AstroVision: Enabling visualization and sharing of imaging data from large astronomical surveys through web applications

Shelly Wu

Mentor: Dr. Alberto Krone-Martins

Introduction

Visualizing imaging data from large astronomical surveys is challenging due to the size of these research endeavors and the complex nature of astronomical data [1]. Modern surveys count petabytes of images usually stored in astronomy-specific file formats, such as FITS (Flexible Image Transport System). To help amateur astronomers to explore this data, I intend to keep developing a system that I started during my undergraduate 199 courses called AstroVision.

"AstroVision" is a web-based application for visualizing astronomical imaging data from astronomical surveys. It offers access to professional-grade data by simplifying the visualization and analysis of astronomical phenomena. My planned system offers a responsive interface for users to interact with celestial data. Key features include real-time fetching of sky images via the Pan-STARRS1 [2] survey and the ability to upload datasets for customized visualization.

Key to AstroVision is its interactive table feature, which directly links curated astronomical data with visual representations on the celestial map, facilitating an engaging and educational exploration of the night sky. I am implementing everything using Python for backend operations, with libraries such as Astropy for astronomical calculations, and Flask for web interactions.

In the future, I intend to move AstroVision beyond simple data visualization. I envision AstroVision to foster a vibrant community of users encouraged to share data and insights, promoting collaborative discovery. By combining real-time data access with user-friendly analytical tools, AstroVision will not only make astronomical exploration accessible, but will also deepen our collective understanding of the universe, inviting all to partake in the ongoing journey of discovery.

Objectives

In this SURP project, I intend to keep developing AstroVision in the following areas:

(1) Democratize Astronomical Data Visualization: To provide a user-friendly platform that simplifies the process of visualizing celestial data, making it accessible to both amateur astronomers and the general public with an interest in astronomy. I already have implemented a fully functional prototype of the AstroVision system. During SURP I will deploy this prototype in a server hosted at the Donald Bren School of Information and Computer Sciences under the web address https://astrovision.clotho.ics.uci.edu. I will be configuring the subdomain and the redirections on the Apache 2 web server that is currently deployed at the ICS clotho server.

