# Optimized Open Source Python Code for Raspberry Pi-Controlled Eclipse MegaMovie Photography Automatization

Akira DeMoss[1], Juan Carlos Martinez Oliverez[2,3], Siuling Pau [4], Laura Peticolas[2,3]
Iowa State University[1], University of California Berkeley[2], Space Sciences Lab[3], College of San Mateo [4]

## Background

- The total solar eclipse of August 21, 2017 crossing The United States is a unique scientific and educational opportunity to study the solar corona by solar physicists and citizen scientists.

**Fig 1.** Path of Totality for August 21, 2017 North American Total Solar Eclipse[1]

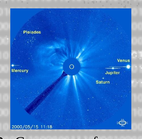



**Fig 2.** Corona seen from space[2]

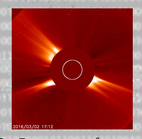


**Fig 3.** Corona seen from space [2]

**Fig 4.** Corona seen from total solar eclipse [2]

- Spacecraft, like SOHO and STEREO, place a disk in front of their cameras to create an artificial eclipse. They are then able to take images with a larger view of the Sun's Corona.
- Only during a total solar eclipse can we observe the inner-most region of the Corona in visible (white) light.

## Objectives

- The Megamovie project rely on volunteers and amateur astronomers to collect scientifically valuable images of the solar corona.

**Fig 5.** DSLR camera [3]

- Three primary optical systems:
  - DSLR cameras
  - Smartphones
  - Raspberry Pi setup

**Fig 6.** Smartphone [4]

**Fig 7.** Raspberry Pi setup

- The Raspberry Pi computer camera setup doubles as an affordable solution for eclipse photography and educational tool for amateur astronomers. This research involves developing the Megamovies software for the Raspberry Pi Computer setup with the following objectives:
  - Optimizing frames per second captured in bracketing and burst modes.
  - Writing image data to a USB drive.
  - Calculating eclipse contact times based on GPS location
  - Optimizing a user friendly GUI that integrates the code.

## Challenges

- Interfacing pre-existing code with GUI
- Performing long operations while keeping GUI responsive
- Implementing parallel processing
- Real time text updates to GUI

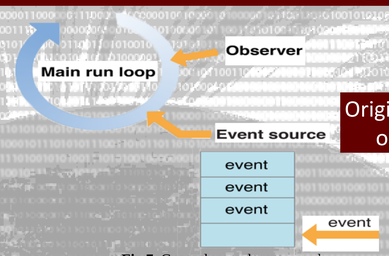



**Fig 7.** General event loop example


**Fig 8.** GUI Freeze screen

```
while True:
    QtCore.QCoreApplication.processEvents()
    strng = serialPort.readline()
    parseGPS(strng.f)
    if time.time() > timeout:
        break
```

**Fig 9.** Blocking "while" loop with member processEvents() statement to move queue forward

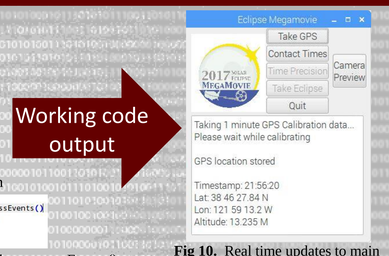



**Fig 10.** Real time updates to main window QTextEdit working properly

## Methods & Design

### Implementation

- Camera preview (toggle button) displays a 4-minute preview to assist the volunteer in focusing the telephoto lens.
- Take GPS acquires signals generated from serial port for one minute then averages last 20 recorded coordinates to complete calibration.
- Time Precision (toggle button) implements network time protocol to synchronize the satellite's atomic clock with the raspberry pi's internal clock for optimized photography.
- Take eclipse uses contact times to for burst and bracket mode photography automatization.

