Asterism: Motorized Astrophotography Mount

Team B1

Yuyi Shen
Electrical and Computer Engineering
Carnegie Mellon University
yuyis@andrew.cmu.edu

Kenny Ramos
Electrical and Computer Engineering
Carnegie Mellon University
kdramos@andrew.cmu.edu

Joy Gu
Electrical and Computer Engineering
Carnegie Mellon University
jagu@andrew.cmu.edu

***Abstract*—A computer-vision-based motorized camera mount for astrophotography that is capable of semi-automatic polar alignment, sky tracking, and object tracking through image-processing is proposed. Since the majority of budget astrophotography camera mounts do not integrate feedback from the camera in this manner, this project promises to provide superior ease-of-use by automating the majority of the setup process, while providing comparable endurance and tracking accuracy. The objective is to attain an endurance of 4 hours, a typical setup time of 15 minutes, and a maximal 5% blur when tracking Jupiter over a 60 second exposure.**

***Index Terms*—Astrophotography, Circuits, Computer vision, Motor control, Object detection, Object mapping, Object tracking, OpenCV, Optical flow, Raspberry Pi, Real-time, Software, Stepper motor, Robot**

I. Introduction

Astrophotography is a common pastime amongst photographers and amateur astronomers, and oftentimes involves the taking of long exposures of the night sky or of a specific astronomical object (moon, planet, comet, etc.). This usually requires the usage of a motorized camera mount to compensate for the natural rotation of the night sky or the natural movement of a relatively close-by object. These equatorial mounts usually possess four mechanical degrees of freedom and 1 motorized axis for providing the motional compensation [1], roughly 8 hours of endurance with a 70 W-hr battery [2], and a setup time on the order of 20 minutes [3]. Additionally, commercial mounts typically possess a tracking percent error less than 0.1% as calculated from measured and expected motor periods for the motorized axis (0.2% is considered highly inaccurate and 0.01% is considered excellent) [4].

Our project consists of an equatorial mount with the same mechanical degrees of freedom, but with 3 motorized axes. A computer vision routine running on a Raspberry Pi will link the camera with the motors to automatically calibrate their motion to the user's specific application, making the ordinarily tedious process of setting up longer astrophotography exposures faster and more convenient. This permits the convenient usage of celestial objects other than Polaris for the mount's setup process, making it easy to use even when Polaris is not visible (in contrast with conventional equatorial mounts, which require a rather time-consuming setup of up to 45 minutes when Polaris cannot be manually sighted [3]). Our primary goal is to reduce the typical setup time to a maximum of 15 minutes, even when Polaris cannot be used for the setup. Because our mount will have three times as many power consuming motors as a typical equatorial mount, we will aim for an endurance of 4 hours with a conventional battery (~70 W-h). Lastly, we will aim to achieve sufficient accuracy to track Jupiter with a maximum blur of 5% over the course of a 60 second exposure (a typical use case for such a mount). This is specified in terms of a usage requirement because of the difficulty of predicting a typical tracking percent error through simulation.

II. Design Requirements

The project has three main functions, each of which has its own set of requirements.

A. Polar-Alignment Accuracy

In order to track the movement of the sky, the camera must be turned on an axis aligned with the celestial pole. The product will be able to semi-automatically align the mount's polar axis with the celestial pole. Alignment error should be within 0.5 degrees from celestial pole given accurate input from the user, and alignment should take less than 15 minutes (2/3rds of the typical figures cited in [3]). This maximal alignment error is derived from the minimum arc-length of Jupiter and the 5% error figure cited earlier:

$$T_d = \textit{stellar day} = 86164.098 \textit{ sec}$$
$$T_c = \textit{capture time} = 60 \textit{ sec}$$
$$\theta_{cap} = \textit{angular displacement over capture}$$
$$P_r = \textit{projected star sphere}$$
$$\theta_{cap} = \frac{T_c}{T_d}$$
$$\theta_{err} = \textit{polar alignment error}$$
$$D = \textit{error deflection from object}$$
$$L = \textit{Jupiter's minimal arc length} = 29.8''$$
$$D = 2\sin(\theta_{err}/2) \times L < 0.05LP_r$$
$$\theta_{err} = 2\arcsin\left(\frac{D}{2P_r\theta_{cap}}\right) = 0.594 \text{ deg}$$

B. Sky-Tracking Accuracy

Tracking the movement of the sky involves turning the camera along the polar axis at a rate near 4.178×10^{-3} degrees per second, one full rotation over the course of a sidereal day, which is approximately 23 hours, 56 minutes, 4.0905 seconds.

Given accurate polar alignment, it is required that the camera be turned at a rate very close to 4.178×10^{-3} degrees per second in order to ensure accurate sky-tracking.

It is also necessary to ensure that there is enough torque to turn the camera and to guarantee that the motor can turn at the rate required for a long time, up to four hours. This means that the mount and its motors must be powered for the duration, which should be possible given access to a large (70 W-hr) battery, such as a car battery. Gearing will be necessary to reduce the speed of the motor so that one step of the motor corresponds to a sufficiently small angle. Our design specifies that the driving compensator motor rotates at 3 steps/s which allows us to have better fine tuning when identifying that results in drifting. Given a 400-step stepper motor the compensator should convert input rate at an effective ratio of:

[(360 deg/400 steps)*3steps/s]/(360 deg/86164s)
= 646.23

Furthermore the final gear in the speed reducer gearbox should have a fairly high teeth count due to the high resolution required in the final step.

C. Object-Tracking Accuracy

The project also involves implementing object tracking, for objects which do not move at the same rate as the rest of the sky. The requirements for object tracking include accurate computer vision object detection and tracking and accurate actuation of the motors.

