

Allsky Camera Network for Detecting Bolides Plan

Members
Tyler Turner, tturner2021@my.fit.edu
Vincent Quintero, vquintero2021@my.fit.edu
Jean-Pierre Derbes, jderbes2021@my.fit.edu
Charles Derbes, cderbes2021@my.fit.edu
Faculty Advisor
Csaba Palotai, APSS, cpalotai@fit.edu

Goal and Motivation:

Motivation:

- We have identified the current system as unreliable; certain controls are conducive to crashes and or unexpected behavior.
- We believe the software side of the project lacks polish and UX.
- We know the software side of the project is of poor quality and is unmaintainable
- We know the project lacks documentation
- We know the project lacks features such as classification that could take it to the next level.

Goals:

- To build a product for a customer that has practical long term use.
- To rebuild the software part of the project, making it more maintainable and modular for future engineers.
- To significantly enhance UX.

Approach:

- Improved the onboarding process for new boxes to minimize the risk of critical errors during installation. New users can set up boxes more confidently, reducing the likelihood of breaking mistakes. Previously, any setup error required sending the box back to the research team, wasting valuable time and money. Users will be able to more smoothly and easily set up boxes on their own with this update, removing the worry of a simple mistake leading to another post office trip.
- A more streamlined box monitoring process. A dedicated domain enables researchers to easily track and configure all deployed boxes whenever needed. Detailed event and system logs track the status of each Pi (box) deployed so issues can be solved remotely.
- A project completely rebuilt with a strong emphasis on simplicity, expansion, and modularity. This redesign ensures that future developers can easily onboard and contribute towards new project needs.

Novel features/functionality:

- Classification - Right now a person has to do video triage to decide what is or is not an interesting event. A classification model would classify videos as either interesting (e.g. a meteor or comet) or uninteresting (e.g. a fly flew over the camera). This would reduce the amount of manual checks the client does.
- Centralized GUI - Currently each box has its own frontend and processes its own data. We would like to centralize this to a server to reduce workload on the Pi.
- IoT - Currently the client has to connect a keyboard, mouse, and monitor to the box to connect to the internet through PiOS. An IoT style setup would solve this cumbersome problem by eliminating the need for external hardware to connect the box to the internet.

Algorithms and tools:

- Python (Hardware IO)
- OpenCV (Capture/Process video on Pi)
- Golang (Backend)
- FastAPI (Create API for uploading files to server)
- PyTorch (Classification Model for detecting bolides)
- Ansible (Declarative configuration)
- Redis (Fast database)
- SQLite (Light database)
- Tailscale (Networking)
- Netdata (Monitoring)
- Tailwind (GUI Styling)
- Htmx/React/Vue/Flutter (GUI Rendering)
- GitHub Actions (CI/CD)
- SwaggerDocs (Documentation)
- Pytest (Unit tests)
- Image Processing Algos

Technical Challenges:

- Code for the current project does not follow software development principles
- The system is unnecessarily complex
- Contradictory state constants littered throughout the program
- The current C++ code that does the image recording and processing on the RaspberryPi is messy and not organized (plan is to rewrite in Python)

Milestone 1 (Sep 30):

- Stabilize system to facilitate testing, data collection, and collecting metrics on the performance of the current infrastructure, whilst the rewrite is occurring.
- Create system architecture diagram
- Storyboard the frontend
- API design
- Design CLI for image processing pipeline

Milestone 2 (Oct 28):

- Implement API
- Implement CLI
- Show that event can be detected and sent to server

Milestone 3 (Nov 25):

- Classification
- Frontend MVP

Task matrix for Milestone 1:

Task	Tyler	Vincent	Jean-Pierre	Charles
Diagnose current issues	Fix stop camera error	Frontend	Fix state (consolidate into single source of truth)	Fix unknown camera control errors (particularly when changing exposure modes)
Create system architecture diagram	All contribute	All contribute	All contribute	All contribute
Storyboard the frontend	Brainstorm	Create	Brainstorm	Brainstorm
API design	All contribute	All contribute	All contribute	All contribute
Design CLI for image processing pipeline	Create	Brainstorm	Brainstorm	Brainstorm

Approval from Faculty Advisor:

"I have discussed with the team and approved this project plan. I will evaluate the progress and assign a grade for each of the three milestones."

Signature: _____ Date: _____