Adapting plate-solving for visual astronomy

XIV Portland Alt–Az workshop, 2021

Akarsh Simha

Oct 21, 2021

Introduction

Motivations

- Clear, dark, weekend nights are precious
- What do you enjoy? Finding objects, observing them, or both?

Motivations

- Clear, dark, weekend nights are precious
- What do you enjoy? Finding objects, observing them, or both?
- If you mostly enjoy observing, why not automate the finding?

Existing systems for visual observers

Method	Accuracy	Ease of use	Robustness	Other comments
Manual Star hopping	Great	Poor	Great	Confusing Patterns
Encoders without pointing model	Okay	Great	Good	Retrofitting needs skill
Encoders with pointing model	Great	Poor	Good	Not great for those without permanent observatories.

Existing systems for visual observers

Method	Accuracy	Ease of use	Robustness	Other comments
Manual Star hopping	Great	Poor	Great	Confusing Patterns
Encoders without pointing model	Okay	Great	Good	Retrofitting needs skill
Encoders with pointing model	Great	Poor	Good	Not great for those without permanent observatories.
Plate-solving: my prototype	Great	Good	Okay	Expensive
Plate-solving: refined	Great	Great	Great	Lots of work needed to get here

The dream: an electronic finder scope

Imagine swapping your existing finder scope with an electronic one and voila, you have push to!

An electronic finder scope will have a camera and an integrated computer and wirelessly communicate with your laptop/tablet/phone

What is plate-solving?

"Astrometry.net: Blind astrometric calibration of arbitrary astronomical images"
Lang. D. et. al., *The Astronomical Journal* **139**, pp. 1782–1800.

- Plate solving: Given an image of the sky, determine a mapping between pixel position $\leftrightarrow$ RA/Dec
- Astrometry.net: Open-source software that can find the location, orientation and scale of any picture of the sky

What is plate-solving?

"Astrometry.net: Blind astrometric calibration of arbitrary astronomical images"
Lang. D. et. al., *The Astronomical Journal* **139**, pp. 1782–1800.

- Plate solving: Given an image of the sky, determine a mapping between pixel position $\leftrightarrow$ RA/Dec
- Astrometry.net: Open-source software that can find the location, orientation and scale of any picture of the sky
- Does so by matching star patterns (triangles and quadrilaterals) and comparing against a database
- Downloadable from `http://astrometry.net`
- Try it out by uploading an image to `nova.astrometry.net`
- Already used heavily by amateur astrophotographers!

What is plate-solving?

"Astrometry.net: Blind astrometric calibration of arbitrary astronomical images"
Lang. D. et. al., *The Astronomical Journal* **139**, pp. 1782–1800.

- Plate solving: Given an image of the sky, determine a mapping between pixel position $\leftrightarrow$ RA/Dec
- Astrometry.net: Open-source software that can find the location, orientation and scale of any picture of the sky
- Does so by matching star patterns (triangles and quadrilaterals) and comparing against a database
- Downloadable from `http://astrometry.net`
- Try it out by uploading an image to `nova.astrometry.net`
- Already used heavily by amateur astrophotographers!
- Never seen a false positive!

