Development of Citizen Science Builder Software for the Planetary Science Institute

Priya Guntur, Physics DepartmentSE

Project Information

Name: Priya Guntur

Research Title: Development of Citizen Science Builder software

Researcher: Dr. Pamela Gay, CosmoQuest

Start date: 6/15/20

End date : TBD, possibly continuing into school year as software project will continue to roll out updates

Internship Role

The purpose of the CSB software is to make the process of image data annotation more rapid and efficient. Currently, the job of annotating the photographs that come from the various satellites in our solar system for topographical features falls to the researchers working directly with the data. This task can be tedious and time consuming, and can lead to researchers spending more time annotating images for relevant features rather than spending it analysing the features. Recent work to outsource this work to machine learning algorithms is underway, however since these images are often of varying qualities and features of interest are not standardized in appearance, it will take time before these algorithms can replace the human eye. CSB aims to outsource the process of image annotation instead to human volunteers who, after some training, can take up some of the responsibility of annotating images and speed up the process.

Most of what we did for the project involved developing features to add onto the existing software base that consisted of an integrated database, server, and front end design. For the summer we were given two tasks. The first was to add a way to import a file containing data related to an image including location, source, and notes alongside the image import so they could be stored together in the database. For this we first had to decide on a format that would be easily accessible to those sourcing the images as well as easily to process efficiently on the software side. The relevant information file would also have to be associated with the image and any pieces of the image so anyone who later pulled up the images would also see the information file when searching. We decided on using a csv format, largely because it is a commonly used one, for example, this is the format Excel spreadsheets and other API's export in, and there are many pre-existing schemas to process the file data so it could be stored as something other than plaintext. Once we had settled on a format, the remaining piece was the development. So far, we have already completed the front end portion of the task and set up a seperate page on the web page specifically for importing files. Once initialized, the page will have a UX experience consistent with what is seen in most apps, two file import buttons (one for the image, the other for the associated csv file) and an indicator of the files progress in being uploaded (this is important for the user experience since uploading large images can often take a couple of minutes). From here, we have been working on writing the connection to the database, making the csv file data into an object with an ID so it can be connected to any of the image storages it is related to. As of this point, around 15 hours have been spent on this task.

The second task involves writing a function to cut images into pieces in such a way that they can be viewed individually with relevant context, as well as pieced back together into a whole image. The reason for this is multifold. Firstly, it would be easier on annotators and researchers to have the image data broken into pieces so they only need to interact with a few of the annotated features at a time - often times the original images will be of large areas of planetary or lunar surface and it can be overwhelming to try and look at or annotate individual topography within such a large image. It would also be useful to have these features be searchable. Future work may involve labeling these features with categories and context labels so a user could find a specific type of feature on a specific planet or moon using a search feature. From this perspective, it also makes sense to break the images into smaller pieces for a more practical concern - file size. Larger images will take longer to render than smaller ones, so it makes sense to save searchers time by only rendering the image pieces most relevant to their work. We also want to preserve the context of each cut piece of the image so they can be pieced back together in the future as users may wish to put them back together into the original image for their own uses - for example, to be used in paper writing or to sense a trend in a body's topographic features. In writing this, we have chosen a simple solution - to cut the pieces according to an X-Y coordinate grid. This approach is more easily doable since all satellite image data will be 2D for the foreseeable future. It will involve building in a functionality that can sense the maximum side lengths needed for a square grid to encompass the image (since some of these images tend to be non-square), then generating the coordinates for the start and end of each square piece of the image to be cut. From there, the pixels within these generated boundaries are saved to their own

image objects in the database, tagged with the ID of the original image they came from, linked to the ID of the csv file of information that came with the original image, and marked with their position in the grid from the original image for when they need to be put back together. Developing this feature has been the more difficult of the two, mostly because image manipulation has proven more difficult than expected. There had been some issues with maintaining clean edges on the image pieces so they don't overlap once pieced back together, this is mostly an arithmetic error and while difficult to recognize, it is fairly easy to fix. There are also edge cases where the square cutting of the original image leads to very small pieces of the original image ending up in a mostly blank square since some of the satellite images are in such atypical shapes. We haven't worked out a way to address this problem yet, but are looking at a couple options including detecting and custom sizing the square piece size for these edge cases or just including them with a disclaimer explaining the reason for their odd appearance to users. We are also considering giving uploaders the ability to determine how their images should be cut up by adding a toggle-able grid overlay on a rendering of the image in the upload page for them to move around as they see fit, possibly to even allow non-square pieces. As of this point, around 21 hours have been spent on this task.