The setup procedure for the mount involves the taking of multiple long exposures within a reasonable amount of time in order to identify object movement based on the drift of point lights (stars) in a set of photographs. Furthermore, the option to track the movement of a single object through the same set of captures is given. Using EQ.1 it is found that most astral objects have an apparent size in the range of 1-30 arcseconds in the night sky.

$$\theta_{apparent} = 2\arctan\left(\frac{ObjectDiameter}{2 \times ObjectDistance}\right) \quad (1)$$

This informs the requirement for the accuracy of object tracking. Due to the diagonal angle of view of 63.00° a typical lens[11] in a far field capture and an 18MP resolution dimensions of (5184x3456) this translates to 6230 diagonal pixels at a angular resolution of (63/6230) ~ 0.0101 of a degree per pixel, or ~36 arcseconds per pixel. For a more accessible lens[13] the listed angle of view is 89.9° and assuming the same 18MP resolution we find an arc resolution of 52 arcseconds/pixel. These are, of course, bigger than most of the point lights “should” be given their angular size. However, due to things like atmospheric scattering, apparent magnitude, and the discrete way in which photons are detected in DSLR camera sensors this increases their apparent size in a photograph. Given that, their apparent size still proves to be a good metric for understanding the overall path of the **source** of the light being captured as an error in the tracking of the source’s center translates quite nicely to an error in the captured result.

Given all of this our target is to eliminate **noticeable** errors in object captures. For this we qualitatively found tolerable levels of error for typical captures using these lenses of both a nearby object (The Moon) and a far away object referenced before (Jupiter). We settled at a 5% total position error for The Moon and a 40% error for Jupiter. This would translate to a 1’42.5’’ position error for The Moon over a given capture and an 11.6’’ position error for Jupiter.

One final note is that these numbers mean different interpretations for different capture qualities/resolutions. That is to say, for large objects we expect our system to be accurate within 1’42.5’’ and for smaller objects the metric of 11.6’ seems adequate given experimentation.

A capture where the lens is a reflective collector telescope will provide a much smaller angle of view and therefore have the effect of a much larger Jupiter capture. In this case the CV portion of our object tracking setup would detect Jupiter as a larger object due to the increased resolution and would therefore be able to track more effectively as it would have a higher absolute accuracy and higher accuracy requirement due to smaller errors being more noticeable.

III. System Architecture

Software running on a Raspberry Pi will use images from a user-provided camera and input from the user via a GUI to calibrate the motion of the three motors of the camera mount to perform one of three functions.

Using a GUI, the user will be able to select the mode of operation, input information so that the mount can semi-automatically polar align, fine-tune the position of the polar axis, select an object to track, or turn sky-tracking on and off.

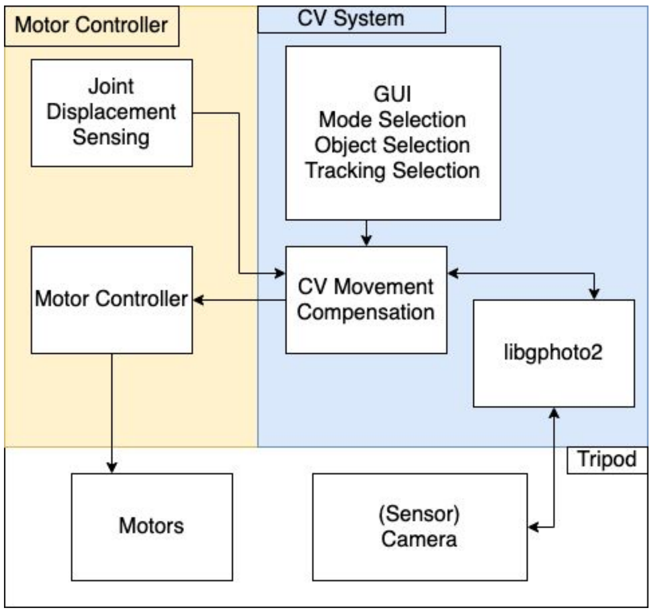
The three stepper motors are connected to a motor controller board, which interfaces with the Raspberry Pi via GPIO pins. The gyroscope for sensing of joint displacement interfaces with the Raspberry Pi via SPI. All PCBs will either be equipped with on-board voltage regulators or SMPSs (not shown in Fig. 1) to minimize the number of power cables. All electrical components will be designed to receive power from a 12V battery, the Raspberry Pi's 3V3 volt rail, or the Pi's USB port.

The software running on the Raspberry Pi (see Fig. 1.b.) uses step-counting and input from the gyroscope to maintain information about the positions of the camera mount's axes. It will be aware of the gearing ratios involved and the zoom level of the camera. The GUI will allow the user to calibrate the software to the specific camera used.

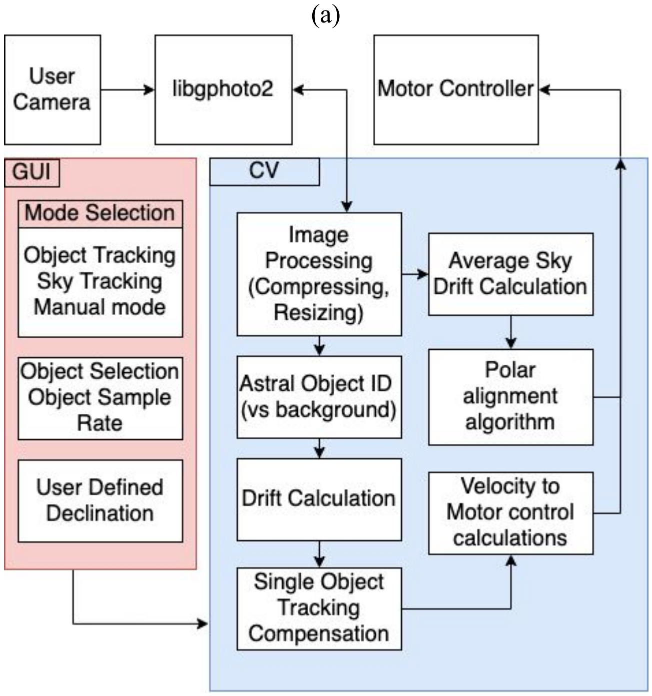
The image-processing aspect of the software polls the camera via libgphoto2 for images at a user selected interval and then processes it to calculate inputs to the motor controller board.

