CS Capstone Progress Report

June 13, 2018

Coal and Open-pit surface mining impacts on American Lands Follow-On (COAL-FO) - Turning spectro-imagery into usable data

Prepared for

NASA JPL

Lewis John Mcgibbney

Prepared by

Group 28
COALFO

Kenny Thompson
Bryce Egley

Abstract

Coal and Open-pit surface mining impacts on American Lands Follow-On (COAL-FO) is the successor project to the 2016-2017 COAL project. COAL initially aimed to deliver a suite of algorithms to identify, classify, characterize, and quantify (by reporting a number of key metrics) the direct and indirect impacts of mining operations and related destructive surface mining activities across the continental U.S (and further afield). COAL successfully delivered a Python library for processing hyperspectral imagery from remote sensing devices such as the Airborne Visible/InfraRed Imaging Spectrometer (AVIRIS) and a Science Data System for running COAL pipelines. COAL-FO will utilize recent funding obtained from a recently awarded NSF-funded XSEDE high performance computing (HPC) grant to further improve, validate and document COAL algorithms, execution runtime performance and geospatial output results.[1]

CONTENTS

1 PART 1: Introduction to Project

1.1 Who requested it?

This project was requested by Lewis John McGibbney a Data Scientist working for the NASA Jet Propulsion Laboratory.

1.2 Why was it requested?

Coal and Open-pit surface mining impacts on American Lands Follow-On (COAL-FO) is the successor project to the 2016-2017 COAL project. COAL initially aimed to deliver a suite of algorithms to identify, classify, characterize, and quantify (by reporting a number of key metrics) the direct and indirect impacts of mining operations and related destructive surface mining activities across the continental U.S (and further afield). COAL successfully delivered a Python library for processing hyperspectral imagery from remote sensing devices such as the Airborne Visible/InfraRed Imaging Spectrometer (AVIRIS) and a Science Data System for running COAL pipelines. COAL-FO will utilize recent funding obtained from a recently awarded NSF-funded XSEDE high performance computing (HPC) grant to further improve, validate and document COAL algorithms, execution runtime performance and geospatial output results.[1]

1.3 What is its importance?

Overall this project is very important since we need to track and monitor the environmental changes and effects coal has on surrounding lands since Coal is one of America's main sources of energy. We want to make sure we are extracting coal and using it safely for the sake of the environment. Satellite imaging using remote sensing devices allows us to gather much more data than a human eye is able to. Satellite imaging can provide a picture of an entire area of land and see things invisible to the human eye such as methane emissions.

1.4 Who was/were your client(s)?

The client for this project was Lewis John McGibbney a Data Scientist working for the NASA Jet Propulsion Laboratory.

1.5 Who are the members of your team?

The team members for this project were Bryce Egley and Kenneth Thompson.

1.6 What were their roles?

Bryce Egley focused on pycoal and the website for the duration of the project. With the specific focus of building a CLI, allowing Pycoal to support USGS Spectral Library Version 7 and the EcoStress Spectral Library by creating conversion functions to convert the Spectral Libraries into a suitable file format for pycoal to use. As well as coming up with installation instructions on the website for installing GDAL and QGIS for any type of Operating System. And other smaller tasks such as finding new data and fixing existing bugs in pycoal.

Kenneth Thompson focused on COAL-SDS for the beginning of the project and then moved over to pycoal to work on the classifier callback.

1.7 What was the role of the client(s)? (I.e., did they supervise only, or did they participate in doing development)

The main role of Lewis the client was supervising. Once code was submitted by Bryce and Kenneth via pull requests, Lewis would offer feedback, or make minor changes to the code before accepting it or asking for us to keep working on it.

Problem Statement: Coal and Open-pit surface mining impacts on American Lands Follow-On (COAL-FO)

Bryce Egley and Kenny Thompson
CS 461: Senior Capstone

Abstract

In 2016-2017, three computer science seniors from OSU developed software to detect the damage caused by coal mining. Their work was impressive, they developed a suite of algorithms designed to detect and track damage using satellite imagery. They even managed to win an award and get mentioned in the OSU alumni newspaper. However there were several promising opportunities to use their work that was uncompleted at the time of graduation. Our project, and our goal, is to take their work, improve upon it, and share it for the world to see. We will be doing it by taking their work and their algorithms, and converting them into tools that will run on the XSEDE cluster. We will run it on the XSEDE cluster, feeding it Terabytes of publicly available spectrometer readings collected from the NASA AVIRIS project. We will then take that data, and create a search engine to view the processed data in a readable form. People have created these tools before, and done these analysiss before, but it has not been created and made publicly viewable online. This will make future projects easier for researchers in this field, and will create new and exciting opportunities for scientific analysis.

Definition and Description

This project will create new as well as improve upon existing methods to classify, characterize and quantify impacts of mining and other destructive surface mining activities across the United States. The COAL project will now go on XSEDE, a single virtual system where individuals will be able to share computing resources. By putting the COAL project on XSEDE others will be able to use the program and data that comes from the COAL project. This will also enable us to further improve the algorithm's, execution runtime performance and geospatial output results. The original COAL project focused on gathering data from remote sensing devices, providing a suite of algorithms for classifying land cover, identifying mines and other graphical features, and correlating them with environmental data sets as described on the coal capstone github page.

This is the fourth capstone project and hundreds of people have forked the code from the previous project off github. Since, so many people are using the code produced in this project it is very important that we make images viewable on XSEDE so people who want to use the data from this project in the future will be able to easily gather the data they want and be able to apply it to their own projects.

Overall this project is very important since we need to track and monitor the environmental changes and effects coal has on surrounding lands since Coal is one of America's main sources of energy. We want to make sure we are extracting coal and using it safely for the sake of the environment. Satellite imaging using remote sensing devices allows us to gather much more data than a human eye is able to. Satellite imaging can provide a picture of an entire area of land and see things invisible to the human eye such as methane emissions.

Proposed Solution

So, our problem is, that to get this data which is essential to many research projects, researchers have to create tools and process the data, sometimes processing data thats already been done. The first half has already largely been solved, thanks to the 2016-2017 capstone team. The real issue is the second part. In order to solve the problem addressed. we wish to put the COAL-FO (COAL follow on) project on XSEDE, a high powered computing cluster, and analyze as much publicly available information as possible. XSEDE only allows for 2 terabytes so we will come up with a method to feed it data that will be looped through. The main priority for our capstone project is making a searchable port for the COAL project. We also plan on having meetings every two weeks where we will turn in and discuss past week deliverables and plan future deliverables. This is where we will create a schedule of how to complete the project. As already discussed we will also plan on improving the existing COAL architecture and systems. We also hope to improve the imagery processing algorithms currently in use. Then we will hope to create a new baseline suite to rank changes to areas of land over time to determine if the changes in one particular area our significant or not.

Performance Metrics

The performance metrics for this capstone project will be to complete as many of the objectives as possible discussed in the abstract and the project description. Our main priority according to our client is to create a searchable port for the project. Once we complete this we will move on to the other objectives. We will also have bi-weekly performance metrics based on the meetings we have with our client and the deliverables that come from those meetings. If we can complete those objectives and the high priority objectives then this project will be a success.

This means we don't necessarily need to complete all the objectives for this project to be successful. As long as the high priority objectives like the searchable port and maybe putting the project on XSEDE are completed. The other objectives should only be given focus once we have completed work on the high priority objectives. These lower priority objectives would include improving existing algorithms and improving the accuracy of finding correlations between land changes in different images.

Overall this project looks challenging and it seems like a lot of work but if we focus on one objective at a time and use the resources available to us I believe we can accomplish several of the high priority objectives. We also hope that the work we do on this project will make it more accessible and usable for other individuals working on their own projects and research. We look forward to working with the NASA Jet Propulsion Lab as well as the other members of the COAL project from previous years.

1.8 Project Proposal

Coal and Open-pit surface mining impacts on American Lands Follow-On (COAL-FO) is the successor project to the 2016-2017 COAL project. COAL initially aimed to deliver a suite of algorithms to identify, classify, characterize, and quantify (by reporting a number of key metrics) the direct and indirect impacts of mining operations and related destructive surface mining activities across the continental U.S (and further afield). COAL successfully delivered a Python library for processing hyperspectral imagery from remote sensing devices such as the Airborne Visible/InfraRed Imaging Spectrometer (AVIRIS) and a Science Data System for running COAL pipelines. COAL-FO will utilize recent funding obtained from a recently awarded NSF-funded XSEDE high performance computing (HPC) grant to further improve, validate and document COAL algorithms, execution runtime performance and geospatial output results.

2 PART 2: REQUIREMENTS DOCUMENT AND DESIGN DOCUMENT

2.1 Original Requirements Document

CS 461 Requirements

Team Name: COAL-FO

Bryce Egley, Kenny Thompson

1. Introduction

1.1 Purpose

The purpose for this document is to layout the requirements for the COAL-FO capstone project and describe how these requirements will be accomplished.

The purpose of our COAL-FO project as discussed in our last meeting with our client is to to create a searchable port for the existing COAL project. This will be augmented to accommodate the desire to (i) port the coal-sds software to the XSEDE platform and undertake test and evaluation of the system performance, (ii) process all AVIRIS and AVIRIS-NG imagery generating and archiving all science data products, and (iii) making the products searchable through a portal.

1.2 Scope

The project will be focused around publicly available COAL algorithms created by the previous capstone group, publicly available spectral analysis collected from the NASA AVIRIS project, and a grant on the XSEDE environment to use HPC. Work for this project will be stored in the Capstone-Coal organization on github and version control as well as future releases for this project will managed within that organization.

1.3 Definitions, Acronyms, Abbreviations

COAL-FO: Coal and Open-pit surface mining impacts on American Lands Follow-On

HPC: High-performance computing

XSEDE: Extreme Science and Engineering Discovery Environment

AVIRIS: Airborne Visible / Infrared Imaging Spectrometer

1.4 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

1.5 Overview

This project will, to put it in rough terms, take existing algorithms, export them to the XSEDE environment, a high powered computing cluster and process as much publicly available spectro imagery as possible, and use that to create an archive. A stretch goal will be to take that archive and make it publicly viewable and searchable. The COAL-FO (COAL Follow On) is a continuation of a previous project completed in the 2016-2017 capstone class. The COAL project was a suite of algorithms that identified, classified, and quantified the effects of open-pit mining on the surrounding environment. Our project will also aim at improving the existing algorithms that were developed in the COAL project and create a new baseline suite to rank changes to areas of land over time.

2. Overall Description

2.1 Product Perspective

The product will be taken from the coal-capstone github organization and the existing code will be exported to the XSEDE environment, and used to perform analysis on existing spectro imagery. We will improve the imagery processing algorithms currently in use. Then we will create a new baseline suite to rank changes to areas of land over time to determine if the changes in one particular area our significant or not. The main priority will be to create a searchable port for the project which will use the XSEDE environment.

2.2 Product Functions

The product will be non functional upon completion, unless further data is generated that requires analysis. The goal will be to create an archive. A stretch goal would be to create a way to search that archive, in which case the functions for that would be a search function and a visual way to represent data.

2.3 User Characteristics

The user will be able to access this archived dataset and access processed data to see damage from coal mining in various part of the country. If the stretch goal is reached, then the user will be able to save time by searching a database to get the same information without having to download it first.

2.4 Constraints

The XSEDE grant only allowed to 2 terabytes of storage and a certain number of clock hours. To counteract this constraint, we will need to develop a system for rotating in imagery that needs to be processed and out data that's already been processed. We will also need to make sure our algorithms are as efficient as they can possibly be to not waste resources so we can use maximize the use of the 2 terabytes provided to us.

In addition, we are limited by the AVIRIS data that is available that, while large, barely covers a fraction of the surface of the United States. This data is focused primarily on flight lines and areas where coal and open pit surface mining is currently being done or areas which have undergone coal and open pit surface mining in the past.

2.5 Assumptions and Dependencies

This project requires the existing COAL algorithms, the XSEDE environment for rotating in imagery, the data provided to use by AVIRIS and Apache.