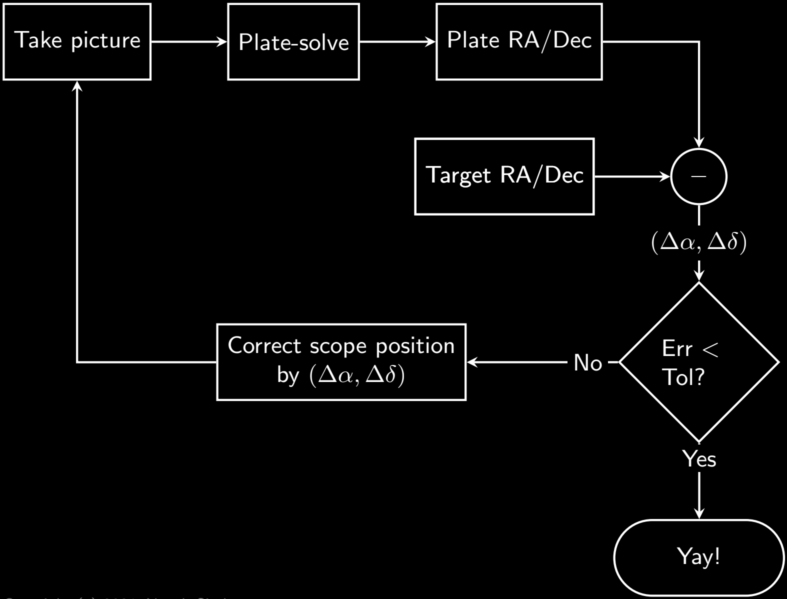
Recorded demo of plate-solving

External video link

The system: In conception

System for plate solving–assisted go-to

Already used by astrophotographers when their go-to does not hit

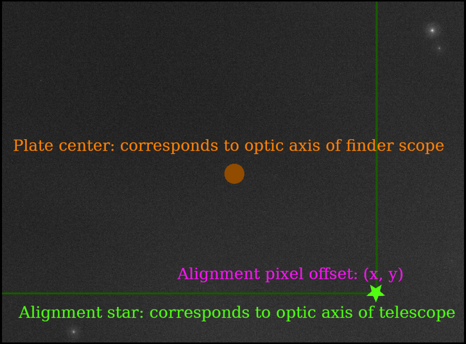


What if the main scope is a visual scope?

- Plate solve through a finder scope

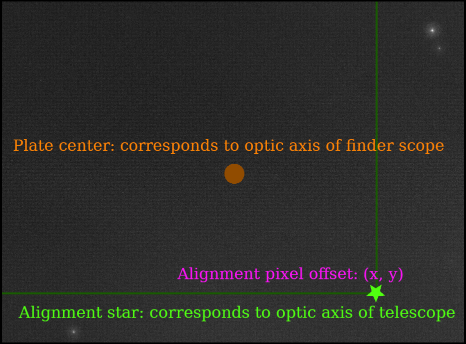
What if the main scope is a visual scope?

- Plate solve through a finder scope
- Need to make a one-star alignment: center a known star in the main scope and plate-solve through the finder
- The alignment determines the pixel offset on the plate (x, y) corresponding to the scope FOV



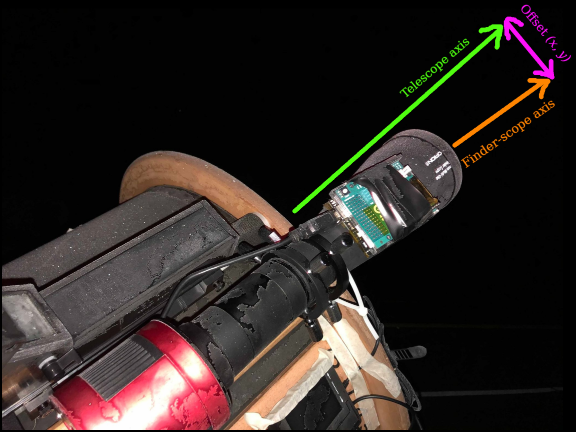
What if the main scope is a visual scope?

- Plate solve through a finder scope
- Need to make a one-star alignment: center a known star in the main scope and plate-solve through the finder
- The alignment determines the pixel offset on the plate (x, y) corresponding to the scope FOV