Experience Description

For the most part, this has been a very rewarding experience. I've really enjoyed getting to know my project partner, Ankita, and Dr. Gay is an incredible mentor to work under, both from a personal and professional stand point. One thing I have found very

difficult however, is having enough time to dedicate the effort this project deserves. In planning my commitments for the summer, I was somewhat overambitious and at this point have two other software development internships in addition to this research one and the Skyglow class. Dr. Gay and Ankita both also have other commitements to honor, so it has been a struggle to find meeting times for us to connect and sync up on progress on the project thus far. Ankita and I have also found that pair programming our tasks works out best because of differing experience levels, however, we have had a hard time finding openings when both of us are available to work on this project. Besides the time management issues however, this has been a novel development experience for me. It is unusual for interns to be given such freedom with development projects, usually there is a more structured guidance in place, but I have found that this approach has strengthened my own independence in the development process and will help me in the future when I take on more senior roles in similar projects. There is however, a well established support system in case we did end up needing help. The software development branch of the PSI has a very active discord chat where we could easily get assistance and pointers from other, more experienced developers whenever we needed it.

Knowledge Gained

The knowledge we have gained on this process is so much more than I was anticipating. At school, I take a lot of computer science course work, however, I have found that the knowledge I gain through participating in projects has always surpassed that and this internship was no exception. This was my first introduction to the concept

of a LAMP stack, a software bundle that integrates Linux, Apache, MySQL, and PHP. This type of stack is incredibly common with web services and will definitely be something I will see again in future development work. While I was already familiar with Linux and MySQL, this was my first time working with Apache and writing in PHP so it was good experience learning the intricacies of designing and working with a web server like Apache as well as using a scripting language that was capable of backend work as PHP is. Since all the components of a LAMP stack are open source, we also had a chance to look through source code as we were understanding it which was also valuable insight into the kind of engineering and machine level interaction that goes into creating such well developed stack layers.

While I was expecting to gain technical knowledge, I also gained experience something more unexpected - working with someone with less development experience than I have. Ankita is just starting out with development work whereas I have done it for a couple summers now which made for an interesting contrast in experience levels and made me hammer out my own explanation process to make the onboarding process as smooth as possible for her. This forced me to re-examine the definitions and concepts I was working off of as I explained them to her as it was crucial that I explain them as clearly and accurately as possible since many of these concepts serve as a base for more complicated understandings which led to my own knowledge being more sound.

Timeline: Attach your spreadsheet or table documenting your time, dates, and tasks performed. If you kept a diary, do not include with your timeline; it is for your personal use.

Timeline

6/15/20, 5:00 -7:00 pm EST	Initial meeting with Dr. Gay, introduction to the project and it's purpose, added to Discord group chat, introduced to the project dev community, pointed to resources for project code
6/20/20, 6:00-7:30 pm	Met with Ankita. Worked on trying to get docker install working, kept running into various errors trying to write db config.
6/23/20, 6:00-8:00pm	Met with Ankita. Kept messing with docker install to no success, realized it is likely because one of the dependencies doesn't have the appropriate binary installing.
6/15/20 - 6/27/20	Familiarized ourselves with source code for project and got used to working with components of stack. Tried to (unsuccessfully) use the Docker container for setting up local workstation but ran into issues with container config that eventually proved to require rewriting the dockerfile.
6/27/20, 6:00 - 9:30 pm EST	Met with Dr. Gay, did installation the manual way and decided to add fixing the docker container to our list of tasks to eventually get to. Dr. Gay walked us through how the software works and how the components we can interact with connect to the source code. Given the first task to add frontend components and making webpages for the tasks we would later implement.
7/4/20, 6:00 - 8:00 pm	Met with Ankita, set up web interface for install page.
7/7/20, 5:45 - 7:00 pm	Met with Ankita, drew up design plan for ux used for grid over images for cutting.
6/27/20 - 7/9/20	Worked on setting up web interface with

	appropriately stylized tabs and pages to correspond to the features we would later write the backend for.
7/9/20, 5:30 pm - 8:00 pm	Met with Dr. Gay to flesh out requirements for features we would be assigned. Given the first task to add an information file import feature to import feature to eventually work with master image uploading.
7/9/20 - 7/14/20	Decided on csv as format for information file, worked on setting up import feature so information files would be saved as an object that could be assigned to image pieces once cutting algorithm is developed.
7/14/20, 5:30 pm - 8:30 pm	Met with Dr. Gay to discuss progress on csv file import feature. Given second tasks to work on image cutting feature to store images as puzzle pieces that can eventually be searched and pieced together.
7/14/20 - present	Worked on developing two assigned features, met with Ankita briefly a couple times to make decisions on how the grid issues would be resolved, edge case images, etc.