3. Specific Requirements

A completed archive, and tools that are exportable to XSEDE. The requirements of our project as discussed with our client is to to create a searchable port for the existing COAL project. Specifically, this will be augmented to accommodate the desire to (i) port the coal-sds software to the XSEDE platform and undertake test and evaluation of the system performance, (ii) process all AVIRIS and AVIRIS-NG imagery generating and archiving all science data products, and (iii) making the products searchable through a portal.

2.2 Final Requirements Document - Client approved

CS 461 Requirements

Team Name: COAL-FO

Bryce Egley, Kenny Thompson

1. Introduction

1.1 Purpose

The purpose for this document is to layout the requirements for the COAL-FO (COAL Follow On) capstone project and describe how these requirements will be accomplished.

The purpose of our COAL-FO project is to improve pycoal algorithms, enable more spectral libraries to work with pycoal, to collect more data to process with pycoal and to improve general functionality of pycoal.

1.2 Scope

The project will be focused around publicly available COAL algorithms created by the previous capstone group, publicly available spectral analysis collected from the NASA AVIRIS project. Work for this project will be stored in the Capstone-Coal organization on github and version control as well as future releases for this project will be managed within that organization.

1.3 Definitions, Acronyms, Abbreviations

COAL-FO: Coal and Open-pit surface mining impacts on American Lands Follow-On

AVIRIS: Airborne Visible / Infrared Imaging Spectrometer

USGS: United States Geological Survey

EcoSIS: Ecological Spectral Information System

EcoStress: ECOsystem Spaceborne Thermal Radiometer Experiment on Space Station

ASTER: Advanced Spaceborne Thermal Emission and Reflection Radiometer

1.4 Overview

This project will take existing algorithms, improve them, and test the ability to work on other systems, as well as improving the overall 'toolkit' of pycoal for the end user. The COAL-FO is a continuation of a previous project completed in the 2016-2017 capstone class. The COAL project was a stable python toolkit providing examples, tests, packages, stable release and stable API that identified, classified, and quantified the effects of open-pit mining on the surrounding environment. Which also included COAL-SDS (Coal Science Data System). Our project will aim at improving the existing algorithms and general functionality as well as enabling the toolkit to work with more spectral libraries.

2. Overall Description

2.1 Product Perspective

The product will be taken from the coal-capstone github organization and the existing code can potentially be exported to other environments, and used to perform analysis on existing spectral imagery. We will improve the imagery processing algorithms currently in use. We will allow the product to identify more spectra in other spectral libraries such as EcoSIS, EcoStress and USGS Spectral Library Version 7.

2.2 Product Functions

The product will be functional upon completion, and used on new data that requires analysis. It will be improved pycoal algorithms, general improvements, bug fixes and COAL-FO will also focus on enabling our classifications to use the USGS Spectral Library Version 7, EcoSIS Spectral Library and the EcoStress Spectral Library. With these additional spectral libraries we will be able to identify more minerals in the AVIRIS images we run mineral classifications on.

2.3 User Characteristics

The user will be able to access this new and improved code from github, and will be able to run it on their own resources that they have available.
The user will also be able to use the convert functions we will create in mineral.py of pycoal to generate convolved library files in envi .hdr and .sli format from USGS Spectral Library Version 7, EcoSIS Spectral Library and the EcoStress Spectral Library.

2.4 Constraints

AVIRIS data that is available is focused primarily on flight lines while we want to examine areas where coal and open pit surface mining is currently being done or areas which have undergone coal and open pit surface mining in the past. USGS Spectral Library Version 6 the current spectral library pycoal is using, was created over a decade ago in 2007. Reading through the documentation they provided it is unclear how they generated the convolved library for this spectral library. And many of the emails they list to send questions to regarding USGS Spectral Library Version 6 are people who are no longer work for USGS or do not know how the convolved libraries were created based on our attempts to contact USGS.

2.5 Assumptions and Dependencies

This project is dependent on the existing COAL algorithms, the data provided by AVIRIS and Spectral Libraries such USGS Spectral Library Version 7, EcoSIS Spectral Library, ASTER Spectral Library and the EcoStress Spectral Library.

Pycoal depends on these spectral libraries being in the correct convolved library format with band wavelengths from 0.37 to 2.5 microns (micrometers) to match that of the AVIRIS images we are analyzing. Pycoal also relies on these spectral libraries mentioned above to provide all the necessary data such as reflectance values, wavelength values, spectra name for generation of the convolved libraries in the .sli and .hdr file format.

3. Specific Requirements

The specific requirements of our project will be to improve pycoal algorithms, fix and find bugs existing in pycoal, make general improvements and enable pycoal to work with more spectral libraries specifically USGS Spectral Library Version 7, EcoSIS Spectral Library and the EcoStress Spectral Library. More details on how we intended to implement each of these requirements is provided below.

USGS Spectral Library Version 7 was release in 2017 and is the newest released version of the USGS(United States Geological Survey) spectral library. USGS Spectral Library Version 6 the prior version was released over a decade ago in 2007. Pycoal doesnt use the entire spectral library instead it consumes a convolved library file of the Spectral Library. For USGS Spectral Library Version 6, USGS staged the convolved library files on their website and the COAL project was able to use these to identify spectra. The problem COAL-FO faces is that much of the documentation used to do this is over a decade old, the spectral libraries are in completely different formats and it doesnt appear as though USGS has plans to stage the convolved library files for USGS Spectral Library Version 7.

The COAL-FO project will also aim at using spectra from the EcoSIS and EcoStress Spectral Libraries. Enabling pycoal to work with more spectral libraries will allow us to identify more minerals in the AVIRIS images. The EcoStress Spectral Library Version 1.0 is the spectral library which is built upon the ASTER Spectral Library Version 2.0. Building the convolved library files for this spectral library may not be so difficult since we already have a conversion for the ASTER spectral library which EcoStress is adapted from. EcoSIS Spectral Library has 62k spectra which will be totally new to us. With both EcoStress and USGS Spectral Library Version 7 we are currently using previous versions of these spectral libraries so many of these spectra will have already been identified in previous products we generated with pycoal.

The specific goal for each one of these Spectral Libraries is to create a convert class in mineral.py or else where in the pycoal subdirectory which will convert these spectral libraries to their convolved library file formats. These formats of the convolved libraries are specifically .sli and .hdr file formats.

A specific goal for completion will also be the implementation of 'classifier callback,' or the ability to control the classification algorithm used. Right now we utilize the spectral angle mapper for determining mineral classification, and we would like to be able to use other techniques in the future, since different options serve different purposes. Implementing the classifier callback will give the end user increased flexibility when using pycoal. This is important for follow on projects, and follow on work is already scheduled that is depending on the completion of this goal, so it is important for the final product.

A stretch goal will be for the COAL-FO project to crop the default example image. Our client has said this task may take a long time to complete. Therefor, this requirement will be left as a stretch goal if and when the requirements for the spectral libraries and classifier callback mentioned above have been completed, tested and verified to be working correctly.

Currently, it takes approximately one to two days to run a mineral classification on the default example. The default example is 17.5 gb. By cropping it down to only contain the main areas of interest in this case the flow lines. We could ideally get it down to a size that would only take a few hours to run. This would be very beneficial to new users of pycoal and would help us a lot with testing. We have found AVIRIS images that are only 1-2 gb in size, these images typically only take a few hours to run on pycoal, however most of these smaller sized images do not contain lots of mineral and mining activity.

4. References

[1] COAL-FO, Coal and Open-pit surface mining impacts on American Lands Follow-On
https://capstone-coal.github.io/team

[2] AVIRIS, Airborne Visible/Infrared Imaging Spectrometer
https://aviris.jpl.nasa.gov/

[3] AVIRIS-NG, airborne Visible/Infrared Imaging Spectrometer Next Generation
https://aviris-ng.jpl.nasa.gov/

[4] COAL, Coal and Open-pit surface mining impacts on American Lands
https://capstone-coal.github.io/

[5] HPC, High Performance Computing
https://en.wikipedia.org/wiki/Supercomputer

[6] USGS Spectral Library Version 7
https://en.wikipedia.org/wiki/Supercomputer

[7] EcoSIS Spectral Library

https://ecosis.org/

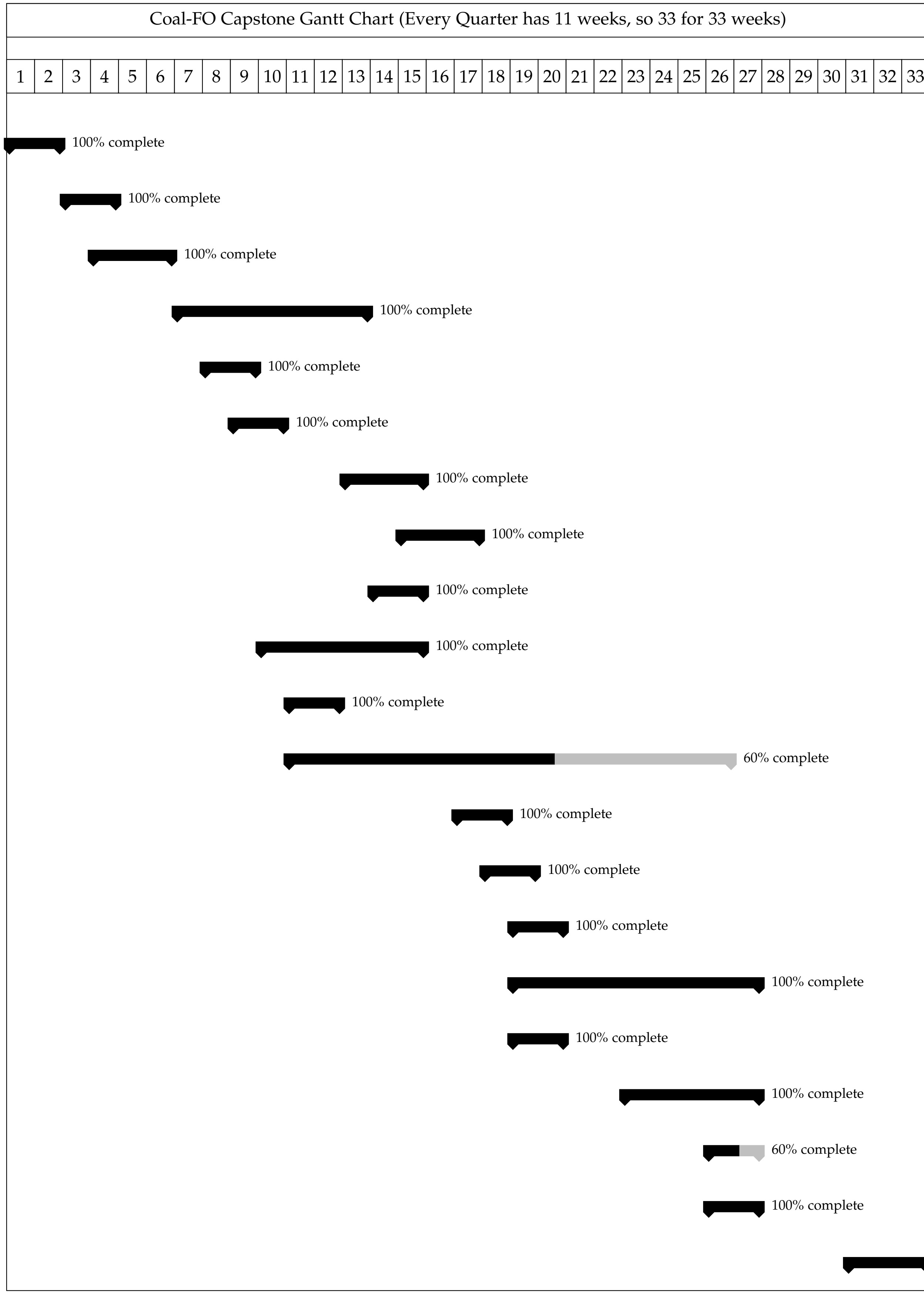
[8] EcoSTRESS Spectral Library

https://speclib.jpl.nasa.gov/

[9] Aster Spectral Library Version 2.0

https://speclib.jpl.nasa.gov/downloads/2009-Baldridge.pdf

2.3 Final Gantt Chart



CS 461 The Preliminary Design Document

Bryce Egley and Kenny Thompson

Group Number: 28

Project Name: COAL-FO

Our Roles: Developers

3 Introduction

COAL and Open-pit surface mining impacts on American Lands Follow-On (COAL-FO) is the successor project to the 2016- 2017 COAL project. COAL initially aimed to deliver a suite of algorithms to identify, classify, characterize, and quantify (by reporting a number of key metrics) the direct and indirect impacts of mining operations and related destructive surface mining activities across the continental U.S (and further afield). COAL successfully delivered a Python library for processing hyperspectral imagery from remote sensing devices such as the Airborne Visible/InfraRed Imaging Spectrometer AVIRIS and a Science Data System for running COAL pipelines. COAL-FO will utilize recent funding obtained from a recently awarded NSF-funded XSEDE high performance computing (HPC) grant to further improve, validate and document COAL algorithms, execution runtime performance and geospatial output results.[1]

4 Body

4.1 HPCC High Performance Computing Cluster

4.1.1 Overview

This section will cover the specification of the High Performance Computing Cluster we will be using to make the data easier to view and less costly to users. Just one of the AVIRIS images can be as large as 18 gigabytes some even larger. Since it takes a lot of resources to store these images we wish to store the AVIRIS images on a High Performance Computing Cluster. For the High Performance Computing Cluster we have chosen to use XSEDE the Extreme Science and Engineering Discovery Environment.