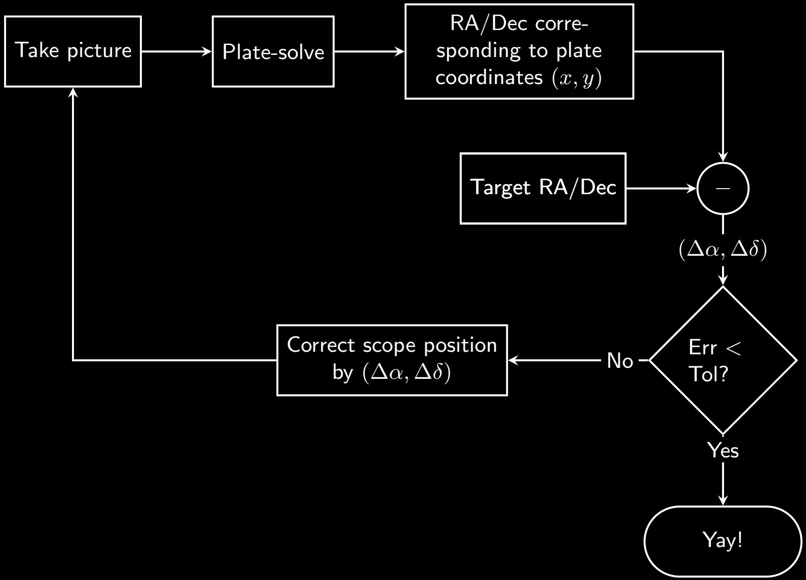
- No need to optically align the finder and the main scope!

One-star alignment: Captures the offset between optic axes



Offset vector (pink) remains fixed as long as the system rotates rigidly

Plate-solving through a finder-scope



What if you don't have go to?

My scope shown is on an equatorial platform

Plate-solve to push-to

For scopes on equatorial platforms / without tracking

- We don't really need go to, or even tracking (sky drifts 1' in 4 seconds)
- Limiting factor seems to be flexure, not tracking (apart from solver not being tuned to work on smudged stars)

Plate-solve to push-to

For scopes on equatorial platforms / without tracking

- We don't really need go to, or even tracking
  (sky drifts 1' in 4 seconds)
- Limiting factor seems to be flexure, not tracking
  (apart from solver not being tuned to work on smudged stars)

The problem:

- Plate-solving takes time
  (best I have managed is once per ≈ 3s).
- For pushing a telescope, we need fast feedback:
  at least once every $\approx \frac{1}{10}$s.

Plate-solve to push-to

For scopes on equatorial platforms / without tracking

- We don't really need go to, or even tracking
  (sky drifts 1' in 4 seconds)
- Limiting factor seems to be flexure, not tracking
  (apart from solver not being tuned to work on smudged stars)

The problem:

- Plate-solving takes time
  (best I have managed is once per ≈ 3s).
- For pushing a telescope, we need fast feedback:
  at least once every $\approx \frac{1}{10}$s.

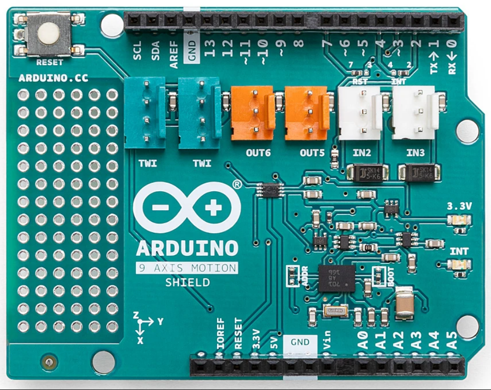
The solution:

- Combine plate-solving with something else that is fast – encoders or IMUs.

The Solution: Inertial Motion Units

These tiny micro-electromechanical devices are everywhere

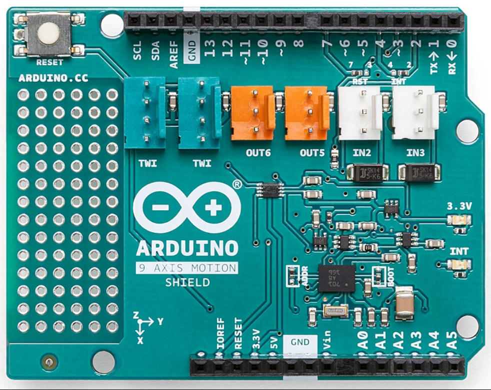
“9-axis” IMU: 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer combined to give 3D orientation



The Solution: Inertial Motion Units

These tiny micro-electromechanical devices are everywhere

“9-axis” IMU: 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer combined to give 3D orientation



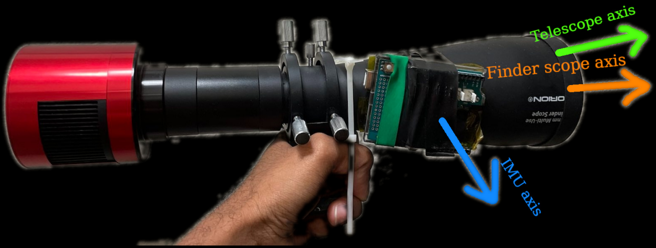
Consumer-grade, open-source devices are very fast ($> 400Hz$), but only provide a rough positioning ($\approx 2^\circ$)

The Solution: Inertial Motion Units

Key idea: Combine plate-solving (slow, but accurate positioning) with IMU output (fast, but rough positioning).

