Camera Project - Kromico

Jon Scheckel, Team Lead & Optical Engineer

Nate Bolin, Mechatronics Engineer

Alba Samsami, Software/Firmware Engineer

Alex Smith, Mechanical Design & Fabrication Engineer

Jacob Cress - Advisor

Background & Goals

- Need high quality mono imaging with Color, UV, IR, and narrowband filters
- Current options are bulky and difficult to setup
- Deliver high-quality, multi-wavelength capture in the field without bulky equipment or laptops
- Integrate filter wheel around sensor to minimize size and optical path length
- Execute end-to-end capture on device with focused UI and local SD storage
- Create practical research instrument for astronomy, environmental monitoring, materials analysis, and forensics

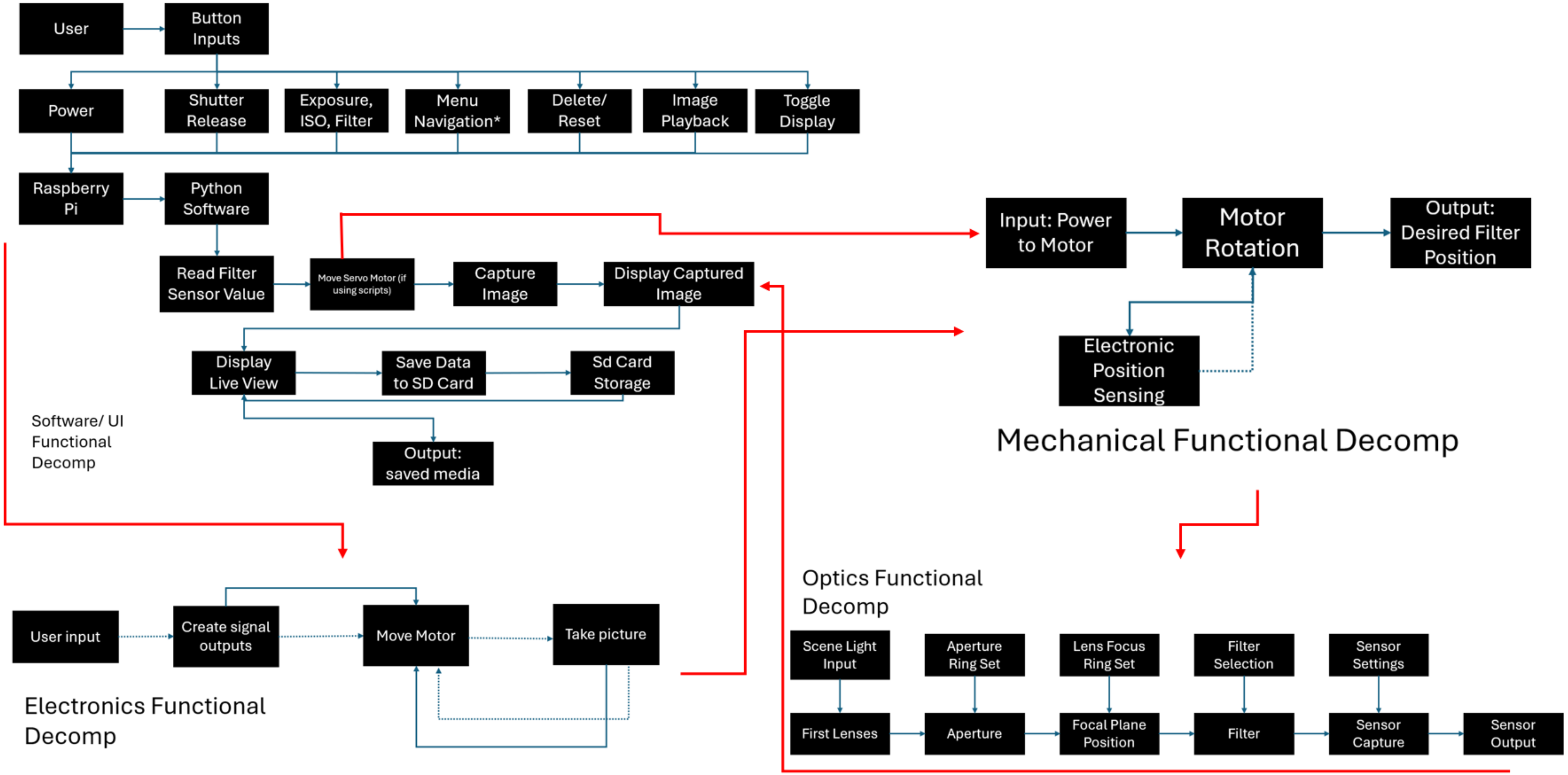
Broader impact

- Expands access to full spectrum imaging using affordable, open hardware instead of expensive proprietary cameras.
- Supports projects in astronomy, environmental monitoring, and microscopy by automating multi-filter imaging with a motorized filter wheel.
- Encourages open, reproducible research by sharing designs and code that others can adapt for their own sensing and vision applications.
- Lowers the barrier for community labs, schools, and hobbyists to do quantitative imaging and data collection in settings that traditionally lack specialized equipment.