4.1.2 Design Concerns

This data is focused primarily on flight lines and areas where coal and open pit surface mining is currently being done or areas which have undergone coal and open pit surface mining in the past. Our goal is to set up an HPCC system which can process and store the AVIRIS images. XSEDE will give us 2 terabytes of storage to do this.

4.1.3 Design Elements

4.1.3.1 Compatibility: We want to move our processing pipeline to XSEDE since hyperspectral imagery provided by AVIRIS takes up huge amounts of storage and this would increase the available memory needed to load and process imagery making the COAL project much more efficient and will improve it's ability to process more hyperspectral imagery.

4.1.3.2 Reliability: This is where we will test that our COAL-SDS correctly code runs and performs well on XSEDE. The code for COAL-SDS is already mostly developed and we will continue to finish it and the test its functionality with XSEDE.

4.2 License

4.2.1 Overview

This section will cover the specifications of the Licenses we will be using maintain ownership and credit for our code but also keep it open source and available for other research projects. We want a license that is free and easy to use. We also wish to have a license that is well known amongst the open source community.

4.2.2 Design Concerns

We need a license for our code that wont require too much change to the existing code and wont be heavy in documentation. A lot of our current code for the python toolkit is protected under the GNU GPLv2 and a lot of our current code on COAL-SDS is protected under the Apache License v2.0. So, we want to use these licenses.

4.2.3 Design Elements

4.2.3.1 Compatibility: Need to use a license that other developers will be familiar with and comfortable with since we want to keep all our code open source. GNU GPLv2 and Apache License v2.0 are both excellent choices for this since they are the 2nd and 3rd most common open source licenses after the MIT license.

4.2.3.2 Reliability: Need to make sure all of our code follows the licensing guidelines and fair use regulations. We shouldn't have to do too much reliability testing for licensing though.

4.3 COAL Cataloguing and Archiving System (CAS)

4.3.1 Overview

We wish to have a way to store AVIRIS imagery so that we can loop images through XSEDE and store other data for our COAL-FO project on. We also wish to have all the data for COAL-FO be managed in one place. This will make it easier to design and organize our project. Apache OODT-powered Science Data System can fill these concerns as well as COAL-SDS which is built on top of Apache OODT.

4.3.2 Design Concerns

We need a common cataloging and archiving system that is compatible with the previous code written for COAL and that will work with XSEDE and AVIRIS imagery. Our main concern is that there could be problems storing AVIRIS imagery on XSEDE.

4.3.3 Design Elements

4.3.3.1 Compatibility: Need to make sure that Apache OODT-powered Science Data System will be compatible with AVIRIS imagery and work well together with XSEDE.

4.3.3.2 Reliability: Need a common cataloging and archiving system that we are all familiar with and will work for the data we are working with. Apache OODT-powered Science Data System will be used as our common cataloging and archiving system. To ensure reliability we will test COAL-SDS, which is built on top of Apache OODT and make sure it works with AVIRIS imagery and XSEDE.

4.4 Export/Import tool view

4.4.1 Overview

This section will cover the specifications of the export/import tool we will be using to take data in from the AVRIS databases and out to the XSEDE data systems. It will also be the tool we use to take data out of the XSEDE data systems and onto our database we will be creating to store this data.

4.4.2 Design Concerns

The design concern will be to make sure that our tool, in this case the GLOBUS command line system, will be compatible with AVRIS, XSEDE, and the database we create to store and present this data.

4.4.3 Design Elements

4.4.3.1 Compatibility: This is where we will ensure that the tool works with the pipeline, from reading it on the AVRIS systems to storing the end result on our database.

4.4.3.2 Reliability: This is where we test our export/import tool and ensure that it performs correctly when taking data in from the AVIRIS databases and out to XSEDE.

4.5 Programming language View

4.5.1 Overview

This section will cover the specifications of the programming language we will be using to conduct our spectro-image analysis of the chosen area. Python was chosen because of its design benefits and the ease of use for a project like this. It also required less work since pycoal is already in python.

4.5.2 Design Concerns

Design concerns: It is important that the code be simple, streamlined, and bug free. Since we are going to be running this code for a long time, having to stop and start over due to a bug or a conflict between parts would be less than ideal. Whats more, its also important that the code we use be portable, as in, we can port it to work on the HPC through our XSEDE grant.

4.5.3 Design Elements

4.5.3.1 Portability: This is where we ensure the programming language will be portable to work on the HPC.

4.5.3.2 Reliability: This is where we ensure that the programming language we choose is reliable and will not cause us to lose significant amounts of progress.

4.6 Imagery view

4.6.1 Overview

This section will cover the specifications of the imagery data. We will be using AVRIS image data, since it is already found in an easy to process form and its the type of image data that our program was specifically designed to deal with. AVRIS has large public databases of image files that we will take, process, and produce data that is easy for researchers to process and draw conclusions from.

4.6.2 Design Concerns

It is important that this imagery be processed uniformly, since the end goal will be to present them in a database for researchers of the future.

4.6.3 Design Elements

4.6.3.1 Image analysis: This is where we ensure that the image we feed in and the image we get out are correct and in a consistent form.

4.6.3.2 Consistency: This is where we ensure the data we feed in and the data we pull out are both presented in a consistent form that is of academic use to further researchers.

5 Conclusion

In this document, we covered the design for six elements we will be using to implement and design our COAL-FO capstone project. We discussed the design concerns and design elements such as reliability and compatibility. When we start implementation of the project we may change design choices. We will try to stick to the design choices we made here in this document as much as possible. There are many other design technologies and features we will need to think about and develop over the course of the COAL-FO project. However for now we have managed to come up with a design process for six elements which will be key to completing the COAL-FO project. For other design features in the future we will refer back to this process to determine how we will complete them.

References

[1] CS461 - CS Senior Capstone, Eecs.oregonstate.edu, 2017. [Online]. Available: http://eecs.oregonstate.edu/capstone/cs/capstone.cgi?project=320 [Accessed: 22- Nov- 2017]

[2] USGS.gov Science for a changing world, Usgs.gov, 2017. [Online]. Available: https://www.usgs.gov/ [Accessed: 22- Nov- 2017]

[3]]AVIRIS - Airborne Visible / Infrared Imaging Spectrometer, Aviris.jpl.nasa.gov, 2017. [Online]. Available: https://aviris.jpl.nasa.gov/ [Accessed: 22- Nov- 2017].

[4] XSEDE User Portal Globus User Guide, Portal.xsede.org, 2017. [Online]. Available: https://portal.xsede.org/software/globus [Accessed: 22- Nov- 2017].

[5] C++, En.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/C [Accessed: 22- Nov- 2017].

[6] ”MySQL”, En.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/MySQL [Accessed: 22- Nov- 2017].

[7] ”XSEDE, Extreme Science and Engineering Discovery Environment”, www.xsede.org, 2017. [Online]. Available: https://www.xsede.org/ [Accessed: 22- Nov- 2017].

[8] ”AVIRIS, Airborne Visible/Infrared Imaging Spectrometer”, aviris.jpl.nasa.gov, 2017. [Online]. Available: https://aviris.jpl.nasa.gov/ [Accessed: 22- Nov- 2017].

[9] ”AVIRIS-NG, airborne Visible/Infrared Imaging Spectrometer Next Generation”, aviris-ng.jpl.nasa.gov, 2017. [Online]. Available: https://aviris-ng.jpl.nasa.gov/[Accessed: 22- Nov- 2017].

[10] ”COAL, Coal and Open-pit surface mining impacts on American Lands”, capstone-coal.github.io, 2017. [Online]. Available: https://capstone-coal.github.io/ [Accessed: 22- Nov- 2017].

[11] ”HPC, High Performance Computing”, en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Supercomputer [Accessed: 22- Nov- 2017].

[12] ”Query Language”, en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Query_language [Accessed: 22- Nov- 2017].

[13] ”Enterprise Module Service”, hpccsystems.com, 2017. [Online]. Available: https://hpccsystems.com/enterprise-services/modules/esp [Accessed: 22- Nov- 2017].

[14] ”HPCC High Performance Computing Cluster”, hpccsystems.com, 2017. [Online]. Available: https://hpccsystems.com/enterprise-services/modules/esp [Accessed: 22- Nov- 2017].

[15] ”OSU Unix HPC Cluster”, cosine.oregonstate.edu, 2017. [Online]. Available http://cosine.oregonstate.edu/unix-hpc-cluster [Accessed: 22- Nov- 2017].

[16] ”MIT License”, en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/MIT_License [Accessed: 29- Nov- 2017].

[17] ”Apache License Version 2.0”, www.apache.org, 2017. [Online]. Available https://www.apache.org/licenses/LICENSE-2.0 [Accessed: 29- Nov- 2017].

[18] ”GNU General Public License, version 2”, www.gnu.org, 2017. [Online]. Available https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html [Accessed: 29- Nov- 2017].

[19] ”Top Open Source Licenses”,www.blackducksoftware.com, 2017.[Online]. Available https://www.blackducksoftware.com/top-open-source-licenses [Accessed: 29- Nov- 2017].

5.1 What design aspects were changed, deleted or added? Why?

This was the final version that was accepted by our client. The only requested changes by the client on this document were styling changes.

6 PART3: TECH REVIEW

CS 461 Technology Review and Implementation Plan

Bryce Egley and Kenny Thompson

Group Number: 28

Project Name: COAL-FO

Our Roles: Developers

Abstract

COAL and Open-pit surface mining impacts on American Lands Follow-On (COAL-FO) is the successor project to the 2016-2017 COAL project. COAL initially aimed to deliver a suite of algorithms to identify, classify, characterize, and quantify (by reporting a number of key metrics) the direct and indirect impacts of mining operations and related destructive surface mining activities across the continental U.S (and further afield). COAL successfully delivered a Python library for processing hyperspectral imagery from remote sensing devices such as the Airborne Visible/InfraRed Imaging Spectrometer AVIRIS and a Science Data System for running COAL pipelines. COAL-FO will utilize recent funding obtained from a recently awarded NSF-funded XSEDE high performance computing (HPC) grant to further improve, validate and document COAL algorithms, execution runtime performance and geospatial output results.[1]

7 INTRODUCTION

7.1 What we are trying to accomplish

The purpose of our COAL-FO project as discussed in our last meeting with our client is to create a searchable portal for the existing COAL project. The project will be focused on (i) utilizing existing open source software e.g. pycoal and coal-sds, created by the previous capstone group, (ii) publicly available airborne hyperspectral imagery data collected from the NASA AVIRIS and AVIRIS-NG projects, and (iii) the XSEDE HPC environment.