Caveat: More things to align!

IMU is oriented arbitrarily with the telescope (but hopefully fixed rigidly)



Caveat: More things to align!

- Arbitrary orientation of IMU circuit board $\implies$ Need to determine frame rotation between scope and IMU frames
- Magnetic north is unreliable $\implies$ Need to calibrate the azimuth offset between "IMU North" and actual north.

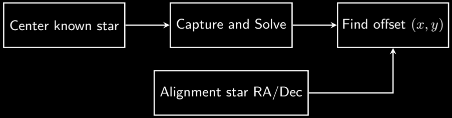
Caveat: More things to align!

- Arbitrary orientation of IMU circuit board $\implies$ Need to determine frame rotation between scope and IMU frames
- Magnetic north is unreliable $\implies$ Need to calibrate the azimuth offset between "IMU North" and actual north.
- We can use plate-solving to calibrate, and keep recalibrating these!

Caveat: More things to align!

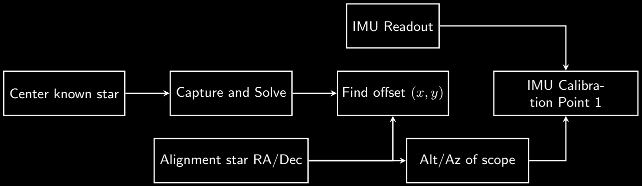
- Arbitrary orientation of IMU circuit board $\implies$ Need to determine frame rotation between scope and IMU frames
- Magnetic north is unreliable $\implies$ Need to calibrate the azimuth offset between "IMU North" and actual north.
- We can use plate-solving to calibrate, and keep recalibrating these!
- Lots of complicated math, but it works.

Plate-solving push-to: Alignment and Calibration



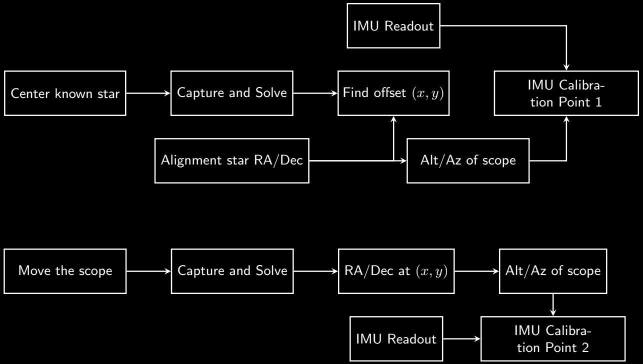
First we do the one-star alignment to calibrate the offset between telescope and finder axes

Plate-solving push-to: Alignment and Calibration



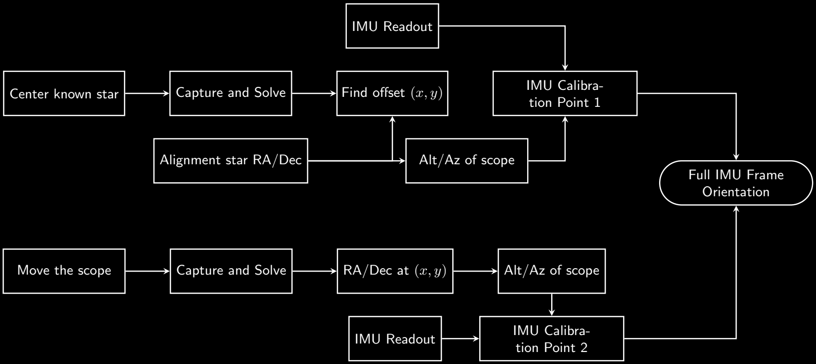
Simultaneously, the system automatically takes one calibration point for the IMU which effectively determines the IMU's orientation with the telescope

Plate-solving push-to: Alignment and Calibration



Simply moving the telescope and triggering a solve provides the system with the second calibration point which determines the orientation of IMU frame from true north

Plate-solving push-to: Alignment and Calibration



With these two data points, the IMU's output can be converted to telescope position. This calibration is continuously repeated with every plate-solve automatically.

