however we were unable to successfully connect a computer to the microcontroller of any of the devices. Due to the age of the models, the original instructions were removed from the STELLA website. This problem was further compounded by the fact that the instructions for flashing the spectrometer with code or even powering it on varied between different models, making it hard to track and follow. This rendered the older models difficult to debug and impossible to use.

To understand the experience of a team or classroom using the STELLA as an educational tool, our first STELLA 1.1 was built by closely following the provided instructions. We began by setting up all the wires and non-sensor components on the protoboard to match the provided images. We also utilized a list of wire and component coordinates to ensure proper wire placement. This provided a good opportunity for our team to practice or learn how to efficiently cut, strip, and solder wires. However, this process was time-consuming and provided very little insight into why the wires were placed as they were. While we did not have any significant wiring misplacements, a user with less electronics experience could be reasonably expected to accidentally misplace wires if they are only given images and a list of coordinates to work with.

The next step was to manually alter the Near Infared Sensor and Datalogger components so that they could be properly implemented in the device. While this task was not difficult, being inexperienced in modifying components resulted in lacking confidence in if the components had been properly modified. Additionally, the reason for the modifications was not given, and would have provided insight into the building process. For example, the Near Infared Sensor required the JP1 connection to be desoldered, and the JP2 connection to be soldered closed. We would later learn that this connection change resulted in the component changing the method with which it communicates to the microcontroller from being on the I2C bus to a UART communication system. The reason that this change was necessary was because the I2C address of the Near Infared Sensor is the same as the I2C address of the Visual Spectral Sensor, meaning that both sensors can't connect via I2C at the same time. This information would have been useful to know earlier in the build process and would be a good learning opportunity for a class learning about the software communication methods utilized in the STELLA.

After the wiring was completed, the sensors were then soldered onto the protoboard in their designated positions. Soldering the sensors at this point in the building process would quickly become our biggest mistake, since at this point none of the sensors had been individually tested. We would later figure out that some of the sensors were faulty but were unable to remove them from the board. Whether the components had always been non-functional or had been broken at some point in the building process, we do not know. Regardless, testing the functionality of the sensors prior to installation is an important step in the building process, and lots of time could have been saved had we properly identified faulty components. This issue that we faced during the build process of the first spectrometer with the different components allowed us to improvise for the build of the second spectrometer and implement a test breadboard that doesn't require soldering and utilizes simple jumper wires. This allowed us to recreate the circuit on the spectrometer but still be able to test each individual sensor with the respective test codes and move the sensors around. We can now avoid having to go through the process of desoldering, and still ensure each sensor is working before soldering them onto the final circuit board.

The Python Test codes were given to us, stored in a file that allowed us to flash each individual test code that tested every sensor and power-seeking components on the spectrometer. Although the code was meant to help debug errors quickly, it wasn't as helpful as intended. There was a lack of print statements or error handling code that was implemented which made it hard to track exactly which function or loop the error was being triggered. For example, during our first several tests of the NIR sensor, it continuously printed a message that told us we hadn't modified the component properly by soldering JP2 and desoldering the JP2 connection. However, we had confirmed that the modification had been made, and after several weeks we were able to track down the issue to an unstable UART connection to the microcontroller and a faulty sensor board, an issue unrelated to the message we were receiving from the test codes. We also saw multiple I2C issues when running the code for other sensors such as Visible Light and Thermal Infared. The issue that was printed implied that there was a problem with the I2C serial bus. After thorough inspection of both the Serial Data and Clock lines, we found that the I2C lines were not the issue, but in fact, another defective sensor.

Implementing the display board was one of our biggest challenges in completing the STELLA 1.1. Due to no fault of the given instructions, we failed to recognize that the 2.8-inch TFT LCD display screen we had purchased was not the correct component. Specifically, the display board that we had lacked the ability to connect to the I2C bus. This meant that the board was unable to use any touch screen capabilities. While the screen itself was perfectly functional and was able to display data when tested, the STELLA 1.1 requires the touch screen function to operate. Also, since the display board's I2C address was not present, the Python code could not be run due to not all necessary components being detected. This was quite an unfortunate mistake, and valuable time was lost waiting for the correct display board to arrive.

5 Future Plans

5.1 Spectrometer Redesign Overview

Due to the numerous difficulties/challenges experienced throughout this process, as mentioned in Section 4, the CARMA team plans to design and build our own version of the STELLA 1.1/1.2 during the summer. We hope to consolidate the sensors and simplify the hardware components of our design. In addition, we plan to transition to an Arduino microcontroller instead of the Feather. Our team has far more experience using the Arduino IDE and we believe that this would benefit us greatly with debugging, writing new test code, and implementing new functions. Additionally, rewriting the test code in C++ allows us to improve our error handling and customize our spectrometer for the specific research objectives we have. We could also potentially implement calibration within the software itself, which would be helpful to avoid needing to calibrate each individual batch of data. As discussed in Section 4, the assembly instructions for the spectrometer are written in such a way that the testing of individual components is not possible, and all wiring must be done before components are placed on the protoboard. This method of assembly makes debugging very challenging and does not follow the standard engineering process. We believe that we can improve upon this and implement new design instructions that allow for testing at each step of the process while also teaching students what each component is doing and how that is integrated into the system.