The project will be focused around publicly available COAL algorithms created by the previous capstone group, publicly available spectral analysis collected from the NASA AVIRIS project, and a grant on the XSEDE environment to use HPC. Work for this project will be stored in the Capstone-Coal organization on github and version control as well as future releases for this project will managed within that organization.

7.2 Project Overview

This project will, to put it in rough terms, take existing algorithms, export them to the XSEDE environment, a high-powered computing cluster and process as much publicly available spectro imagery as possible, and use that to create an archive. A stretch goal will be to take that archive and make it publicly viewable and searchable. The COAL-FO (COAL Follow On) is a continuation of a previous project completed in the 2016-2017 capstone class. The COAL project was a suite of algorithms that identified, classified, and quantified the effects of open-pit mining on the surrounding environment. Our project will also aim at improving the existing algorithms that were developed in the COAL project and create a new baseline suite to rank changes to areas of land over time.

The product will be taken from the coal-capstone GitHub organization and the existing code will be exported to the XSEDE environment, and used to perform analysis on existing spectro imagery. We will improve the imagery processing algorithms currently in use. Then we will create a new baseline suite to rank changes to areas of land over time to determine if the changes in one area our significant or not. The main priority will be to create a searchable port for the project which will use the XSEDE environment.

8 TECHNOLOGIES

In this paper we will discuss 6 different technologies we will be using for our COAL-FO project. We explore three different options for each of these technologies, even if we already have decided on which option we are going to use.

8.1 HPCC High Performance Computing Cluster

8.1.1 Options

I Enterprise Service Platform

II OSU Unix HPC Cluster

III XSEDE

8.1.2 Goals

Our goals for this portion of the project is too make the data easier to view and less costly to users. We are limited by the AVIRIS data that is available that, while large, barely covers a fraction of the surface of the United States. This data is focused primarily on flight lines and areas where coal and open pit surface mining is currently being done or areas which have undergone coal and open pit surface mining in the past. Our goal is to set up an HPCC system which can process and store the AVIRIS images.

8.1.3 Criteria

We want an HPCC to store AVIRIS images so users dont need to use their own personal computing resources to view and collect data from the AVIRIS images. The system will need to have several terabytes of storage and have a low cost.

8.1.4 Comparison

Out of the three options for HPCC discussed so far only XSESE has been already paid for and we were allocated 2 terabytes of storage. We may be able to use OSU Unix HPC Center as part of our tuition but with XSEDE we would have a more permanent place to store data since we will graduate and our storage through an OSU system would expire upon graduation. Enterprise Service Platform is a good HPCC however it will cost us money to use and we have already been given a grant to use XSEDE.

8.1.5 Discussion

XSEDE is the Extreme Science and Engineering Discovery Environment. XSEDE will allow us to use 2 Tera-bytes for processing images. We will loop images through XSEDE and have them stored on another database for other users to then access them on the XSEDE platform. This will save other users a lot of time since they wont have to deal with generating the images each time and will just have the images ready to go on XSEDE. We hope this will make the COAL capstone project from the previous year and continuation with our project this year can reach a wider audience and be used in more research for people studying the effects of coal on American lands. Putting our program on the XSEDE platform is the main objective of our project as outlined by our client. Our group has been given a grant to do this and the 2 terabytes on XSEDE for now, although we could be given more money and space in the future.

8.1.6 Selection

XSESDE - Out of the three options for HPCC discussed so far only XSESE has already been paid for and we were allocated 2 terabytes of storage.

8.2 Common Cataloging and archiving system

8.2.1 Options

I Apache OODT-powered Science Data System

II FireHold

III Widen Collective

8.2.2 Goals

We wish to have a way to store AVIRIS imagery so that we can loop images through XSEDE and store other data for our COAL-FO project on. We also wish to have all the data for COAL-FO be managed in one place. This will make it easier to design and organize our project. Apache OODT-powered Science Data System can fill these concerns as well as COAL-SDS.

8.2.3 Criteria

We need a common cataloging and archiving system that is compatible with the previous code written for COAL and that will work with XSEDE and AVIRIS imagery. Our main concern is that there could be problems archiving AVIRIS imagery and corrupting are losing data.

8.2.4 Comparison

We want to make sure that Apache OODT-powered Science Data System will be compatible with AVIRIS imagery and work well together with XSEDE. FireHold is a good archiving system however it wouldn't necessarily work with the AVIRIS data. [22] Widen Collective is also a good archiving system however it wouldn't necessarily work with AVIRIS data. [23]

8.2.5 Discussion

We want a common cataloging and archiving system that we are all familiar with and will work for the data we are working with.

8.2.6 Selection

Apache OODT-powered Science Data System - Coal-SDS is existing code for our capstone project which uses Apache OODT-powered Science Data System so we will use this as our common cataloging and archiving system. [21]

8.3 License

8.3.1 Options

I MIT License

II GNU GPLv2

III Apache License v2.0

8.3.2 Goals

Our goal is to have a license that will allow us to maintain ownership and credit for our code but also keep it open source and available for other research projects. We want a license that is free and easy to use.

8.3.3 Criteria

We need a license for our code that won't require too much change to the existing code and won't be heavy in documentation. A lot of our current code uses the GNU GPLv2 and Apache License v2.0. So, we would like to use these licenses or licenses very similar to them.

8.3.4 Comparison

The MIT License is the most popular software license as of 2015[19]. This gives the MIT License an advantage over other open source licenses we may consider using, since it would give us good experience using an open source license which we will probably use again in the future. It should also meet most of our criteria since a license that is very popular will likely be easy to use and do a good job at allowing us to maintain ownership of our code. The MIT License is also compatible with GNU which is good since some of the existing code for our project is already under the GNU GPLv2 License.

GNU 2.0 General Purpose 2.0 License is very easy to share and use. It also the second most common open source license as of 2015[19]. This like the MIT License would give us good experience since it is very likely that we will use it in the future.

Apache License v2.0 shares the same benefits as the two other open source licenses discussed previously. It is the third most common open source licenses as of 2015[19].

8.3.5 Discussion

Since our existing code is already covered under the GNU GPLv2 for pycoal and Apache License v2.0 for coal-sds we will continue to use these open source licenses on the code we develop for our capstone project.

8.3.6 Selection

Apache License v2.0 and GNU GPLv2 - GNU 2.0 General Purpose 2.0 License is very easy to share and use. It also the second most common open source license as of 2015[19]. Apache License v2.0 shares the same benefits as the two other open source licenses discussed previously. It is the third most common open source licenses as of 2015[19].

8.4 Imagery

8.4.1 Options

I Google Maps

II AVIRIS

III US geological survey data

8.4.2 Goals

The goal for this section is to process as much publicly available data as possible.

8.4.3 Criteria

The data we are going to read will need to be able to detect damage from coal mining (the presence of heavy metals in the rivers and surrounding areas, etc) and must be conducted in areas where coal mining has occurred, and also in areas where coal mining hasnt occurred, so we can get a baseline.

8.4.4 Comparison

AVIRIS was what the client requested, but there are pros and cons for using it. For one thing, it only covers certain sections of land, a fraction of the US landmass. It also isn't ideal for the stated goal, which is analyzing coal mining damage. Since it only covers certain areas, we arent getting a complete picture of coal mining. The pros are, the technology already developed works well with AVIRIS, since it was specifically designed to analyze spectro-imagery.

This meets the first criteria well, but is not great on the second criteria, because as earlier stated, it is very limited in the area it covers. While not ideal, this will still work because on the area it did cover we can extrapolate to make assumptions about other areas where that type of mining was conducted.[3] A possible alternative would be google maps. Google maps includes all imagery nationwide, although some of it is less current and less high resolution than others. Google earth also gives us a host of other data that other sources do not have, and its possible we could use all this data to extrapolate coal damage in all sorts of ways. For example, Google maps shows the activity of foot traffic in certain areas, and what time of day that foot traffic is highest. This certainly meets our second criteria in spades. Not to mention Google is already in a searchable easy to manipulate database. Unfortunately, google maps does not include spectro analysis, and would not work with the technology we have developed. The quality of photography just isnt enough to determine damage from things like spillage and contamination. So while you could certainly extrapolate using other methods, it will likely be less high quality of analysis compared to actual imagery on the ground. Another possibility is US geological survey data. This is data that is published online from the USGS and thanks to government funding is far reaching. This source includes a lot of data not included in either AVIRIS or google maps, and has live feeds coming from all across the US. Its free to use and trustworthy, and has access to resources and datasets that no one else can match, it even has datasets of spectro imagery so you can figure out what you are looking at. It does not meet either criteria well, since its not all in the same easy to process format as AVIRIS and Google maps it would be harder to make simple code that can analyze all of it at once. [2]

8.4.5 Discussion

We will be using AVIRIS. Since the data is in the format we already want, and its free and publicly accessible, we will be able to use it to meet both criteria. We will also be making use of some of the datasets in the USGS database so we can determine what we are looking at.

8.4.6 Selection

AVIRIS - We will be using AVIRIS. Since the data is in the format we already want, and its free and publicly accessible, we will be able to use it to meet both criteria.

8.5 Export/Import Tool

8.5.1 Options

I Globus Command Line Interface

II Regular Globus

III Globus url copy and uberftp

8.5.2 Goals

For this project, we will be using a high powered computing cluster. We will need a tool to import data from our image analysis to the server, and a tool to export the processed data onto a computer.

8.5.3 Criteria

Only a certain amount of data can be stored, but we cant stop every time it fills up to manually run it, so we will need this tool to run automatically so things can run smoothly.

8.5.4 Comparison

The Globus command line interface is a tool to export and import data out of XSEDE. Globus command line interface is a little harder to use than some other ones, it requires a ssh key and some additional setup procedures, as well as just advanced knowledge on databases to begin with. However, its easier to automate, and since we will be dealing with massive amounts of data, automation is a good thing. [4] A possible alternative would be regular Globus. Globus is easier to use, has a web interface, can be set up to use just a simple login, and there is even a desktop download to run it from your computer. The downside is its less easy to automate so a lot of the work would have to be done manually, which when we are dealing with this amount of data tends to be a problem unless you want to manually move everything over constantly. Another possible alternative would be globus url copy and uberftp. This has the same drawbacks as globus command line interface, in that its harder to use, requires advanced knowledge, and requires additional authentication procedures, however this is more designed for high performance jobs, and is more efficient than the other two.

8.5.5 Discussion

XSEDE will provide the globus service for file transfer and sharing. Globus will allow us to securely share our date with other people using COAL. Globus is easier to use, has a web interface, can be set up to use just a simple login, and there is even a desktop download to run it from your computer.

8.5.6 Selection

Regular Globus - XSEDE will provide the globus service for file transfer and sharing. Globus will allow us to securely share our date with other people using COAL. [20]

8.6 Programming Language

8.6.1 Options

I Python

II C++

III Java

8.6.2 Goals

The goal for this section is to have code that scans spectro-imagery and determines damage.

8.6.3 Criteria

We used python for the reason that the program was already set up in python and thats what the client wanted, but python is still the correct tool for this scenario because it is a general purpose language that can be used to both take in the imagery, run analysis on it, and pump out readable results. This might require 2 or even 3 programming languages if you tried to do it without python.

8.6.4 Comparison

Another possibility is C++. C++ would be a good option because its easier to run on XSEDE and works well with GLOBUS, its also a program more people are familiar with. However its limited flexibility would require another programming language or some 3rd party tool to convert spectro-imagery into a format its able to read.[5] Java would also be a good possibility, because it works well with databases, and might make it easier to export/import data from other sources. However the tooling would require significant work to take the functions we have on Python and make them work on Java, and even then it might not work as well as the existing applications.

8.6.5 Discussion

We will be using python for the reasons stated above. We will try to convert the pycoal code to cython so we can get C like run time making our code faster since speed is a major issue in the project. As for coal-sds code java will remain as the programming language for that portion of our project. Java works well with databases, and might make it easier to export/import data from other sources.

8.6.6 Selection

Python, Cython and Java - Python is already used in the pycoal code which was left behind by the COAL project. We do hope to convert the python code to cython so that we can have C like run time. Java was used for the coal-sds code so Java will remain in the coal-sds portion of our project.