If in object-tracking mode, the computer vision algorithm will process an input image for the location of the tracked object and perform calculations based on the previous locations of the object to determine inputs to the motor controller in order to compensate for the motion of the object.

If in polar alignment mode, the system will instruct the user through a set of calibrating steps that involve pointing the mount's finder scope to astronomical objects that are both user known and stored inside a database by our system.



(a)



(b)

Fig. 1. System picture. (a) Overall system. (b) Software block diagram.

The user will choose to align with two of these predefined objects and, after calibration, the system will have approximate knowledge of the user's zenith position relative to the polar axis. This will then allow the system to align the axis of the "compensator" to the polar axis.

If in sky-tracking mode, the compensator will simply turn at the rate required to track the sky. The user will be able to start and pause this compensation and adjust the starting position via the GUI.

Section V goes into system architecture in further detail.

IV. Design Trade Studies

When designing the system, we considered a number of different configurations for the overall system and its subsystems in order to meet the design requirements specified in Section II.

A. Motor Selection

When considering motorization of the mount, it was necessary to decide on the type of motor to use and whether different motors should be used for different axes.

Ultimately, a bipolar stepper motor with a step angle of 0.9 degrees, with a rated voltage of 3V, a rated current of 1.7A/phase, and a holding torque of 48 N-cm, was chosen. It was selected for its high resolution and reasonable cost per motor. In addition, with a stepper motor, a motor encoder sensor for feedback is unnecessary, because motor steps can be counted to approximate position. This motor also satisfies the torque requirements for each motor axis by a decent safety margin.

The torque requirement for the motor on the polar axis shaft was calculated to be 1.77×10^{-4} N-cm [12]:

$$DSLR\ weight = 1.13\ kg,\ 200\ mm\ lens\ mass = 0.77\ kg$$
$$m = 1.13 + 0.77 = 1.9\ kg$$
$$Max\ linear\ deflection\ of\ cantilever = \frac{WL^3}{3E}\frac{64}{\pi D^4}$$
$$E = 10 \times 10^6\ [aluminum, 70\ \deg F]$$
$$\theta_p = 60\ \deg\ [worst\ case\ latitude\ of\ observation]$$
$$\delta\theta_p = 0.594\ \deg\ [polar\ alignment\ spec]$$
$$\delta d = L[\sin(\theta_p + \delta\theta_p) - \sin(\theta_p)] = 2L\sin(\frac{\delta\theta_p}{2})\cos(\frac{\delta\theta_p}{2} + \theta_p)$$
$$\delta d = 0.0051L = \frac{WL^3}{3E}\frac{64}{\pi D^4}$$
$$D = 0.75\ in.\ [solid\ aluminum\ shaft\ diameter]$$
$$L_{max} = 1.9\ ft$$
$$I_{shaft} = 0.5 \times 0.234\ kg \times (0.952\ cm)^2 = 1.06 \times 10^{-5}\ kg \cdot m^2$$
$$I_{camera} = 0.5 \times 2 \times 1.9\ kg \times (16\ cm)^2 = 0.0486\ kg \cdot m^2$$
$$\omega = 4.178\ mdeg/sec$$
$$T_{rampup} = 2\ sec\ [half\ the\ time\ of\ a\ "short"\ long\ exposure]$$
$$\rightarrow \alpha = 3.64 \times 10^{-5}\ rad/s^2$$
$$\rightarrow \tau = 1.77 \times 10^{-4}\ \text{N-cm}$$

Similarly, the torque was calculated for the 1st polar alignment motor (which needs to lift the polar axis shaft, compensator motor, and camera):

$$I_{compensator} = mr^2 \approx 0.3\ kg \times (0.02\ m)^2 = 0.00012\ kg \cdot m^2$$
$$I_{shaft} = \frac{1}{3}ml^2 = \frac{1}{3} \times 0.234\ kg \times (0.305\ m)^2 = 0.0072\ kg \cdot m^2$$
$$I_{camera,counterweight} = 2 \times 1.9\ kg \times (0.305\ m)^2 = 0.353\ kg \cdot m^2$$
$$I_{tot} = 0.36\ kg \cdot m^2$$
$$\omega = \frac{180\ \deg}{10\ sec}\ [arbitrary]$$
$$\rightarrow \alpha = 0.0628\ rad/sec^2$$
$$\tau = 2.3\ \text{N-cm}$$

Lastly, the torque for the 2nd polar alignment motor (placed as a turntable under the 1st polar alignment motor) was found:

$$I_{1stpolarmotor} = 0.5 \times 0.3\ kg \times (0.02\ m)^2 = 6 \times 10^{-5}\ kg \cdot m^2$$
$$I_{tot} = 0.36\ kg \cdot m^2$$
$$\omega = \frac{180\ \deg}{10\ sec} \rightarrow \tau = 2.3\ \text{N-cm}$$

To simplify driver circuitry, the same model of motor is to be used for all three axes. A large set of motor types was considered qualitatively before settling on bipolar hybrid stepper motors.

TABLE I. Motor Types and Considerations

Motor Type	Pros	Cons
Brushed DC motor	Low driver circuit complexity, decent torque. Permanent magnet DC motors are very cheap.	Difficult to model transfer function between motor power and speed (might require a motor encoder sensor for feedback), datasheets usually don't specify torque vs. current curve. High motor speeds mean that a higher-ratio gearbox is required (expensive).
Brushless DC motor	Built-in speed control and tachometer sensor, decent torque.	Decently sized ones are on the order of 80 dollars (Digikey). There is one model that costs ~20 dollars, but it is low in stock.
Synchronous AC motor	Rotates at speed of powerline frequency, accuracy is only dependent on frequency control	Anything larger than a vibration motor is on the order of ~$50.
Permanent magnet stepper motor	Produces detent torque (remains stationary when undriven). Gives higher torque than variable reluctance motors at limited speeds. Reasonable prices.	Moves in steps (usually 100 per rev.) Requires a polyphase driver circuit.
Variable reluctance stepper motor	Less torque drop-off at higher speeds.	No detent torque, notorious for noise.
Hybrid stepper motor	Highest resolution of stepper motor types, cheap on Pololu, can be half-stepped or microstepped for further resolution benefits.	Usually more expensive than the other stepper motor types. Bipolar motor drivers are effectively H-bridges, and are harder to implement, while unipolar motor drivers require more wiring.