5.2 Sensor Consolidation

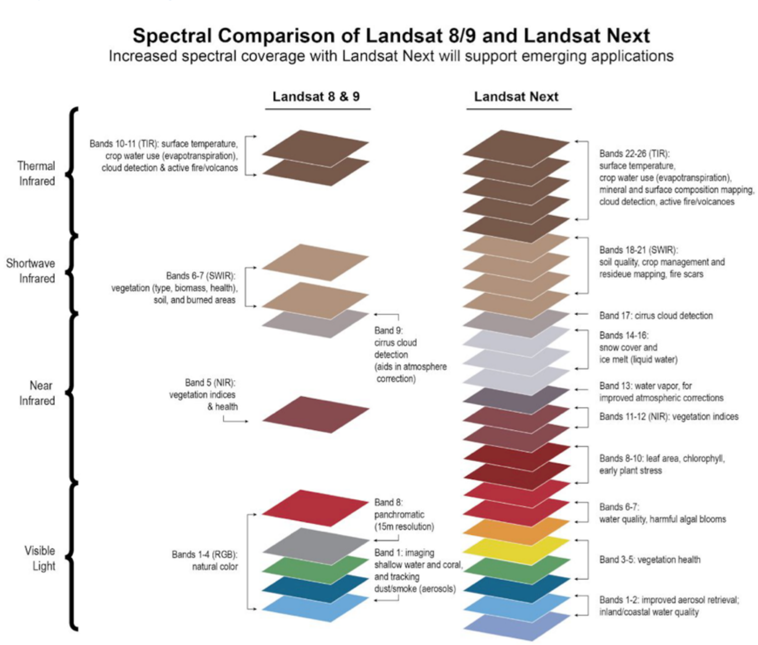
As for the consolidation of sensors we plan on using the SparkFun Triad Spectroscopy Sensor - AS7265x, which combines the Near Infared Spectrum Sensor and the Visible Light Spectrum Sensor in addition to an Ultraviolet Spectrum sensor. Using this part would reduce the number of components connected to the I2C bus and simplify debugging. Currently, using a separate NIR and VIS sensor forces us to use both the I2C communication protocol and the UART protocol. This is due to the sensors sharing the same address on the I2C bus, meaning that they cannot be detected individually. We will continue to use the BME280 sensor which allows us to measure barometric pressure, ambient temperature and humidity. The third and final sensor will be a surface temperature sensor, the SparkFun IR Array Breakout - MLX90640. This sensor is capable of measuring surface temperature accurately from several feet away with a high degree of accuracy, allowing researchers to measure various parts of plant life, that may be otherwise inaccessible. Using only 3 sensors will streamline testing and assembly on the breadboard making the process more intuitive and straightforward. These sensors will all be connected through the I2C protocol and eliminate the need for other protocols. This simplification also allows clears up board space for the addition of more sensors such as, LIDAR, GPS, CO2, Magnetometer, and Ambient Light.

5.3 Microcontroller Change

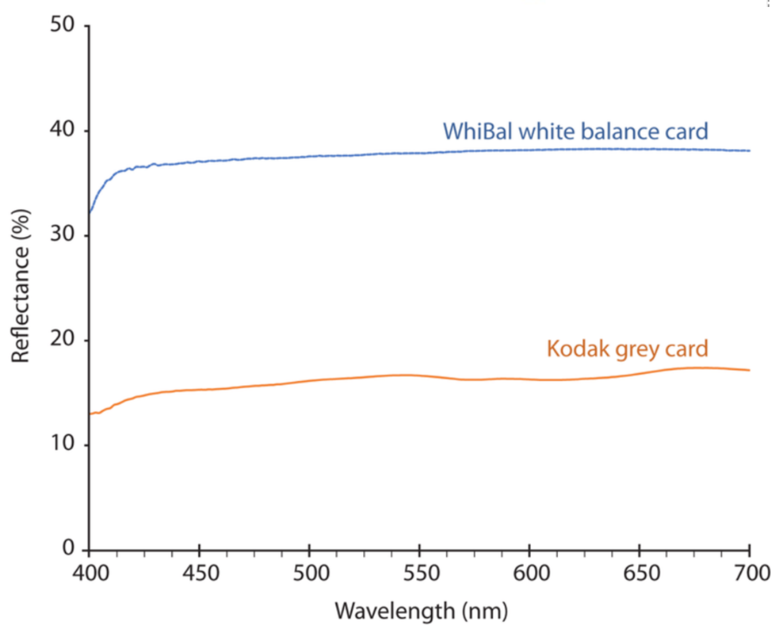
Transitioning to an Arduino microcontroller is another step that we hope to take to simplify the process of making the spectrometers. Our team has much more experience coding using the Arduino software and its associated libraries. Translating the code from Python to C++ will be a relatively smooth process, as it is mainly a change in syntax rather than logic. In addition, we ran into detection issues when using the Feather boards, as mentioned in Section 4. The Arduino does not rely on other components being attached to it in order to be detected by the computer. This makes implementing test code and working with individual components far easier.

References

(1) Hernández-Clemente, R., Hornero, A., Mottus, M. *et al.* Early Diagnosis of Vegetation Health From High-Resolution Hyperspectral and Thermal Imagery: Lessons Learned From Empirical Relationships and Radiative Transfer Modelling. *Curr Forestry Rep* 5, 169–183 (2019). https://doi.org/10.1007/s40725-019-00096-1



(2)



(3)

Verhoeven, Geert. (2016). Basics of photography for cultural heritage imaging.