9 Conclusion

In this document, we covered six different technologies we will be using to implement and design our COAL-FO capstone project. We discussed pros and cons of different pieces for each technology we would be using. When we start implementation of the project we may change design choices as certain pros and cons may not have been thought of and adequately represented in our planning. We hope we can in general stick to the technology choices we made in this document.

References

[1] CS461 - CS Senior Capstone, Eecs.oregonstate.edu, 2017. [Online]. Available: http://eecs.oregonstate.edu/capstone/cs/capstone.cgi?project=320 [Accessed: 22- Nov- 2017]

[2] USGS.gov Science for a changing world, Usgs.gov, 2017. [Online]. Available: https://www.usgs.gov/ [Accessed: 22- Nov- 2017]

[3]]AVIRIS - Airborne Visible / Infrared Imaging Spectrometer, Aviris.jpl.nasa.gov, 2017. [Online]. Available: https://aviris.jpl.nasa.gov/ [Accessed: 22- Nov- 2017].

[4] XSEDE User Portal Globus User Guide, Portal.xsede.org, 2017. [Online]. Available: https://portal.xsede.org/software/globus [Accessed: 22- Nov- 2017].

[5] C++, En.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/C [Accessed: 22- Nov- 2017].

[6] "MySQL", En.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/MySQL [Accessed: 22- Nov- 2017].

[7] "XSEDE, Extreme Science and Engineering Discovery Environment", www.xsede.org, 2017. [Online]. Available: https://www.xsede.org/ [Accessed: 22- Nov- 2017].

[8] "AVIRIS, Airborne Visible/Infrared Imaging Spectrometer", aviris.jpl.nasa.gov, 2017. [Online]. Available: https://aviris.jpl.nasa.gov/ [Accessed: 22- Nov- 2017].

[9] "AVIRIS-NG, airborne Visible/Infrared Imaging Spectrometer Next Generation", aviris-ng.jpl.nasa.gov, 2017. [Online]. Available: https://aviris-ng.jpl.nasa.gov/[Accessed: 22- Nov- 2017].

[10] "COAL, Coal and Open-pit surface mining impacts on American Lands", capstone-coal.github.io, 2017. [Online]. Available: https://capstone-coal.github.io/ [Accessed: 22- Nov- 2017].

[11] "HPC, High Performance Computing", en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Supercomputer [Accessed: 22- Nov- 2017].

[12] "Query Language", en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Query_language [Accessed: 22- Nov- 2017].

[13] ”Enterprise Module Service”, hpccsystems.com, 2017. [Online]. Available: https://hpccsystems.com/enterprise-services/modules/esp [Accessed: 22- Nov- 2017].

[14] ”HPCC High Performance Computing Cluster”, hpccsystems.com, 2017. [Online]. Available: https://hpccsystems.com/enterprise-services/modules/esp [Accessed: 22- Nov- 2017].

[15] ”OSU Unix HPC Cluster”, cosine.oregonstate.edu, 2017. [Online]. Available http://cosine.oregonstate.edu/unix-hpc-cluster [Accessed: 22- Nov- 2017].

[16] ”MIT License”, en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/MIT_License [Accessed: 29- Nov- 2017].

[17] ”Apache License Version 2.0”, www.apache.org, 2017. [Online]. Available https://www.apache.org/licenses/LICENSE-2.0 [Accessed: 29- Nov- 2017].

[18] ”GNU General Public License, version 2”, www.gnu.org, 2017. [Online]. Available https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html [Accessed: 29- Nov- 2017].

[19] ”Top Open Source Licenses”,www.blackducksoftware.com, 2017.[Online]. Available https://www.blackducksoftware.com/top-open-source-licenses [Accessed: 29- Nov- 2017].

[20] ”Globus at XSEDE”, portal.xsede.org, 2017. [Online]. Available https://portal.xsede.org/software/globus [Accessed: 13- Dec- 2017].

[21] ”Apache OODT”, cwiki.apache.org, 2017. [Online]. Available https://cwiki.apache.org/confluence/display/OODT/Home [Accessed: 13- Dec- 2017].

[22] ”FireHold”, unbouncepages.com, 2017. [Online]. Available http://unbouncepages.com/filehold-document-management-software/ [Accessed: 13- Dec- 2017].

[23] ”Widen Collective, www.capterra.com, 2017. [Online]. Available https://www.capterra.com/p/127696/Widen-Collective/ [Accessed: 13- Dec- 2017].

10 PART 4: WEEKLY BLOGS

11 FALL TERM, BRYCE EGLEY

11.1 Week 1

11.1.0.1 Plans: This week I plan to choose my project preferences.

11.1.0.2 Problems: After reviewing the project preferences list there were several projects I didn’t think I would have the required knowledge for.

11.1.0.3 Progress: I have made my selection on my project preferences.

11.1.0.4 Summary: For week 1 I selected my project preferences. My preferences selection was as follows 1. Stock Analytics Website 2. Prison Match 3. RPM (Release Point Maximizer) 4. Campus Events Mobile App 5. Modernize Email List Serve I choose the projects based on what I thought I could contribute to the project as well as my interest in the particular topic.

11.2 Week 2

11.2.0.1 Plans: This week I plan to read into my capstone project once I have been assigned and contact my client.

11.2.0.2 Problems: I still don’t know what my project is but I should know on Tuesday.

11.2.0.3 Progress: Now we have been assigned to our Project. Our project is Prison Match!

11.2.0.4 Summary: This week we were assigned to our projects and we got to meet up with our partners. My partner is Kenny Thompson and my project is Prison Match.

11.3 Week 3

11.3.0.1 Plans: This week we plan to have a meeting with our client to get more details on our project.

11.3.0.2 Problems: Our project was reassigned because our old client for the Prison Match project didn't want to do a capstone project this year. Also one of our group members dropped the course so now we only have two people.

11.3.0.3 Progress: On Thursday we were able to have a google hangouts meeting with our client. We went over deliverables and the main objectives for our project.

11.3.0.4 Summary: This week we were reassigned from the prison match project to Coal and Open-Pit surface mining impacts on American Lands Follow-On our client is Lewis McGibbney who works for NASA JPL

11.4 Week 4

11.4.0.1 Plans: This week my partner and I want to finish and send our problem statement to our client.

11.4.0.2 Problems: In order to finish our problem statement we need to combine the individual problem statements we wrote up into one problem statement which we will send to our client.

11.4.0.3 Progress: On Tuesday we put together our problem statements and edited it for errors. We then sent our problem statement to our client in an email to sign off on. We cc'd the professors and the TA in the problem statement email. Our Client, Lewis replied the same night and checked off on our problem statement.

11.4.0.4 Summary: This week went well. My partner and I created our problem statement and submitted it to our client. Our client then checked off on our problem statement.

11.5 Week 5

11.5.0.1 Plans: This week my plan is to create the rough draft of requirements document.

11.5.0.2 Problems: The problem for this is week is that I'm still not sure which parts of the capstone my client wants in the requirement. We will discuss this with him on Thursday in our weekly meeting.

11.5.0.3 Progress: This week we met with our client and created the rough draft of the requirements document.

11.5.0.4 Summary: This week we needed to create the requirements document. We also had a meeting with our client and TA on Thursday.

11.6 Week 6

11.6.0.1 Plans: This week we planned to complete the requirements document and send it to our client for approval.

11.6.0.2 Problems: We need to come up with the requirements for the requirements document due on Friday that we need to send to our client.

11.6.0.3 Progress: As of Wednesday we completed our requirements document and sent it to our client to look over and approve. We already had the rough draft on our github. On Friday our client approved our requirements document.

11.6.0.4 Summary: This week we completed the requirements document on time and submitted it to our client who approved our requirements document after telling us to add a reference section.

11.7 Week 7

11.7.0.1 Plans: This week we plan to have a meeting with our client on Thursday. I also plan to start the technology review assignment.

11.7.0.2 Problems: This week we need to meet with our client and discuss some more specifics on the capstone coal project. Specifically what deliverables he wants for the week 9 meeting.

11.7.0.3 Progress: This week I've worked on the technology review and discussed with my partner whether or not we should have a meeting in the Thursday in two weeks since that will be Thanksgiving Day.

11.7.0.4 Summary: This week we met with our client on Thursday.

11.8 Week 8

11.8.0.1 Plans: This week we plan to write up the Technology Review rough draft, choose issues to work on in the github for our project COAL-FO and run the existing code.

11.8.0.2 Problems: The problem is that we need to create the rough drafts for our technology review and start choosing issues to work on.

11.8.0.3 Progress: This week I ran the tests for our existing code but I'm still working on running the example problems. I also wrote up the technology review document.

11.8.0.4 Summary: Overall this week went well. I choose an issue to work on github and I was able to run the tests for our project.

11.9 Week 9

11.9.0.1 Plans: This week we will finish the Technology Review and Implementation Plan.

11.9.0.2 Problems: This week we need to finish the tech review and implementation plan. We also need to run the existing code for our project.

11.9.0.3 Progress: This week I worked on the Technology Review and Implementation plan and submitted it via email.

11.9.0.4 Summary: This week went well now we will combine my partners and I technology reviews and implementation plan and sent that to our client for approval.

11.10 Week 10

11.10.0.1 Plans: This week my partner and I will combine our technology review and send it to our client. We will also do the design document.

11.10.0.2 Problems: My partner and I need to combine our technology review documents and fix the errors in them that Kirstin graded us on.

11.10.0.3 Progress: I combined our technology review documents and sent it to my client for approval to be checked off. I started the design document.

11.10.0.4 Summary: This week we finished the technology review and started on the design document and progress report.

11.11 Week 11

11.11.0.1 Plans: This week we plan to finish the progress report presentation.

11.11.0.2 Problems: We need to create presentation over viewing our capstone project.

11.11.0.3 Progress: This week we made the presentation. Each was about ten minutes long and we combined them into one presentation video.

11.11.0.4 Summary: This week was a successful way to end our capstone project. We created a presentation over viewing our capstone project.

12 Winter term, Bryce Egley

12.1 Week 1

12.1.0.1 Plans: This week my partner and I cam back to campus and discussed goals for this quarter on what we would like to accomplish for the capstone.

12.1.0.2 Problems: This quarter we have a lot of work to do on Coal, we must prioritize what we want to complete first.

12.1.0.3 Progress: I managed to get the COAL examples working. I will now go on to fixing issues on the github tracker.

12.1.0.4 Summary: This week was a good start for winter quarter. We need to stay focused and make sure we stay on top of assignments and deadlines.

12.2 Week 2

12.2.0.1 Plans: This week I plan to fix all the problems I encountered when trying to run the examples.

12.2.0.2 Problems: Right now the examples have lots of problems when I try to run them. Example mineral.py, example mining.py, example environment.py all don't seem to work.

12.2.0.3 Progress: This week I fixed most of the problems I found with the examples and pushed this to pycoal. Our client, Lewis, then approved the changes I had made.

12.2.0.4 Summary: Overall this week was very successful. Now we are able to run the pycoal examples. We can see what the hyperspectral images look like after mineral, mining and environmental correlation and we can start fixing other issues on pycoal.

12.3 Week 3

12.3.0.1 Plans: This week we plan to get AWS accounts to store hyperspectral imagery on. We also plan to solve one issue on pycoal relating to the Docker Build.

12.3.0.2 Problems: We need to fix the Docker build since right now pycoal does not build on Docker.

12.3.0.3 Progress: This week we fixed the issue with Docker and made changes to the example scripts so now we can run any imagery through them.

12.3.0.4 Summary: This week we fixed problems with the examples as well as fixed issues relating to Docker. This greatly improves Pycoal since now people who fork/clone/download pycoal will be able to see That works correctly with Docker. Also that the examples will run correctly.

12.4 Week 4

12.4.0.1 Plans: This week my partner and I plan to work on issues on the github issue tracker for pycoal and coal-sds.

12.4.0.2 Problems: For my portion I need to solve two issues on the github issue tracker. One is creating a CLI and which will enable upon install. The other is to improve QGIS install instructions.

12.4.0.3 Progress: So far I have solved 2 issues on the github issue tracker for pycoal and managed to get the examples running on my local machine. Over the next two weeks I plan to solve 3 more issues and start work on the export process to XSEDE.