Fig. 2. The types of motors considered and characteristics of each [14,15].

B. Gyroscope Selection

In order to permit our equatorial mount to compensate for the motion of the sky, the shaft of its compensator motor must be aligned with the celestial pole. In the case of our project, this process involves the manual sighting of a known astronomical object by the user through a polar-axis scope attached to the mount, which is accomplished by turning on the mount's stepper motors and manually adjusting the mechanically free declination axis of the mount. The adjustments made by the user during this phase are recorded and used to automatically align the compensator motor shaft. The declination axis is not motorized, and therefore requires some manner of rotational sensor.

The options were to install a motor encoder on the declination axis or to place a gyroscope/accelerometer on the knob of the declination axis shaft. We first examined this qualitatively:

TABLE II. SENSOR TYPES

Sensor Type	Pros	Cons
Motor encoder	High precision, easy to map sensor output to desired quantity	Needs >720 pulses per revolution to attain 0.5 deg. polar alignment spec, cheapest unit is $8.36 magnetic encoder (magnet not included), needs to rotate with the shaft
Accelerometer	Can be placed on surface of rim of shaft knob (easy to install)	Needs to be integrated twice over, requires knowledge of knob dimensions to convert to angular quantity, minimum range on Digikey is 0.5g (very high), cheapest unit on Digikey is $7.84
Gyroscope	Most gyros' BW>100 Hz -> minimum gyro sensitivity is 0.02 LSB/(deg/sec) to obtain 0.5 deg resolution w/ 1 LSB. This applies to most gyros on Digikey. Easy to install (goes on the axis of the shaft).	Needs to be integrated over time to get angular quantity.

Clearly, the selection is a contest between a motor encoder and a gyroscope. Accelerometers appear to be meant for relatively strong signals (on the scale of 1g) and will be difficult to attain the 0.5 deg. polar alignment spec with. We chose to go with a gyroscope because a motor encoder would require a potentially costly adaptor between the declination axis shaft and the encoder "shaft", while a gyroscope can be simply taped/glued to the end of the declination axis shaft.

A specific gyroscope was selected from the offerings on the Pololu website:

TABLE III. POLOLU'S GYROSCOPE OFFERINGS

Part	**Price**	**Voltage range (V)**	**I (mA)**	**BW (Hz)**	**Sensitivity (mdps/ digit)**
L3GD20 3-Axis Gyro Carrier with Voltage Regulator	6.95	2.5-5.5	7	760	8.75
L3GD20H 3-Axis Gyro Carrier with Voltage Regulator	8.95	2.5-5.5	6	758	8.75
LSM6DS33 3D Accelerometer and Gyro Carrier with Voltage Regulator	11.95	2.5-5.5	2	1666	4.38

All of these gyroscopes have the sensitivity-BW product to satisfy the 0.5 degree polar alignment specification, so the selection can be made based on price and power consumption. Since all three choices consume reasonably low amounts of power, we can simply choose the cheapest gyroscope, the L3GD20. However, since this board is being replaced by the newer L3GD20H (the next cheapest option), we choose the L3GD20H instead.

C. Custom Motor Controllers vs. Store-bought

The stepper motor we selected is rated for 3V and a continuous current of 1.7A/phase. To get an idea of the specifications of commercially available motor drivers, we inspected Pololu's online catalog of stepper motor drivers. Only the TB67S249FTG, AMIS-30543, TB67S128FTG, and DRV8711 were listed as capable of supplying 1.7A of continuous current per phase without additional cooling, and the one requiring the lowest supply voltage still needs at least 6V to operate. Therefore, it is necessary to design our own stepper motor driver for a 3V supply voltage in order to support the specific motor we selected.

D. On-Board Computer Selection

Microcontrollers are unlikely to be able to meet the memory and processing power requirements, and a full laptop computer would be cumbersome to attach to the unit, so single-board computers received more consideration. The main candidates for an on-board computer were the Raspberry Pi 3B+ (or a similar model) and the BeagleBone Black development board. Both can run a full Linux kernel and are compatible with any candidate software libraries. Both are powerful enough for basic computer vision applications. Both can use SPI to communicate with peripheral devices and have USB ports for peripherals.

The BeagleBone Black, at $62.38 from Digi-Key[5], is slightly more expensive than the Raspberry Pi 3B+, which is provided at no cost by the course staff. It has 65 GPIO pins, more than the Pi, which has 26 pins. Both have enough pins for the motor controller board, which requires 4 pins per motor and 4 pins for SPI for the gyroscope, which comes out to 16 pins total. Unlike the Raspberry Pi, it lacks a GPU, which would be useful for accelerating image processing. Due to these factors, the Raspberry Pi seems more suitable for the project.

E. Software Tools and Algorithms

For the project, in order to implement object-tracking, it's necessary to be able to identify an object across several images and to be able to identify the movement of that object from frame to frame. In the case that our project is extended to perform drift alignment for polar alignment, an optical flow algorithm is required.

Some libraries for computer vision applications include OpenCV, TensorFlow, CCV (unfortunately known to have issues compiling on the Pi)[7], SOD (an embedded computer vision library with licensing fees for trained models).

Known algorithms for object detection include YOLO[6], Faster R-CNN object detection, and SSD object detection. SSD and YOLO are single-shot object detectors and are typically faster than Faster R-CNN object detection. YOLO is known to struggle with small objects within an image, so it may not be suitable for our application, tracking stars. Object tracking involves taking in an initial object detection and tracking the object as it moves across video frames. It's not necessary that the object tracking algorithm we use be robust to occlusion, because we intend for the object to remain within frame, and the objects we intend to track will be moving at slow rates.