Plate-solving push-to: Usage

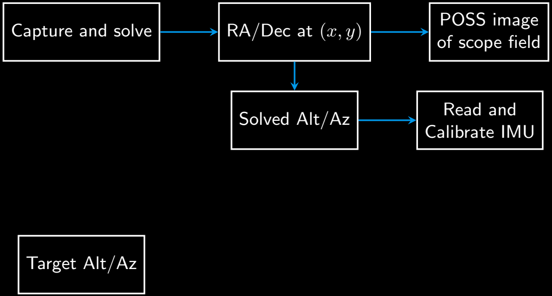


Plate-solving push-to: Usage

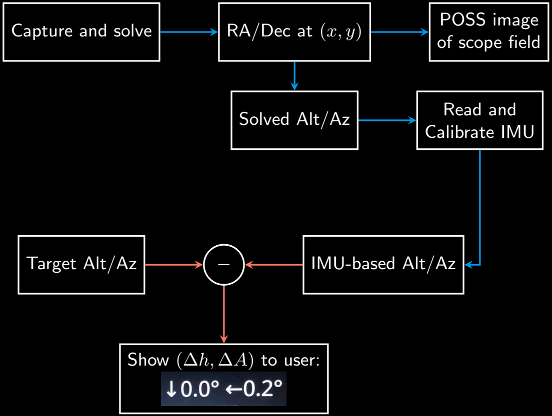


Plate-solving push-to: Usage

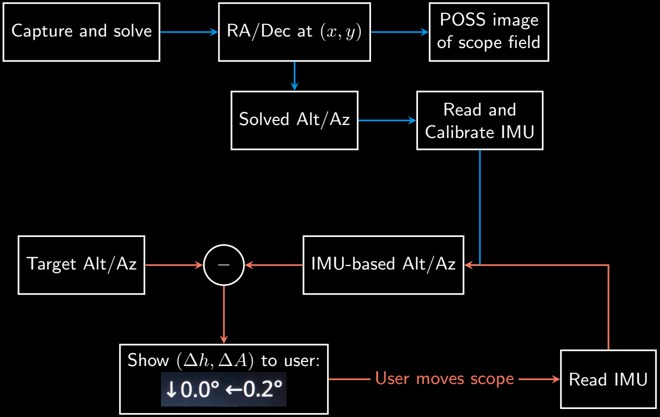


Plate-solving push-to: Usage

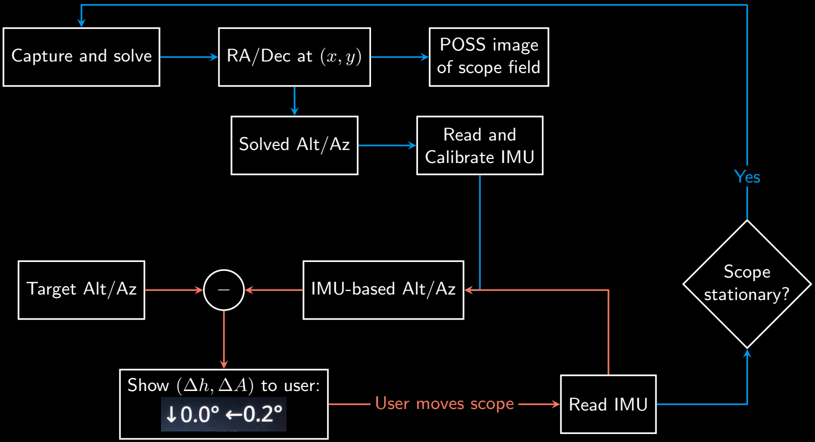
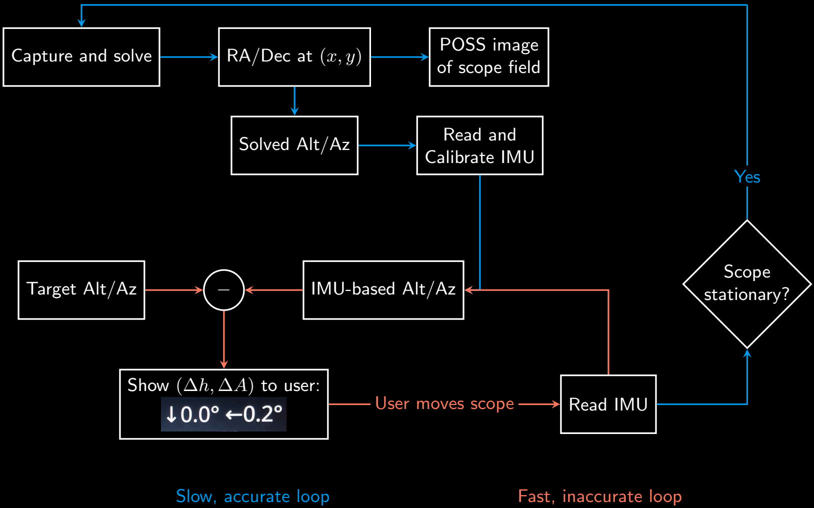
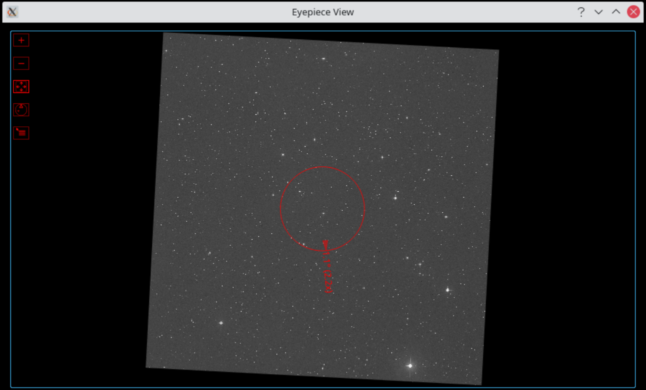


Plate-solving push-to: Usage



Last mile in the eyepiece!

Display next to the eyepiece renders a correctly oriented POSS image with directions



Example of instructions in the eyepiece

The system: In practice

The dream

I don't plan to build this, but hopefully someone will!

The dream

I don't plan to build this, but hopefully someone will!

Ideal user installation:

- Remove the existing finder scope
- Replace with fully-integrated electronic finder
- Mount tablet/HDMI display next to eyepiece and connect to the system

The dream

I don't plan to build this, but hopefully someone will!

Ideal user installation:

- Remove the existing finder scope
- Replace with fully-integrated electronic finder
- Mount tablet/HDMI display next to eyepiece and connect to the system

Ideal user experience:

- Seamless combination of fast and slow loops such that iterations are not needed
- $\lesssim 2'$ accuracy
- A pointing model is quickly developed to learn differential flexure (which mostly carries forward between sessions)

Where I'm at

Recorded demo of my system

External video link

What I'd like to try next

Your suggestions and contributions are invited!

- Transition to a shorter focal length:
  - Limiting factor is flexure, not arcsec/pixel
  - Maybe can get away without tracking!
  - Shorter exposures since brighter stars are likely available in larger FOV!
  - (Harshad, author of SkEye, reports 5' accuracy with just a phone camera!)

What I'd like to try next

Your suggestions and contributions are invited!

- Transition to a shorter focal length:
  - Limiting factor is flexure, not arcsec/pixel
  - Maybe can get away without tracking!
  - Shorter exposures since brighter stars are likely available in larger FOV!
  - (Harshad, author of SkEye, reports 5' accuracy with just a phone camera!)
- Transition to a lighter, cheaper camera:
  - Less flexure
  - Sony STARVIS IMX291 boards available for $35 (Global Meteor Network!)