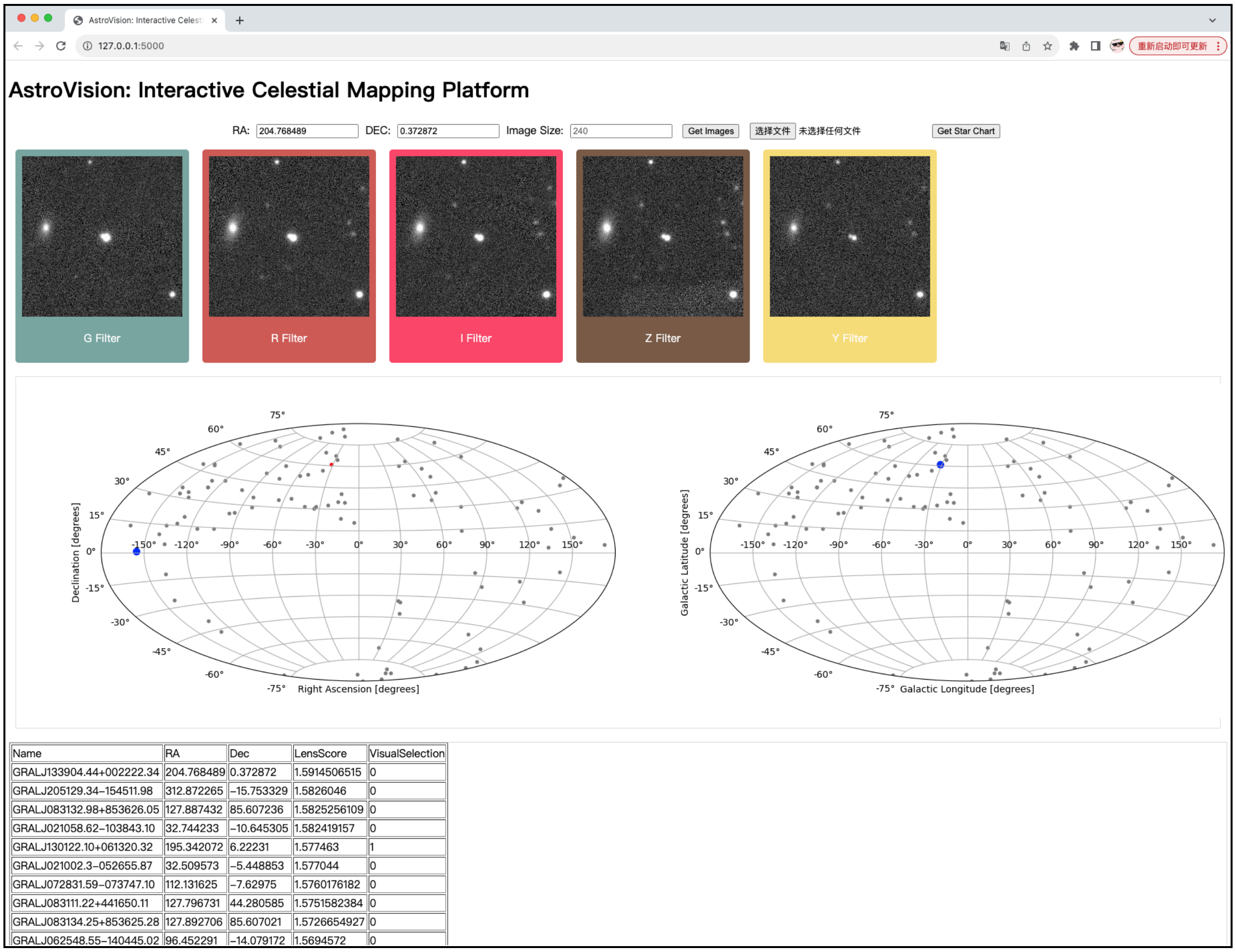

Figure 1. Screenshot of my current AstroVision prototype. The current implementation is focused on software functionality instead of Human-Computer Interaction considerations. This SURP funding will allow me to customize the Graphic User interface better and add additional functionality to AstroVision such as collaborative features, and to study the integration of machine learning classifiers.

(2) Facilitate Amateur Astronomical Research: To offer tools that empower amateur astronomers to conduct their research, including data upload and visualization capabilities, thereby fostering a community of users contributing to astronomical discoveries by integrating near real-time astronomical data in AstroVision. During SURP I intend to add to AstroVision the capabilities of employing near-real-time survey data from additional astronomical surveys beyond Pan-STARRS1 (i.e., Pan-STARRS1 is already implemented), including surveys such as ZTF (Zwicky Transient Facility) [3] and the upcoming Rubin/LSST [4], allowing users to explore up-to-date images of the night sky across different filters.

(3) Study the possibility of incorporating predictive analytics and machine learning: During SURP I intend to study how to introduce in AstroVision seamless machine learning models that could classify images of celestial objects and their time series [5, 6], predict astrophysical events with error inferences, thus enhancing the platform's utility and educational value.

Methodology

In this project I will follow the timeline as specified in the specific *Timeline* subsection below. Moreover, I will make sure to perform two additional tasks throughout the development of the project that will ensure that AstroVision will scale to the required data volumes.

1) Scalable data Integration from near-real-time astronomical data sources

To incorporate near-real-time celestial data, I will implement a simple caching system that will establish connections to additional astronomical survey brokers, such as systems providing ZTF data in the present but planning to provide Rubin/LSST data in the future, such as Alerce and Fink [7]. This process involves developing a data extraction pipeline that efficiently handles large datasets, leveraging Python scripts, a small SQL database to track data provenance and previously downloaded data, and pandas data frames for optimal computational efficiency. Parallel processing techniques using multiple parallel CPU threads (through the Python multiprocessing library) will be employed to manage the high volume of data, ensuring timely updates to the platform.