Both OpenCV and TensorFlow have pre-trained models available for free that implement such algorithms. It may be necessary to train our own computer vision models, but this is not preferred, since it can be time-consuming and difficult. We are likely to use OpenCV.

In order to interface with a camera, we plan to use libgphoto2, a library designed to allow access to a digital camera by external programs. It is compatible with many cameras, including some smartphones. Downsides to using libgphoto2 and a user-provided camera include latency of image transfer. An alternative could be to use a Raspberry Pi camera module for visual feedback. This would reduce the latency of image transfer and make calibration of object tracking easier given a consistent position on the mount and a consistent zoom level. However, the Raspberry Pi camera module may have insufficient image quality compared to a user-provided camera, and it may be difficult to consistently ensure that its position and direction corresponds with that of the user-provided camera.

F. Mechanical Considerations

We chose to build an equatorial mount with three degrees of freedom so that we could support longer exposures than with a barn door construction, which has a more limited range and tends to accrue error more quickly. We planned for three motorized axes: one for the sky-tracking compensator to turn along, and two to align the polar axis or assist with object tracking.

We needed to choose a gearing ratio for the sky-tracking compensator. Given that we planned to use a stepper motor with a step angle of 0.9 degrees, we needed to reduce the angle per step so that we could step at a higher frequency and at a higher resolution for the desired rate of 4.178×10^{-3} deg/s (Fig. 3). We chose to use a 20:1 worm gear gearbox from Amazon with a 3D-printed gearbox with a ratio of 32:1, intending to step the motor at 2.971 steps/second.

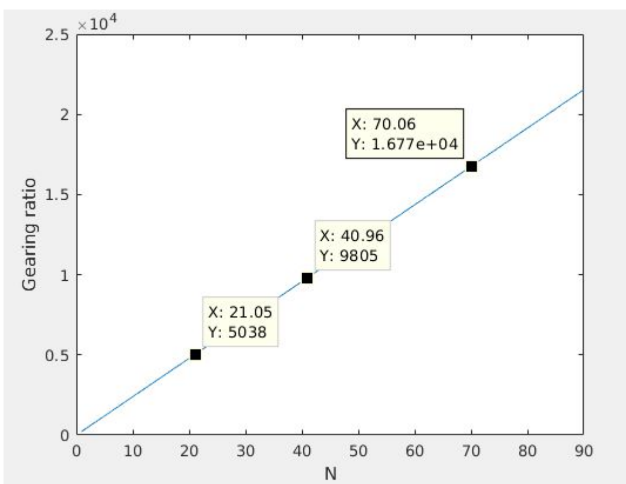


Fig. 3. Gearing ratio vs steps per second for the desired rate.

V. System Description

Our system has two major parts, a circuits subsystem which includes a motor controller board (with the accompanying power distribution setup), and a software subsystem which includes a computer vision component and an embedded component. For testing and verification, we are building a testing rig to simulate the motion of the sky.

A. Motor Controller and Logic Buffers

The equatorial mount will make use of three individual motor controller boards with logic buffered inputs, each of which will be supported by a 8V power supply and a high power 3V power supply. Each motor controller board will possess two H-bridges constructed from a pair of BUK9K89-100E power MOSFETs driven by LM5109B gate driver ICs. Flyback transients will be damped by PMEG60T10ELPX schottky diodes located between power rails and motor coils. The gate driver IC inputs will be merged together into 1 PWM input and 1 enable input per motor coil using SN74HC08 AND gates (for the enable functionality) and delay equalized complementary output CD4041UB buffers for driving the complementary inputs of the gate driver chips. In total, 4 GPIO pins on the Raspberry Pi will be required to control each motor.

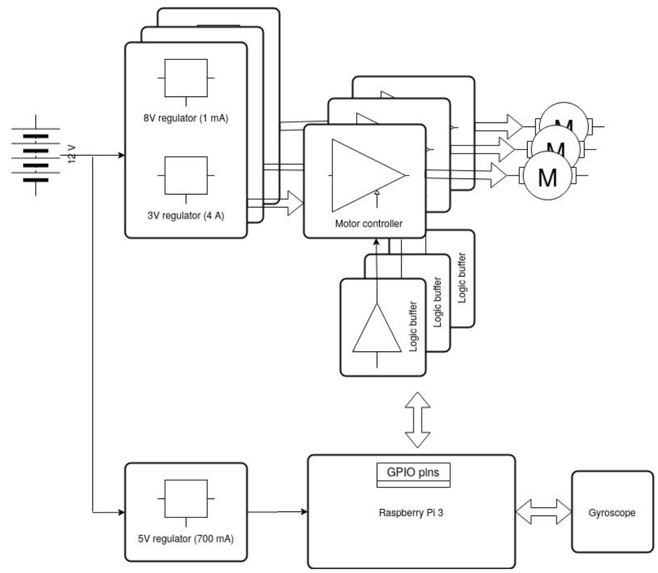


Fig. 4. Block diagram for motor controller boards.

The full schematic of a single H-bridge (2 per motor controller board) is shown below:

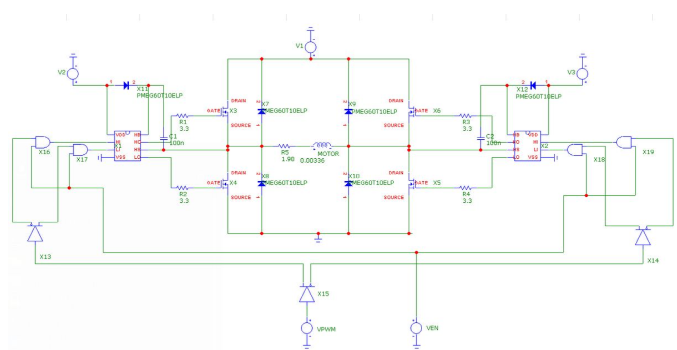


Fig. 5. H-bridge circuit schematic

SPICE simulation with the above schematic indicates a tphl of 1.1 ms and a tplh of 1.12 ms for the PWM input. The ENABLE input has a tphl of 0.4 ms and a tplh of 1.12 ms. These propagation delays for the H-bridge input pins are much smaller than the time periods over which they'll be actuated, making timing constraints less strict for the motor driver software on the Raspberry Pi.