12.4.0.4 Summary: This week we talked to our client about problems we had Solveing issues on pycoal. We also decided to prioritize the XSEDE export process, since the Alpha release is in two weeks in week 6.

12.5 Week 5

12.5.0.1 Plans: This week I plan to sort out the problems and submit with pull request the three github issues I am currently working on. Which are changing Docker image to use python 3, Create CLI and enable upon install, Improving QGIS.

12.5.0.2 Problems: These issues specifically, changing the Docker image to use Python 3 and Creating CLI and enable upon install have a lot of tedious details behind them and involve a lot of reading and planning to get them done.

12.5.0.3 Progress: This week I finished Improving QGIS instructions and getting the Docker image to python 3. I make need to come back to the Docker image with Python 3 issue. I also made a pull request with my commits for changes to the CLI for my client to review.

12.5.0.4 Summary: Overall this week was productive. I tackled 3 key issues on github and in my client meeting we discussed the next 3 issues I should work to solve which are Enabling Ecosis Library, Upgrade Library to Spectral Library Version 7 and Labeling the Docker Image. I hope to finish these before the Alpha Level release then I will have successfully fixed 9 issues on pycoal which is very good progress for the Alpha Level Release.

12.6 Week 6

12.6.0.1 Plans: This week on pycoal I plan to finish creating the CLI and then start on Enabling Ecosis Library, Upgrade Library to Spectral Library Version 7 and Labeling the Docker Image. We also have the alpha release where we should get something with AWS or XSEDE working. As well as make the poster and presentation. A very busy week!

12.6.0.2 Problems: The CLI is trickier than I thought. But I pretty much finished creating it with help from my client who recommend some slight changes to be made.

12.6.0.3 Progress: This week I created the CLI, the Poster for our capstone and started working on the new issues of Enabling Ecosis Library, upgrading Spectral Library to Version 7 and Labeling the Docker Image.

12.6.0.4 Summary: This week overall was productive. I finished one key issue of creating the cli. Which is needed since we are trying to automate our project. The CLI command line interface will allow users to pass files as arguments to go through mineral, mining and environmental correlations. We also created the poster and Alpha release presentation.

12.7 Week 7

12.7.0.1 Plans: This week I plan to solve the issues of Enabling the Ecosis Library and Upgrading our Library to Spectral Version 7. If I can finished those I plan to start working on labeling the docker image and picking up a new issues on the pycoal issue tracker to start working on.

12.7.0.2 Problems: This week to need to figure out the new issues we are going to be working on. I have chosen three issues and Kenny is starting to do more work with AWS and COAL-SDS.

12.7.0.3 Progress: This week I began work on Finding new data to use to run through pycoal. I looked at the AVIRIS-NG site and found some photos of coal, sulfur mines and one coal mine which we could potentially use.

12.7.0.4 Summary: This week overall was productive. We have taken a new direction from the previous week since we finished the Alpha release. I have began more work on the three issues I mentioned before which should consume my time for the next two or so weeks and Kenny is doing more work getting COAL-SDS to work with AWS(Amazon Web Services)

12.8 Week 8

12.8.0.1 Plans: This week I plan to download 4 AVIRIS-NG images and run full mineral, mining, environmental correlation at least one of them. This would complete the get more data issue, at least for the time being. I also plan on doing some work to upgrade to spectral version 7.

12.8.0.2 Problems: We need to fix problems with GDAL right now we are getting trace errors when running. This isn't a pressing issue though since we can still generate environmental classifications. Fixing this would just make the environmental image more clear.

12.8.0.3 Progress: This week I have finished gathering data by finding a few AVIRIS-NG images of coal and sulfur mines which we can run mineral, mining, environmental correlations on. I wont close the get more issue since this issue could technically be opened forever seeing as though in the future pycoal would be used to run the entire AVIRIS and AVIRIS-NG libraries on and move to XSEDE.

12.8.0.4 Summary: Overall this week was productive. I completed more work on getting more data where I found images of coal and sulfur mines which we can run mineral, mining, and environmental correlations on. Next week I plan to do more work on updating to spectral library version 7 and enabling ecosis library.

12.9 Week 9

12.9.0.1 Plans: This week I plan to run mineral, mining and environmental correlation on three to four images using pycoal and stage the products in our google drive. I also plan to fix a bug with gdal pycoal currently has and hopefully finish getting pycoal up to date with spectral version 7.

12.9.0.2 Problems: Right now pycoal runs on spectral version 6 we can classify more minerals if we upgrade to spectral version 7 and generate clearer images. There is also a bug with gdal which effects the environmental correlation images we are currently generating.

12.9.0.3 Progress: This week I fixed the GDAL bug and began work on updating to Spectral Version 7. We had our client meeting where we fixed up some misunderstandings I had with Updating to Spectral Library Version 7.

12.9.0.4 Summary: Overall this week was productive for our capstone project. I was able to fix the bug with GDAL and Kenny was able to get some work done on the COAL-SDS side of things.

12.10 Week 10

12.10.0.1 Plans: This week I plan to finish on updating pycoal to use Spectral Library Version 7 and then I plan on getting started on Enabling the Eco-SIS library. If I can complete these we will have fulfilled the portion of our project of updating the algorithms and classification methods of the existing code.

12.10.0.2 Problems: This week the problem I need to solve is upgrading pycoal to use Spectral Library Version 7. To do this I need to create a convert function.

12.10.0.3 Progress: This week I made progress on this and finished the update to Spectral Version 7. This week was overall productive even with the barrier of dead week and having a lot of other work in my other courses.

12.10.0.4 Summary: This week was overall productive to the development of pycoal. Spectral Library Version 7 is now being used so the algorithm have been improved which was a goal in our requirements document that we have completed. We can always improve the algorithms more but we have at least met that requirement in one way with this.

12.11 Week 11

12.11.0.1 Plans: This week I will start work on enabling eco-sis. This will further improve pycoal's classification algorithms. We also plan to make the Beta Release Presentation and Video.

12.11.0.2 Problems: This week the main problem is to create the beta release video. This is due on Wednesday and will conclude the winter portion of the capstone project.

12.11.0.3 Progress: This week we completed the Beta Release. It is due on Wednesday and we will have completed it by Wednesday.

12.11.0.4 Summary: Overall this week and quarter have been very productive. On the pycoal side of things we got the examples working, we created a cli, we fixed the gdal and qgis installation instructions, we found more data to run mineral, mining, envrionmental correlation on, we updated Docker to python version 3, we got pycoal to use spectral library version 7 to improve the pycoal algorithms and many more things. This fufilled all the parts of the requirements document hat related to pycoal. Mainly improving upon existing algorithms. I continue to spend time working on pycoal but I should also start giving some of my time COAL-SDS since the pycoal requirements have been met.

13 SPRING TERM, BRYCE EGLEY

13.1 Week 1

13.1.0.1 Plans: This week we re wrote the requirements document, since we removed the COAL-SDS part of our capstone project.

13.1.0.2 Problems: The COAL-SDS part of our capstone project was removed. This wasn't a big deal for me since I am working on pycoal.

13.1.0.3 Progress: This week I worked more on the Spectral Version 7 issues. And Kenny and I re wrote the requirments document.

13.1.0.4 Summary: This week we rewrote the requirements document and I worked on the USGS Spectral Version 7 issue I have been working on.

13.2 Week 2

13.2.0.1 Plans: I have sent out about 15 emails to people working at USGS asking them about Spectral Library Version 7's format. I found one who got me in contact with the correct people.

13.2.0.2 Problems: This week I needed to fix certain issues in my spectral library 7 conversion code that my client wanted.

13.2.0.3 Progress: I made the requested changes to my code and Made a pull request to github for my client to review.

13.2.0.4 Summary: Overall this week was productive. I have now made contact with the people At USGS and I made requested changes to my code on github.

13.3 Week 3

13.3.0.1 Plans: Get convolved USGS spectral Library 7 envi .hdr and .sli files.

13.3.0.2 Problems: You can generate convolved spectral library files using prism software or writing conversion code.

13.3.0.3 Progress: I talked to people working at USGS and we got the convolved library files for USGS Spectral Library Version 7.

13.3.0.4 Summary: Overall this week was very productive. We got the USGS Spectral Library Version convolved files and I tested it on pycoal and it worked correctly.

13.4 Week 4

13.4.0.1 Plans: Begin writing tests for Spectral Version7 conversion funciton.

13.4.0.2 Problems: I need to write test cases for the spectral version 7 conversion I created. Once these tests pass this issue will be complete

13.4.0.3 Progress: I began writing these tests I pushed an initial version to github. Some tests failed and there are changes I need to make.

13.4.0.4 Summary: Overall this week was productive since we finished the spectral version 7 issue Now we just need to have test cases to make sure it works correctly.

13.5 Week 5

13.5.0.1 Plans: Finish tests for USGS Spectral Library 7 conversion method. Update website to reflect this.

13.5.0.2 Problems: The USGS Spectral 7 update issue is complete now we just need to update the website and fix the broken travis-ci.

13.5.0.3 Progress: This week was very productive. We are nearing the end of the capstone. And this issue being solved will be a major addition to pycoal.

13.5.0.4 Summary: Overall this week was very productive. We managed to updated to spectral 7, Now pycoal will be able to identify more spectra and produce more accurate correlation images.

13.6 Week 6

13.6.0.1 Plans: Enhance aster conversion function to work with EcoStress Library. Discuss with my client if convolving ECOSIS Spectral Library is really possible in current format.

13.6.0.2 Problems: I wish to apply my same method applied to Spectral Version 7 to the EcoStress Spectral Library and possibly Ecosis. By doing this we will allow pycoal to work with more spectral libraries.

13.6.0.3 Progress: This week was the last week before the code freeze. Overall I have made significant modicifications to pycoal and met the requirements I had started out on.

13.6.0.4 Summary: This capstone project was very successful. I certainly learned a lot about software development. And I look forward to applying what I've learned through this project to my future career.

13.7 Week 7

13.7.0.1 Plans: This week the plan is to prepare for the engineering expo. Which is on Friday

13.7.0.2 Problems: We have finished all the code we need to. I may keep running mineral correlation images on my Desktop computer so my client will have more samples to show for this project, if he wishes to continue it in the future.

13.7.0.3 Progress: We have made our expo poster, completed all documentation and all coding requirements we set out with. All we have to do now is present our project at Expo!

13.7.0.4 Summary: Overall this has been a very productive capstone project and now we will present our year long project at the engineering expo!

13.8 Week 8

13.8.0.1 Plans: This week just have to do documentation following expo making sure that we meet all the requirements for the class.

13.8.0.2 Problems: We will touch base with our client and our making sure we meet all documentation requirements.

13.8.0.3 Progress: Overall this has been a productive week and we have completed everything we need to for the engineering capstone project.

13.8.0.4 Summary: This capstone project has been very rewarding it terms of getting experience of developing software and working with a client. I will definitely be able to use these skills in the future.

13.9 Week 9

13.9.0.1 Plans: This week we worked on documentation for the final report.

13.9.0.2 Problems: All capstone work besides the final report is completed.

13.9.0.3 Progress: We worked on the final [presentation and report for the capstone project.

13.9.0.4 Summary: Overall this week was productive as we close down the capstone project.

13.10 Week 10

13.10.0.1 Plans: This week we need to make the final presentation for the capstone project.

13.10.0.2 Problems: The final presentation needs to have a demo of how to use all our projects features.

13.10.0.3 Progress: This week we finished the final presentation and submitted it to canvas.

13.10.0.4 Summary: Our final presentation is complete and now all that is left is the final report.

13.11 Week 11

13.11.0.1 Plans: This week we need to make the final report.

13.11.0.2 Problems: To finish our capstone project we need to write up the final report.

13.11.0.3 Progress: This week we wrote the final report and finished the capstone project!

13.11.0.4 Summary: It has been a great capstone project and a great year! Thank you for giving me this experience and I look forward to applying what I have learned through this project in my future career!

14 Fall term, Kenneth Thompson:

14.1 Week 1

This was the first week, we started the new class, had several tasks assigned by our professor, and finished them all in time. It was also a bit of a meet n greet with all our new colleagues to discuss the future of this capstone assignment.