Intellectual Merits

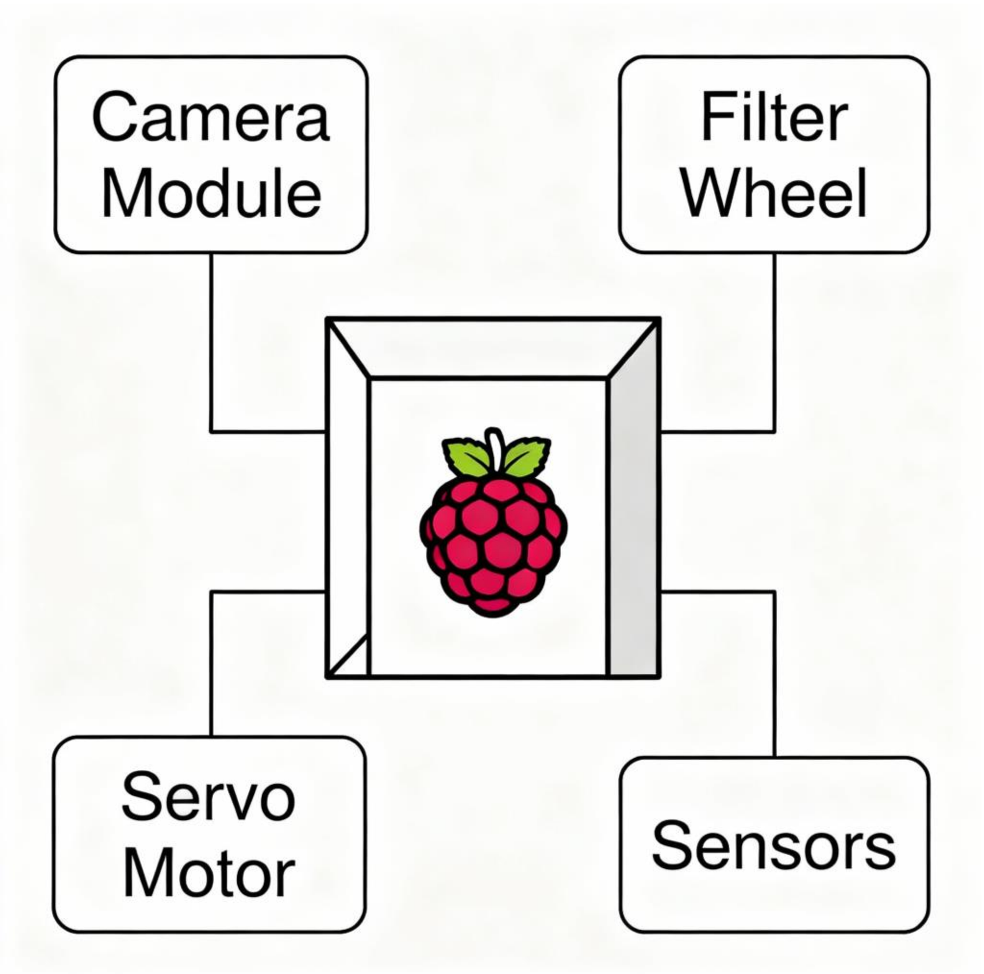
- Research-grade low-light imaging on a low-cost Raspberry Pi using a monochrome sensor.
- Integrated system: camera, light sensor, motorized filter wheel, display, and buttons in one standalone device.
- Fine-grained control of exposure (auto/manual ISO and shutter) for systematic imaging experiments.
- Automated multi-filter capture via stepper-motor filter wheel for spectral/astrophotography-style imaging.
- On-device menu UI and planned stacking/combination features for experimental imaging workflows without a PC.

Design Specifications



Technologies - Software

- Monochrome camera on Raspberry Pi, optimized for low light.
- Python Camera module using picamera2 with auto/manual ISO and shutter control.
- Motorized filter wheel driven by a stepper motor with defined positions for each filter.
- Light sensor to measure scene brightness for positioning.
- On-device display and buttons for menu navigation, mode selection, and capture control.
- Modular Python structure (hardware/, core/, ui/, tests/) with a central main loop coordinating all components.



Milestones

- Feb. 2026 — Camera bring-up and first images captured on Raspberry Pi
- Feb. 2026 — Camera module implemented with auto/manual exposure control
- Feb. 2026 — Stepper motor tested for reliable stepping and homing
- Feb.–Mar. 2026 — Light sensor, display, and button inputs integrated into a single Python application
- Mar. 2026 — Filter wheel control plus multi-filter capture workflow implemented and validated
- Mar 2026 — Full UI (menus, timers, capture modes) and image-stacking features completed and demo-ready
- Mar. 2026 — Filter Wheel Design & Body of the camera
- Apr. 2026 — Final Testing

Results

Planning & Proposal phase

Conceptual design for optics train and filter wheel

Camera operational on Pi with working Camera class (auto/manual exposure, test suite).

Designed camera housing and mounting geometry for motor, filters, and sensor alignment.

Planned stepper-motor step sequences, filter positions, and button layout for navigation/shutter.

Remaining tasks:

- Finalize and fabricate mechanical parts; assemble motor, sensors, and camera into enclosure.
- Implement stepper control for repeatable filter-wheel positioning.
- Wire light sensor, display, and buttons; build on-device menu UI for modes and settings.
- Implement multi-filter capture, image stacking, and end-to-end demo with processed results.

Challenges:

Camera instance conflicts: Multiple Picamera2 instances caused "device busy" errors; solved by implementing a singleton pattern in the Camera class to ensure only one instance exists.

Overexposed images: Initial test captures produced white frames due to excessive ISO/shutter values; debugged by comparing direct picamera2 calls, tuned exposure parameters, and validated with auto-exposure baseline.

Mechanical alignment: Achieving precise filter-to-sensor alignment in housing requires iterative design of mounting points and tolerances to prevent light leaks and maintain focus.

Motor positioning accuracy: Ensuring repeatable filter-wheel indexing demands reliable step counting and homing logic; designed a structured approach with defined step sequences and position tracking.

Real-time integration: Coordinating camera capture timing with motor movement to avoid vibration blur; planning wait periods after motor stop before triggering exposure for stable imaging.

Thank You