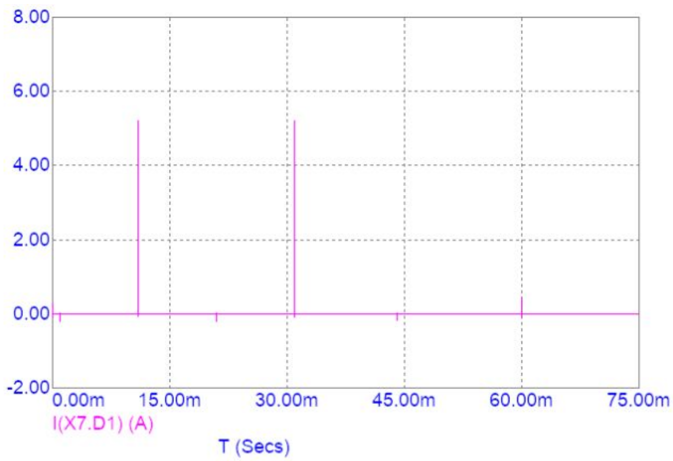


Fig. 6. Flyback diode current spikes

Furthermore, SPICE simulation for the current through individual PMEG60T10ELPX diodes indicates that they'll experience current spikes on the order of 5A while absorbing any voltage spikes from the $\frac{dI}{dt}$ through the motor coil inductances. This is far less than their non-repetitive peak forward current rating of 35A, and is an indicator of the soundness of the design.

B. Power Distribution

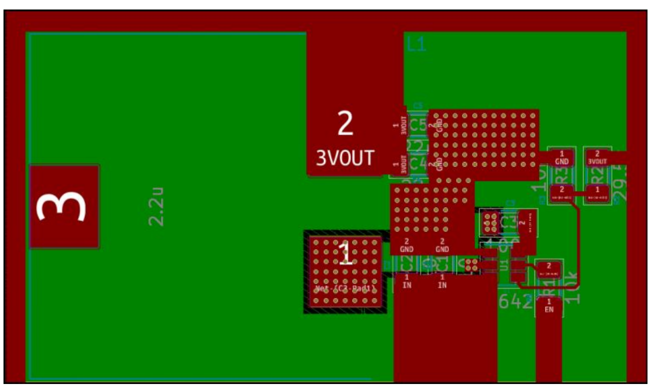


Fig. 7. PCB layout for the stepper motors' 3V 4A power supplies.

A 3V 4A buck converter has been designed and laid out using a TPS564201DDCR buck converter controller IC for powering individual motors. The gate driver chips of the motor controllers are to be supplied with 8V (~ 1 mA quiescent per chip) from a LK112M80TR LDO regulator. The logic buffers on-board the motor controller PCBs and the gyroscope will be supplied from the 3.3V rail of the Raspberry Pi, while the Pi itself will be powered from a AP1186T5-50L-U 5V linear regulator. The Pi's display will be powered by USB cable from the Pi. Lastly, the testing rig's motors will each be powered by 3V 4A TPS564201DDCR-based supplies, while the test rig's laser diodes are to be powered from additional 8V LK112M80TR LDO regulator.

C. Software Subsystems

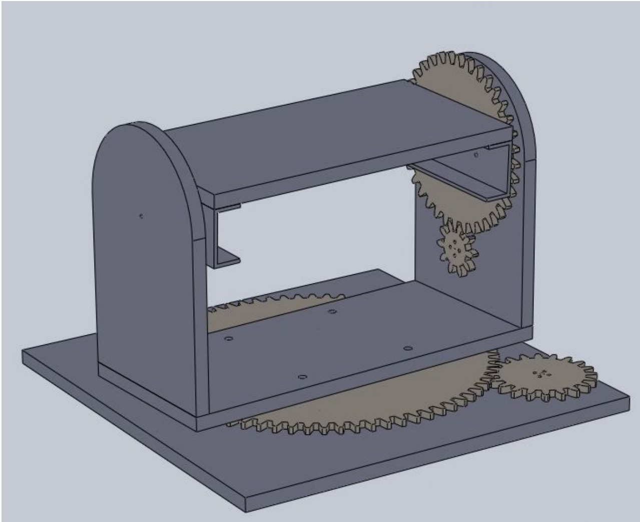
As described earlier in this document, we will be running a computer vision algorithm on a Raspberry Pi which will process images from a user-provided camera and calculate the corresponding inputs to the motor controller board given a user-selected function.

For testing the software subsystem, our main concern deals with object detection accuracy and our ability to accommodate multiple capture types, resolutions, and rates. The most pressing issue is creating a subsystem test for our implementation of OpenCV object detection and movement calculation. For this we will assume (and require) that our sample rate for object detection given by the user is high enough that star and object movement is roughly linear. Using this assumption we are able to generate tests by simply applying linear offsets to captures of desired and scoped for objects and constellations (e.g. Jupiter, Ursa Major) and assessing the performance of our object drift calculations by comparing "inverse" calculations with our generating parameters. Furthermore, due to the large error from sample to sample that will likely be seen for very high sample rates (e.g. 1Hz) we want to explore samples that are more temporally separate, such as two captures that are ~60s apart. For nearby objects (i.e. "large", e.g. The Moon) we expect about half of our aforementioned 5% "large object error" to be attributed to this portion of our system due to the larger margin of error that can be seen when trying to find centers of large objects in a photograph. For faraway objects (e.g. Jupiter) we expect this error to decrease the more time we spend gathering sequential samples. As a result we expect about ⅓ of our 40% small object error (or less) to come from this subsystem.