14.2 Week 2

This was the week we received our project, and started contacting the client. We had a few initial hiccups because our client wasnt answering our emails, but we talked to the professor and coordinated on potential ways to address this.

14.3 Week 3

This week, we finally got our new client and we resolved the issues wed been having with our old one. As we had become aware, the old client was backing out of the project, leaving us high and dry. We managed to secure a new client and had our first meeting.

14.4 Week 4

This week we began working on our problem statement, and updating it to fit the new requirements that our client wanted. We exchanged a few emails back and forth with the client and we sent them our rough draft and updated drafts to read over and tell us what they thought. He had several notes that we quickly incorporated into our assignment.

14.5 Week 5

Work on documentation that is being required by the client, attend necessary classes, and continue weekly meetings with client to discuss future of project.

14.6 Week 6

Work on documentation that is being required by the client, attend necessary classes, and continue weekly meetings with client to discuss future of project.

14.7 Week 7

Work on documentation that is being required by the client, attend necessary classes, and continue weekly meetings with client to discuss future of project. Deadlines approaching for tech review and design document.

14.8 Week 8

Worked on our tech review rough draft, talked to client and discussed potential issues, tasked out separate parts of the assignment. We also began testing and debugging some of the code for the project on our own, although we didnt have any formal deadlines or goals yet.

14.9 Week 9

This was the week that several of our requirements came to a head. We turned in our tech review, turned in our design document, and got final approval on our client to make sure we were submitting something that accurately reflected what he wanted.

14.10 Week 10

Had a quick meeting with our client to discuss plans of action to accomplish over winter break, and then pledged to reconvene in January at the start of winter term.

15 Winter term, Kenneth Thompson:

15.1 Week 1

Reconvened for discussions of winter term goals. Reconsolidated the team, got everyone back on the same page, and discussed notes that we accomplished from the winter break. Scheduled more frequent meetings to try and hurry along progress.

15.2 Week 2

Worked on our project, confirmed documentation with the client, and assigned out issues on github. Began researching and studying up on proper way to approach those problems. Attended necessary classes.

15.3 Week 3

Began the process of upgrading certain versions of our project and changing out outdated information that was posted during the original project. Testing of new versions being undertaken.

15.4 Week 4

Continued the process of upgrading testing and fixing earlier issues. Continued working through outdated documentation and implementing newer more updated documentation.

15.5 Week 5

Continued the process of upgrading testing and fixing earlier issues. Continued working through outdated documentation and implementing newer more updated documentation. Started working on our midterm progress report.

15.6 Week 6

Completed the midterm progress report, turned into teacher and client for approval and sign off. Continued working on issues identified earlier.

15.7 Week 7

Client made several request for issues to be addressed, we focused on working our way through those and trying to finish everything up. We began working on data collection improvmenets and increasing the amount of data being fed into the project, we also worked on a blueprint of items to hit in the meeting.

15.8 Week 8

We worked on several milestones we wanted to hit before the Beta, and finished up our roadmap to get those done in time. Made a lot of progress, and worked hard to meet our deadlines.

15.9 Week 9

Continued working on the items we identified last week, worked hard to accomplish our beta release.

15.10 Week 10

Continued working to finish items assigned by beta release. Finished in time. Broke for spring break with plans to reconvene for spring term and finish out the project.

16 Spring term, Kenneth Thompson:

16.1 Week 1

Meeting with professor and client to address issues that had been raised over spring break. Were able to come up with an agreeable roadmap for all parties to be able to complete the project.

16.2 Week 2

Stepped up our meeting frequency, worked on new design doc to get approved by client and professor. Have finalized plan for resolving all issues in place so we can finish our project normally.

16.3 Week 3

Began working towards new goals, classifier callback, familizarization with the issue, and completing testing.

16.4 Week 4

Begin testing and improving testing functions in code for new sections of code. Committed rough outline of planned functionality.

16.5 Week 5

Worked on poster, handed back project to client, stepped away from coding and development.

16.6 Week 6

Worked on pitch then attended and completed expo.

16.7 Week 7

Worked on final documentations and closing out project.

16.8 Week 8

Worked on final documentations and closing out project.

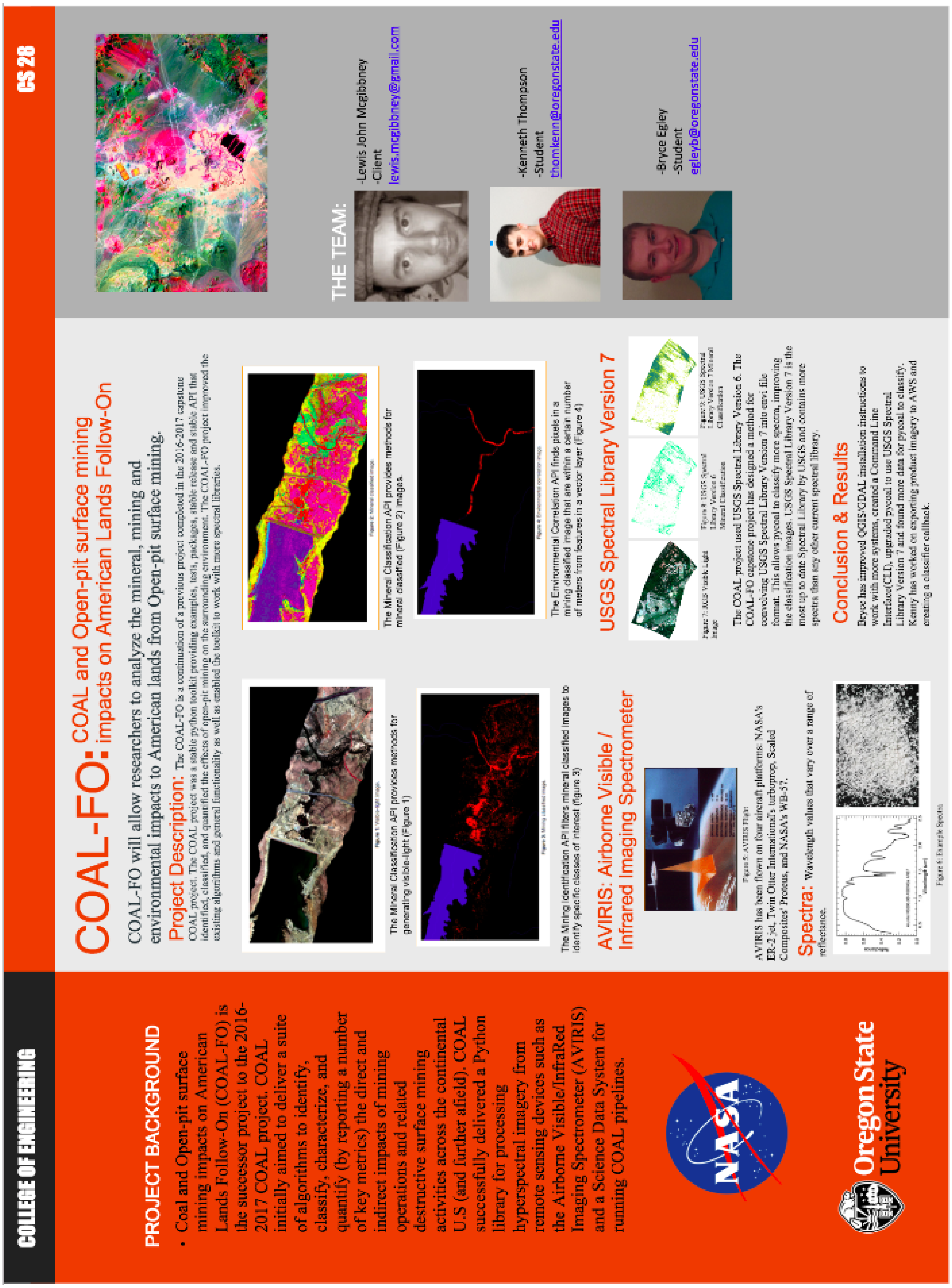
16.9 Week 9

Worked on final documentations and closing out project.

16.10 Week 10

Worked on final documentations and closing out project.

17 PART 5: EXPO POSTER



18 PART 6: PROJECT DOCUMENTATION

18.1 How does your project work

A very basic description of how our project works is that pycoal consumes images from AVIRIS and AVIRIS-NG. Which are infrared images picking up on a range of wavelengths over a swath of land. Pycoal also consumes a convolved Spectral Library file. Spectral Libraries contain values of light reflectance vs wavelength for hundreds of thousands of minerals or spectra. Pycoal matches up the wavelengths in the images from AVIRIS and AVIRIS-NG to the values in the convolved Spectral Library files to determine what minerals are present in the specified AVIRIS image. We then do mining and environmental correlations where we narrow down our focus to a few specific spectra or minerals and compare those to streams and other environmental features in the surrounding area.

18.2 Theory of Computation

Our project could be used for lots of different uses. The intent theory of operation is for looking at the impact Coal and Open pit surface mining has on surrounding lands. As well as monitoring impact of of Coal and Open pit surface mining over time.

18.3 How does one install your software, if any?

First git clone pycoal from github repo Pycoal and follow installation instructions and get the dependencies on ReadTheDocs. It might be useful to also follow the instructions in the example readme Capstone github.

Pycoal works by taking infrared images from AVIRIS or AVIRIS-NG. Data Portal here . Downloadable Data here. Once you find data you want to download on the data portable go to the year in Data Portal. and find the file your looking for by its name.

18.4 How does one run it?

The AVIRIS file needs to be downloaded to the pycoal directory. After building pycoal and following installation instructions, then run the classification files in the examples directory or the CLI if you choose the non default AVIRIS image. Note** AVIRIS files are huge most ranging from 1-50GB. Running a mineral correlation on pycoal will take hours to days. In general Ive seen that every 8 GB in size will be a full day(24 hours), but this may very on your computer.

18.5 Are there any special hardware, OS, or runtime requirements to run your software?

The AVIRIS files are very large. 1-50GB you will want to have a computer with at least 30GB of available storage to use pycoal.

18.6 Any user guides, API documentation, etc.

ReadTheDocs

19 PART 7: Recommended Technical Resources for Learning More

19.1 What web sites were helpful? (Listed in order of helpfulness)

https://github.com/capstone-coal/pycoal

https://github.com/capstone-coal/pycoal

https://github.com/capstone-coal

https://capstone-coal.github.io/

https://speclab.cr.usgs.gov/spectral-lib.html

https://speclib.jpl.nasa.gov/

https://ecosis.org/

http://www.gdal.org/

https://qgis.org/en/site/

https://www.python.org/

https://aws.amazon.com/

19.2 Were there any people that were really helpful?

Lewis John McGibbney (For question on pycoal, coal-sds or capstone website) Raymond Kokaly (For questions on USGS Spectral Libraries)

20 PART 8: Conclusions and Reflections (each team member answers all questions individually)

20.1 What technical information did you learn?

I learned a lot more about the python language, how to convolve spectral libraries, how to create a CLI, improved my skills with git, how to use GDAL and QGIS to view hyper spectral imagery.

20.2 What non-technical information did you learn?

Better time management, quickly reading through large amounts of code and figuring out what it does. Improved my skills with working in a team.

20.3 What have you learned about project work?

I have learned that project work tends to take a lot longer than individual work since you dont just need to meet your concerns and expectations but everyones in the group. As well as get approval from everyone in the group.

20.4 What have you learned about project management?

I have learned that project management is essential for keeping up with deadlines and not allowing yourself to get overwhelmed.

20.5 What have you learned about working in teams?

I have learned that working in teams can make working on large project easier by dividing up the work amongst team members.

20.6 If you could do it all over, what would you do differently?

We spent a lot of time at the start on documentation. I think the day we got assigned the project we should have tried to use all the existing code. It took about a month for me to figure how to use the entire pycoal package and install it on my computer and do a complete mineral, mining and environmental classification. At the very start we were focused too much on documentation. We also really needed three people maybe even four for a project of this size.
We also over committed at the start of the project. I think familiarizing ourselves with all of the different parts of the existing code earlier than we did would have helped with this.