What I'd like to try next

Your suggestions and contributions are invited!

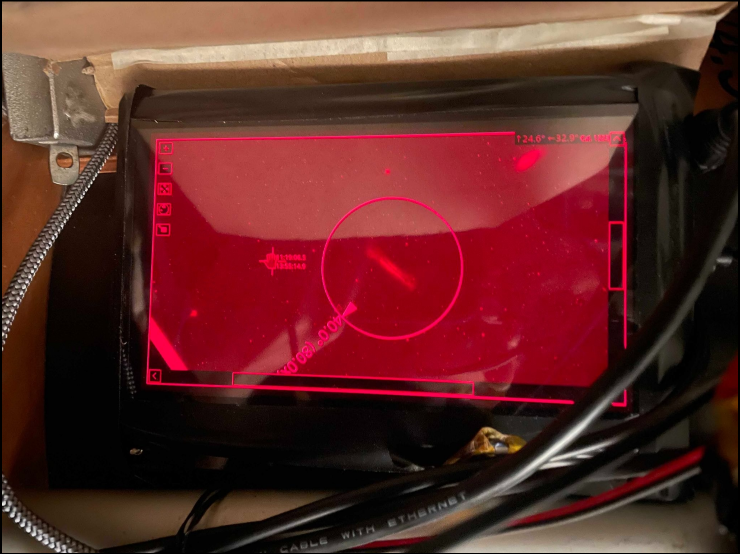
- Transition to a shorter focal length:
  - Limiting factor is flexure, not arcsec/pixel
  - Maybe can get away without tracking!
  - Shorter exposures since brighter stars are likely available in larger FOV!
  - (Harshad, author of SkEye, reports 5' accuracy with just a phone camera!)
- Transition to a lighter, cheaper camera:
  - Less flexure
  - Sony STARVIS IMX291 boards available for $35 (Global Meteor Network!)
- Transition to single-board computer:
  - Full stack is already running on RPi 4B+, taking marginally longer to plate-solve
  - Need to stabilize power supply
  - Eliminate as many wires as possible

Questions?