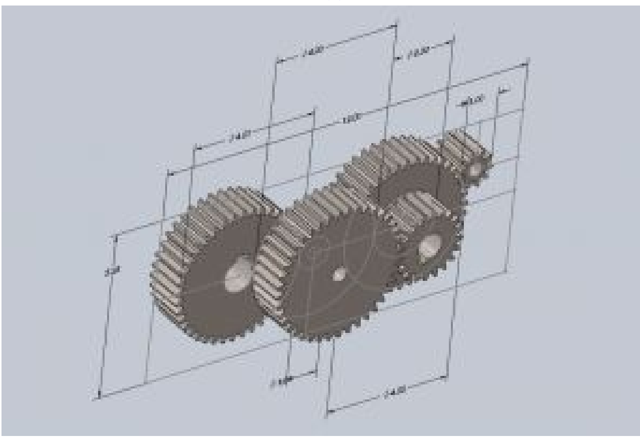
For accommodating different resolutions/capture sizes we plan to use an object-detection mode calibration step where the user first calibrates their camera to a desire capture spec. Then the system takes a few samples with known offsets (offsets that will be known due to a previous star-tracking calibration step) and stores a 1:1 correlation between resolution space and real space. Under the assumption that camera specs do not change or are recalibrated upon change, this correlation is used to map desired resolution space offset detected by CV to actual space offset in the actuators. This will then be tested for rigor given different generated samples and output correlations.

D. Mechanical Subsystems

The base of the mount will be fabricated from MDF. Simple turntable bearings from McMaster-Carr will be used. Gearing ratios for the compensator will be accomplished with a 20:1 worm gear gearbox and a 3D-printed 32:1 gearbox. The gearing for the hinge joint and the turntable will be laser-cut from acrylic. Professor Nace mentioned that the TechSpark makerspace has a gear cutter, which we may use to fabricate gears.



(a)



(b)

Fig. 8. CAD drawings. (a) Mount base.. (b) 32:1 gearset.

E. Testing Rig

Due to the fickle and often inconvenient nature of stargazing we are building a testing rig to help test the overall efficacy of our system. It will consist of an array of 12 laser diodes fixed in a foam ball, which we can turn on a motorized turntable or by hand. The motorized rig will then project these lights onto a surface (either walls or large sphere) and rotate at roughly a sidereal or faster rate. To this projection an additional laser diode can be used to simulate an object that moves at a different rate and direction to help test whole system object tracking. The purpose of this rig is to simulate point light movement projected onto a surface to roughly approximate real sky movement. This is to provide an easy to demo approximation of a real world use case by showing how our system can compensate for object movement and provide a relatively sharp image of normally blurry moving dots.

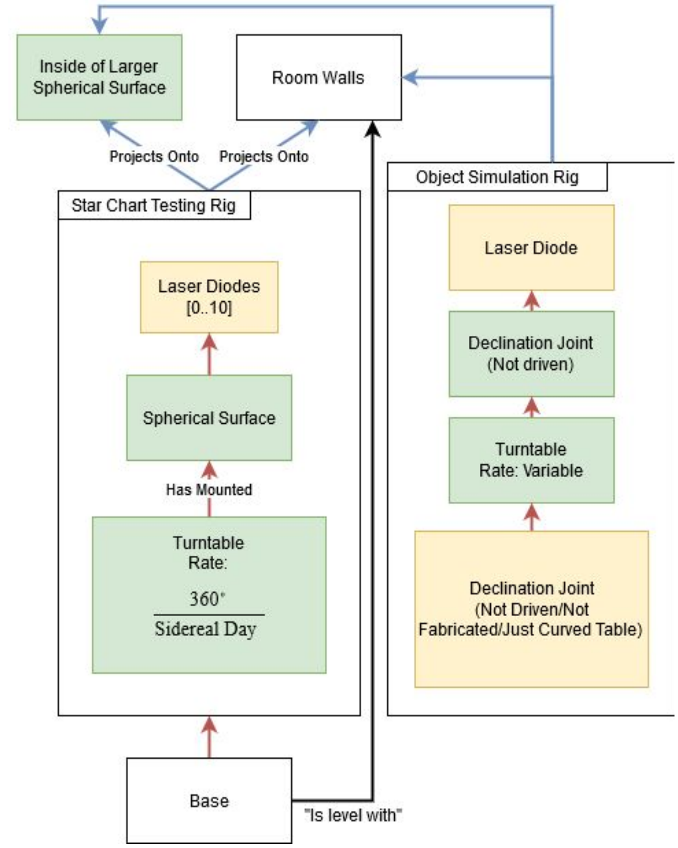


Fig. 9. Block diagram for testing rig. (LEGEND: Green: Fabricated, Yellow: Purchased, Blue Arrow: Projects onto, Red Arrow: Parent-to-child mechanical relation)

VI. Project Management

A. Schedule

The main milestones for the project are completion of circuit layout, construction and assembly of the mount and gearing, prototyping and testing of the computer vision software for object tracking, completion of a testing rig, and integration. It is expected that construction and assembly of the mount, prototyping of the computer vision software, and fabrication of mount circuitry will be completed by the end of March, leaving April for more testing and integration.

Refer to the end of this document for a full Gantt chart.

B. Team Member Responsibilities

Yuyi is responsible for designing and fabricating the motor controller and gyroscope boards, designing and fabricating part of the test environment, and designing and fabricating any necessary circuitry for the user interface.

Kenny is responsible for implementing the user interface and the polar alignment algorithm, designing and fabricating the gearing for the mount, and constructing the test environment.

Joy is responsible for design and construction of the mount, prototyping of the object mapping software, and integration of the object mapping software with the mount movement.

It's anticipated that some tasks will be shared between members and that some task divisions will change over time.

C. Budget

Refer to the end of this document for a full parts list.