### Code and Testing


**Fig 11.** Raspberry Pi combined with Debian distribution of Linux = Raspbian OS

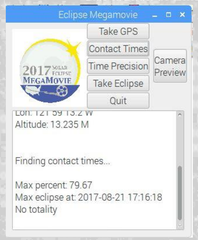



**Fig 12.** Contact times sample output for San Francisco

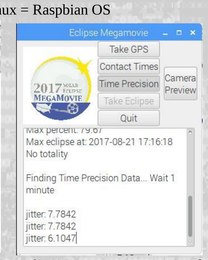



**Fig 13.** Time precision sample output

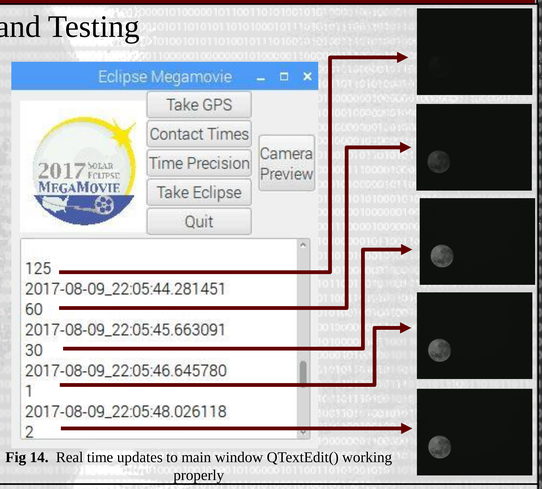
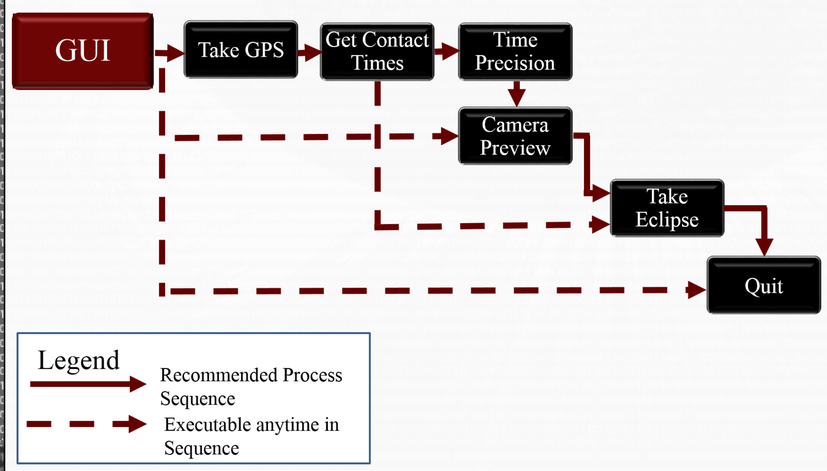



**Fig 14.** Real time updates to main window QTextEdit() working properly

## GUI Event Timing Flowchart




## Raspberry Pi Setup: Electronics and Hardware

- Raspberry Pi boxes include two 85.60 mm × 56.5 mm planks of raft wood, four bolts, GPS hat, a Raspberry Pi, and PiCamera module. These boxes are attachable to a camera tripod (See **Fig 7.**).

**Fig 15.** Raspberry Pi & GPS HAT

**Fig 16.** PiCamera Module

**Fig 17.** Telephoto lens, adapter, and electronics case

**Fig 18.** GPS antenna

- Raspberry Pi is a small, simple computer features include: USB ports, ethernet port, HDMI input, GPU, CPU, and an SD card slot.
- User will need to load pre-configured system image.
- Requires power source, & touchscreen or keyboard, monitor, and mouse to operate

## Expected Outcomes

- Produce a time-expanded solar eclipse video
- Analyze Coronal Mass Ejection behavior

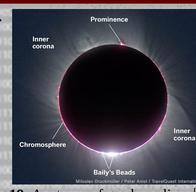



**Fig 19.** Anatomy of a solar eclipse [5]

**Fig 20.** A composite image reveals subtle structure in outer corona[6]

## Future Research

- Future research includes continued optimization for the upcoming solar total solar eclipses.
- Additional optimizations include increasing FPS during burst and bracket mode and reducing GUI lag
- Upcoming total solar eclipses include 2019 in Argentina and Chile, 2020 in Southern Chile and Argentina in 2020, 2021 in Antarctica, and 2024 in Mexico, Central United States, and East Canada.

## Acknowledgements

Much thanks to the ASSURE program for making this opportunity possible, and to Dr. Juan Carlos Martinez Oliveros, Dr. Laura Peticolas, and my fellow scholar Siuling Pau for collaboration on this project. This projected was funded by grant AGS14612277.

## References

[1]"Shadow of the Eclipse." *Nasa Visualization Explorer*. Ernie Wright, 12 June 2017. Web. 3 Aug. 2017. <http://svs.gsfc.nasa.gov/12458 >.
[2] **Fig 2.**, **Fig 3.**, and **Fig 4** are photographs taken courtesy of NASA.
[3] Eclipse Photography – Some Notes for the August Event! (2017, March 16). Retrieved August 04, 2017, from http://regulusastro.com/blog/?p=2474
[4] "Top Eclipse Solar Viewing Kit – 12x Zoom Telephoto/Telecope Tripod & Universal Smartphone Holder Solar Filter Lens 2x Eclipse Glasses | price tracking | price alert | price history on tenedor.net." *The Best Cell Phone & Accessories Price Tracking System*. N.p., n.d. Web. 04 Aug. 2017.
[5] Rick Fienberg. "A Solar Eclipse Glossary." *Solar Eclipse Across America - August 21, 2017*. N.p., 06 Dec. 2016. Web. 04 Aug. 2017.
[6] Writer, Sarah Lewin Staff. "How Scientists Predict the Path of the 2017 Total Solar Eclipse." *Space.com*. N.p., n.d. Web. 04 Aug. 2017.

Berkeley UNIVERSITY OF CALIFORNIA

IOWA STATE UNIVERSITY