2) Preprocessing and Data Homogenization

Given the diverse nature of astronomical data, preprocessing steps are essential to normalize and clean the incoming data. This includes adjusting for different time zones, color scales for visualization purposes, and measurement units (the AstroPy framework will be used to help with these transformations). For missing or incomplete data, we will explore various missing data completion techniques (such as MICE, or multiple imputation by chained equations [8]), assessing their impact on the accuracy and reliability of celestial object classification and event predictions. The goal is to create a homogeneous dataset that can be effectively used for machine learning models and visualization purposes.

Responsibilities

I will be responsible for all parts of this project as this is not a team project.

Timeline

Week 1: Implement advanced image processing techniques to enhance visual data from celestial observations, including artificial color rendering from astronomical filters.

Week 2: Conduct thorough background readings on machine learning applications in astronomical data analysis and prediction. Outline methods for integrating near-real-time astronomical data feeds from surveys like Pan-STARRS1 and ZTF.

Week 3: Implement data preprocessing techniques to homogenize incoming near-real-time data for analysis. Start developing machine learning models for classifying celestial objects.

Week 4: Start developing machine learning models for classifying celestial objects, using pre-processed datasets for training.

Week 5: Integrate the machine learning models into the platform, enabling automatic classification and prediction features.

Week 6: Design and implement community features, such as a system for users to tag and share observations, participate in discussions, and collaborate on amateur projects.

Week 7-8: Refine the UI/UX based on the integration of new features, focusing on usability enhancements. Initiate a closed beta testing phase, inviting a select group of users to provide feedback on the new features and overall platform experience.

Week 9: Analyze feedback from beta testing to make final adjustments to features and the user interface. Prepare the deployment environment, ensuring all new features are fully integrated and the platform is stable.

Week 10: Deploy the platform. Monitor the platform closely for any necessary bug-fixing.

References

[1] Moitinho, A., “Gaia Data Release 1. The archive visualisation service”, Astronomy and Astrophysics, vol. 605, 2017. doi:10.1051/0004-6361/201731059.

[2] Chambers, K. C., “The Pan-STARRS1 Surveys”, arXiv e-prints, 2016. doi:10.48550/arXiv.1612.05560.

[3] Bellm, E. C., “The Zwicky Transient Facility: System Overview, Performance, and First Results”, Publications of the Astronomical Society of the Pacific, vol. 131, no. 995, IOP, p. 018002, 2019. doi:10.1088/1538-3873/aaecbe.

[4] Ivezić, Ž., “LSST: From Science Drivers to Reference Design and Anticipated Data Products”, The Astrophysical Journal, vol. 873, no. 2, IOP, 2019. doi:10.3847/1538-4357/ab042c.

[5] Baron, D., “Machine Learning in Astronomy: a practical overview”, arXiv e-prints, 2019. doi:10.48550/arXiv.1904.07248.

[6] Djorgovski, S. G., Mahabal, A. A., Graham, M. J., Polsterer, K., and Krone-Martins, A., “Applications of AI in Astronomy”, arXiv e-prints, 2022. doi:10.48550/arXiv.2212.01493.

[7]Möller, A., “FINK, a new generation of broker for the LSST community”, Monthly Notices of the Royal Astronomical Society, vol. 501, no. 3, OUP, pp. 3272–3288, 2021. doi:10.1093/mnras/staa3602.

[8] van Buuren S, Boshuizen HC, Knook DL. Multiple imputation of missing blood pressure covariates in survival analysis. Stat Med. 1999 Mar 30;18(6):681-94. doi: 10.1002/(sici)1097-0258(19990330)18:6<681::aid-sim71>3.0.co;2-r. PMID: 10204197.