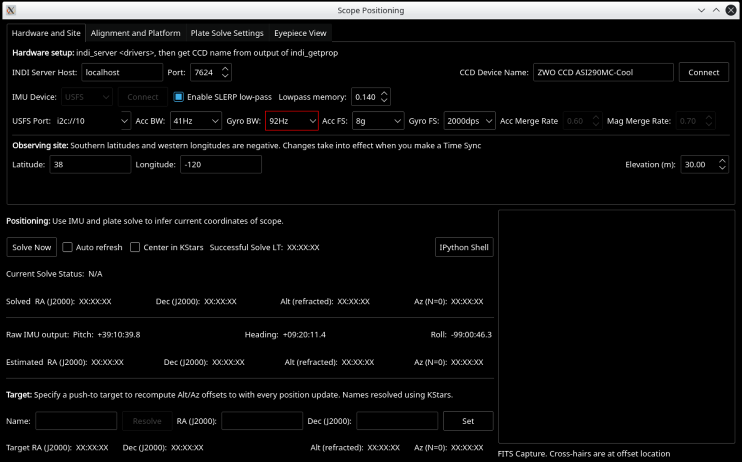
My setup

Finder scope assembly: Finder scope + CMOS Camera + Arduino with IMU Shield

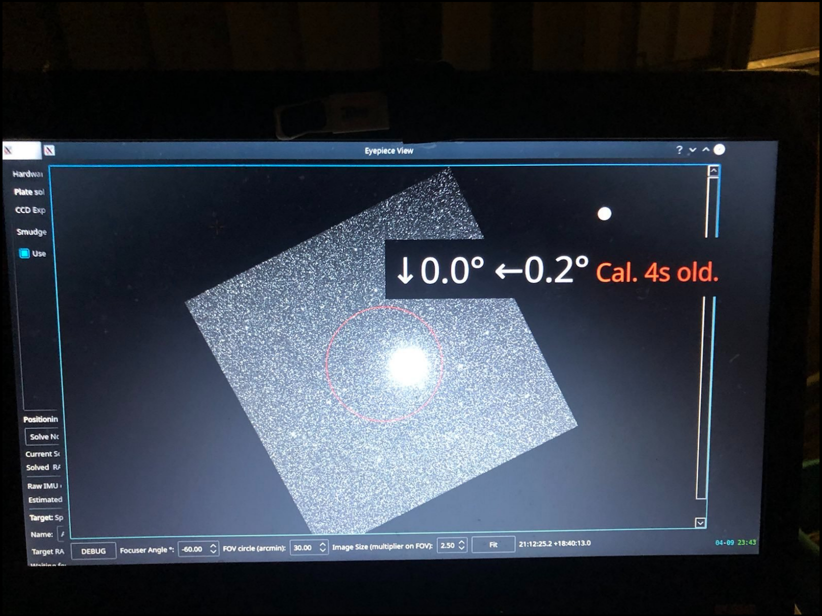
Display assembly: HDMI Monitor + Red filter + Touchscreen panel



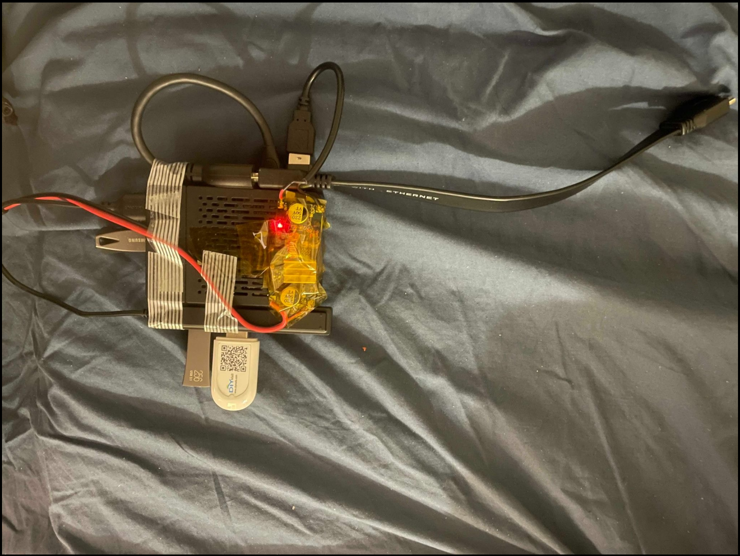
HDMI display in action showing POSS image and directions



Prototype software UI with controls



Alt/Az pushing instructions just like DSC systems

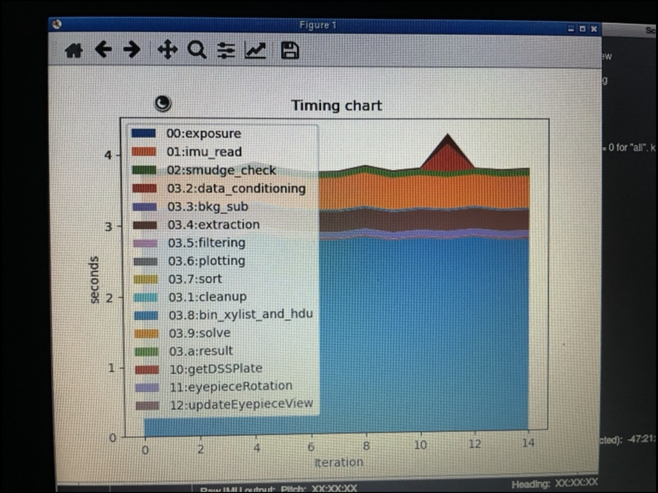


Raspberry Pi 4B+ Computer with peripherals

Testing the capture and solve with the finder scope on a simple tracker

Profiling the slow loop

The faster we can solve, the smoother the finding experience, since fewer iterations are needed



Time taken for various parts of the slow loop on RPi 4B+

Handling an equatorial platform

Equatorial J2000 → Equatorial JNow → real Alt/Az → Alt/Az platform LST!

- Software needs to know the travel time of platform and has to be informed of resets
- Need not re-align every time you reset the platform!
- Just complicates the math