21 PART 9: APPENDIX 1: ESSENTIAL CODE LISTINGS

USGS Spectral Library Version 7 conversion code

```
class FullSpectralLibrary7Convert:
    def __init__(self):
        """
            This class method converts the entire 'USGS Spectral Library Version 7
            <https://speclab.cr.usgs.gov/spectral-lib.html>'_ library into
            its convolved envi format

            Args:
            none
            """
        pass

    @classmethod
    def convert(cls, library_filename=""):
        """
            This class method converts the entire 'USGS Spectral Library Version 7
            <https://speclab.cr.usgs.gov/spectral-lib.html>'_ library into
            its convolved envi format

            Spectral Library Version 7 can be downloaded 'here <https://speclab.cr.usgs.gov/spectr

            Args:
            library_filename (str): path to USGS Spectral Library Version 7 directory
            """
        if not library_filename:
            raise ValueError("Must provide path for USGS Spectral Library Version 7.")

        #This will take all the necessary .txt files for spectra in USGS
```

```
#Spectral Library Version 7 and put them in a new directory called
#"usgs_splib07_modified" in the examples directory
directory = 'usgs_splib07_modified'
if not os.path.exists(directory):
    os.makedirs(directory)

for root, dir, files in os.walk(library_filename + "/ASCIIdata"):
    dir[:] = [d for d in dir]
    for items in fnmatch.filter(files, "*.txt"):
        if "Bandpass" not in items:
            if "errorbars" not in items:
                if "Wave" not in items:
                    if "SpectraValues" not in items:
                        shutil.copy2(os.path.join(root,items), directory)

#This will take the .txt files for Spectra in USGS Spectral Version 7 and
#convert their format to match that of ASTER .spectrum.txt files for spectra
# create a new mineral aster conversion instance
spectral_aster = SpectalToAsterFileFormat()
#List to check for duplicates
spectra_list = []
# Convert all files
files = os.listdir(directory +'/')
for x in range(0, len(files)):
    name = directory+'/' + files[x]
    #Get name
    input_file = open(name,'r')
    spectra_line = input_file.readline()
    spectra_cut = spectra_line[23:]
    spectra_name = spectra_cut[:-14]
    #Remove first and last char in case extra spaces are added
    spectra_first_space = spectra_name[1:]
    spectra_last_space = spectra_first_space[:-1]

    #Check if Spectra is unique
    set_spectra = set(spectra_list)
    if not any(spectra_name in s for s in set_spectra):
        if not any(spectra_last_space in a for a in set_spectra):
            spectral_aster.convert(name)
```

```
                spectra_list.append(spectra_name)

        set_spectra = set(spectra_list)
        print(set_spectra)

        #This will generate three files s07av95a_envi.hdr, s07av95a_envi.hdr.sli,splib.db and data
        #For a library in 'ASTER Spectral Library Version 2.0 <https://asterweb.jpl.nasa.gov/>'_ f
        data_dir = "dataSplib07.db"
        header_name = "s07_AV95_envi"

        # create a new mineral aster conversion instance
        spectral_envi = AsterConversion()
        # Generate .sli and .hdr
        spectral_envi.convert(directory,data_dir,header_name)
```

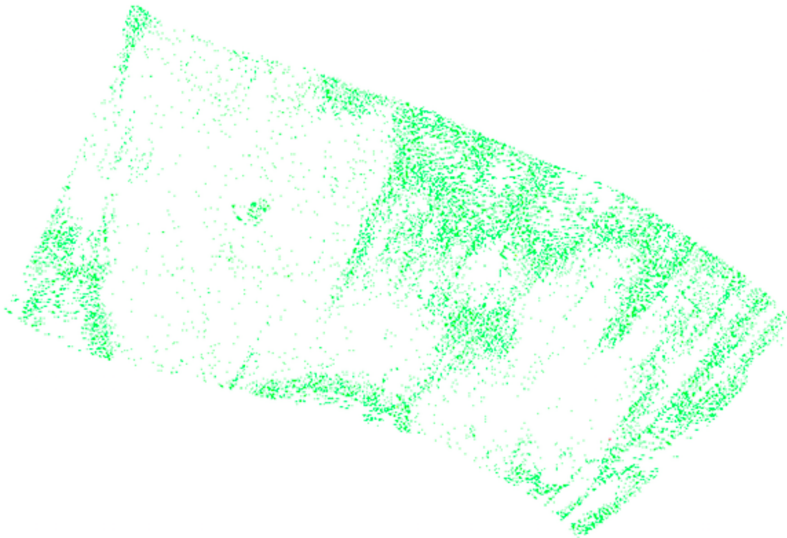
Code for unit tests of USGS Spectral Library 7 conversion function

```
# test files for FullSpectralLibrary7Convert conversion test
_test_SpectralConversion_data = 'usgs_splib07'
_test_SpectralConversion_dir = 'usgs_splib07_modified'
_test_SpectralConversion_db = 'dataSplib07.db'
_test_SpectralConversion_envi = 's07_AV95_envi'

# tear down FullSpectralLibrary7Convert conversion test by deleting test files
def _test_spectral_conversion_teardown():
    _remove_files([_test_SpectralConversion_db,
                   _test_SpectralConversion_dir])

# verify that a small subset of the USGS Spectral Library 7 is converted to ENVI format
@with_setup(None, _test_spectral_conversion_teardown)
def test_spectral_conversion():
    data, dir, db, envi = _test_SpectralConversion_data, _test_SpectralConversion_dir, _test_Spectr
    spectral_conversion = mineral.FullSpectralLibrary7Convert()
    spectral_conversion.convert(library_filename=data)
    spectral7 = spectral.open_image(envi+'.hdr')
    assert isinstance(spectral7, spectral.io.envi.SpectralLibrary)
```

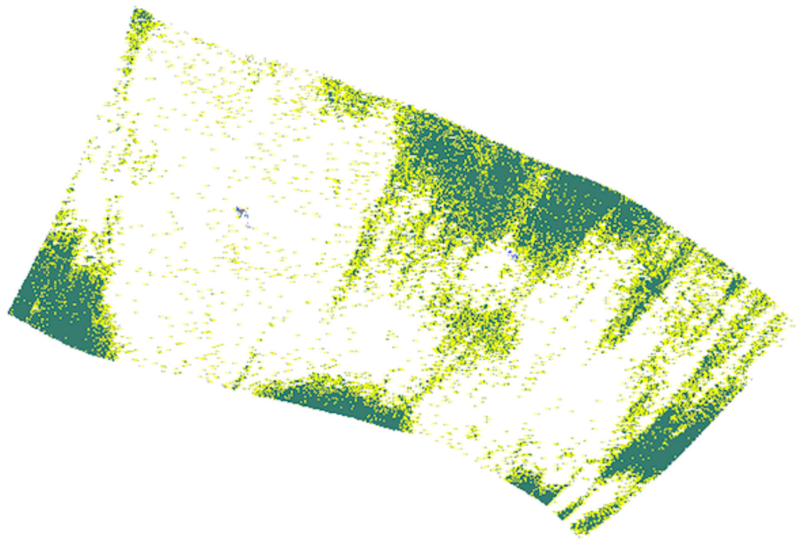
22 PART 10: Appendix 2: Anything else you want to include. Photos, etc.

I had to remove photos since the engineering server did not accept images in Latex.

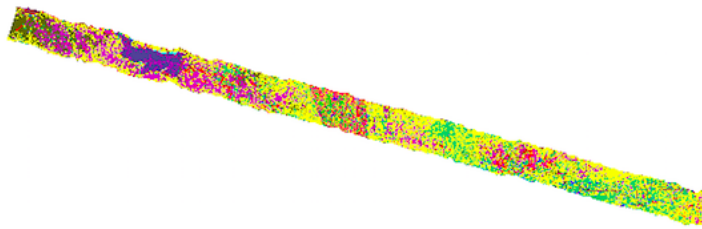
I included the images in in the images folder



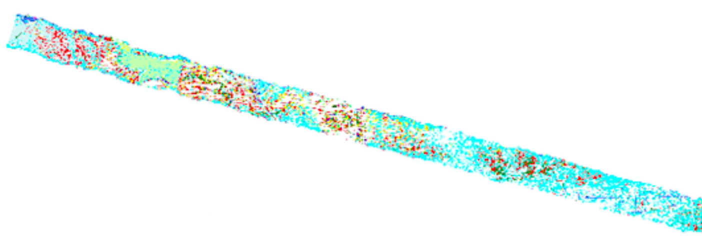
USGS Spectral Library version 6 classification image is images



USGS Spectral Library version 7 classification image is images



USGS Spectral Library version 6 classification of San Juan Coal Mine is



USGS Spectral Library version 7 classification of San Juan Coal Mine is

REFERENCES

[1] CS461 - CS Senior Capstone, Eecs.oregonstate.edu, 2017. [Online]. Available: http://eecs.oregonstate.edu/capstone/cs/capstone.cgi?project=320 [Accessed: 22- Nov- 2017]

[2] USGS.gov Science for a changing world, Usgs.gov, 2017. [Online]. Available: https://www.usgs.gov/ [Accessed: 22- Nov- 2017]

[3]]AVIRIS - Airborne Visible / Infrared Imaging Spectrometer, Aviris.jpl.nasa.gov, 2017. [Online]. Available: https://aviris.jpl.nasa.gov/ [Accessed: 22- Nov- 2017].

[4] XSEDE User Portal Globus User Guide, Portal.xsede.org, 2017. [Online]. Available: https://portal.xsede.org/software/globus [Accessed: 22- Nov- 2017].

[5] C++, En.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/C [Accessed: 22- Nov- 2017].

[6] "MySQL", En.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/MySQL [Accessed: 22- Nov- 2017].

[7] "XSEDE, Extreme Science and Engineering Discovery Environment", www.xsede.org, 2017. [Online]. Available: https://www.xsede.org/ [Accessed: 22- Nov- 2017].

[8] "AVIRIS, Airborne Visible/Infrared Imaging Spectrometer", aviris.jpl.nasa.gov, 2017. [Online]. Available: https://aviris.jpl.nasa.gov/ [Accessed: 22- Nov- 2017].

[9] "AVIRIS-NG, airborne Visible/Infrared Imaging Spectrometer Next Generation", aviris-ng.jpl.nasa.gov, 2017. [Online]. Available: https://aviris-ng.jpl.nasa.gov/[Accessed: 22- Nov- 2017].

[10] "COAL, Coal and Open-pit surface mining impacts on American Lands", capstone-coal.github.io, 2017. [Online]. Available: https://capstone-coal.github.io/ [Accessed: 22- Nov- 2017].

[11] "HPC, High Performance Computing", en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Supercomputer [Accessed: 22- Nov- 2017].

[12] "Query Language", en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Query_language [Accessed: 22- Nov- 2017].

[13] "Enterprise Module Service", hpccsystems.com, 2017. [Online]. Available: https://hpccsystems.com/enterprise-services/modules/esp [Accessed: 22- Nov- 2017].

[14] "HPCC High Performance Computing Cluster", hpccsystems.com, 2017. [Online]. Available: https://hpccsystems.com/enterprise-services/modules/esp [Accessed: 22- Nov- 2017].

[15] "OSU Unix HPC Cluster", cosine.oregonstate.edu, 2017. [Online]. Available http://cosine.oregonstate.edu/unix-hpc-cluster [Accessed: 22- Nov- 2017].

[16] "MIT License", en.wikipedia.org, 2017. [Online]. Available: https://en.wikipedia.org/wiki/MIT_License [Accessed: 29- Nov- 2017].

[17] "Apache License Version 2.0", www.apache.org, 2017. [Online]. Available https://www.apache.org/licenses/LICENSE-2.0 [Accessed: 29- Nov- 2017].

[18] "GNU General Public License, version 2", www.gnu.org, 2017. [Online]. Available https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html [Accessed: 29- Nov- 2017].

[19] "Top Open Source Licenses",www.blackducksoftware.com, 2017.[Online]. Available https://www.blackducksoftware.com/top-open-source-licenses [Accessed: 29- Nov- 2017].