D. Risk Management

Some notable risks for our project deal with failure to implement certain subsystems effectively or in a timely manner.

One risk we face is a failure to implement a working motor controller. This can happen due to several factors including stress and voltage spike test failures, and controller-driver interfacing errors. A solution for this subsystem failure would be to forgo implementing our own motor controller. Purchasing a driver controller module as a replacement would pose it's own challenges as power distribution design would have to be altered to accommodate for the new module.

Another subsystem that could fail is the mechanical compensator for star-tracking mode. This could happen due to inadequate gearbox design or large error margins due to aspects like backlash or low final resolution. To remedy this we would move ahead with implementing an easier to fabricate version of a compensator known as a barn door tracker. This version of a compensator works by linearly displacing a planar surface with a threaded rod that runs through one of its ends. This of course pushes some of the technical challenges from fabrication to software as this type of mount has compounding error, a smaller range of operation and must be "rewound" every so often.

A third subsystem that could suffer failures is the computer vision software. If object-tracking is inaccurate, slow, or buggy, or if the latency of image transfer between the user-provided camera and the Raspberry Pi is too high to allow for real-time correction, real-time object-tracking will not be feasible. We can instead implement blind tracking based on known information about celestial objects or given input from the user. We could also repurpose the computer vision for drift alignment for polar alignment.

VII. Related Work

Some similar products have been designed when it comes to polar alignment. For example these two patents[8][9] detail an implementation for a product that uses positioning protocols to derive the user's location and/or altitude-azimuth in respect to the polar axis, in effect allowing it to point the system to stars and constellations inside of a manufacturer defined controller. However, there is no mention of using visual feedback to account for drift or the possibility of tracking objects outside of the manufacturer defined astronomical object database which are both defining parts of our system.

Another patent exists that describes the use of an intelligent motor controller system for a telescope mount[10]. This patent seems to be often used in conjunction with the aforementioned two and it details the use of optical encoders and servos in order to more accurately position a user's telescope. Furthermore, the patent discusses the option to use a brushless mount for communication between the top and bottom of the system in order to eliminate problems that might arise from wires wrapping around the mount as it rotates. These two are both distinct from our system in that we employ the use of stepper motors and a gyroscope in order to derive and affect position, rather than encoders and servos. Furthermore, our system does not scope for the possibility of a full 360° rotation, instead limiting it the polar-aligned compensator to a range of ~120° and a limited single rotation turntable for the bottom.

VIII. References

[1] "The DIY Homemade German Equatorial (GEM) Telescope Mount." *Homemade GEM German Equatorial Telescope Mount - How I Made My Own DIY Mount for Astrophotography*, The Curdridge Observatory, astro.neutral.org/homemade-diy-gem-telescope-mount.shtml.

[2] "The Ultimate Guide to Powering Your Telescope: Behind the Scenes with Celestron PowerTank and PowerTank Lithium Family of Products." *Celestron - Telescopes, Telescope Accessories, Outdoor and Scientific Products*, Celestron, 4 Oct. 2019, www.celestron.com/blogs/news/the-ultimate-guide-to-powering-your-telescope-behind-the-scenes-with-celestron-powertank-and-powertank-lithium-family-of-products.

[3] R., Tom. "How Long Does It Take to Set Up an EQ Mount? - Mounts." *Cloudy Nights*, 13 May 2007, www.cloudynights.com/topic/110141-how-long-does-it-take-to-set-up-an-eq-mount/.

[4] Arnholm, Carsten. "Telescope Drive Tracking Accuracy." *Arnholm.org*, arnholm.org/astro/tracking/index.htm.

[5] "BBB01-SC-505-ND." *Digi-Key Electronics.*

[6] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement." arXiv, 2018.

[7] "Does not compile correctly... - Issue #75 - liuliu/ccv." Github Issues. https://github.com/liuliu/ccv/issues/75

[8] Baun, Kenneth W., et al. *Fully Automated Telescope System with Distributed Intelligence*. US6304376B1, 16 Oct. 2001, https://patents.google.com/patent/US6304376B1/en.

[9] Baun, Kenneth W., and John E. Hoot. *Fully Automated Telescope System with Distributed Intelligence*. US6392799B1, 21 May 2002, https://patents.google.com/patent/US6392799B1/en.

[10] Baun, Kenneth W., et al. *Telescope System Having an Intelligent Motor Controller*. US6563636B1, 13 May 2003, https://patents.google.com/patent/US6563636B1/en.

[11] Canon. *Canon 51JV-IcjWBL Lens Specifications*. https://images-na.ssl-images-amazon.com/images/I/51JV-icjWBL.pdf.

[12] Beish, Jeff. "Design a German Equatorial Mount." *The Association of Lunar and Planetary Observers*, ALPO, www.alpo-astronomy.org/jbeish/Equ_Mount.pdf.

[13] *Canon B1FDpTkTjKS User Manual.Pdf.* https://images-na.ssl-images-amazon.com/images/I/B1FDpTkTjKS.pdf. Accessed 2 Mar. 2020.

[14] Collins, Danielle. "Stepper Motors: Permanent Magnet, Variable Reluctance, and Hybrid." *Linear Motion Tips*, 26 Apr. 2018, www.linearmotiontips.com/stepper-motors-differences-between-permanent-magnet-variable-reluctance-and-hybrid-types/.

[15] Williams, John. "Types of Electric Motors." EE 410/510 - Electromechanical Systems, The University of Alabama in Huntsville, www.ece.uah.edu/courses/material/EE410-Wms2/Electric%20motors.pdf.

TASK TITLE	TASK OWNER	Week												
		4	5	6	7	8	SB	9	10	11	12	13	14	15
		2/3	2/10	2/17	2/24	3/2	3/9	3/16	3/23	3/30	4/6	4/13	4/20	4/27
Fabrication and Mechanical														
CAD Mount (not including gearing)	JG													
CAD Mount with gearing	JG													
Design compensator gearing	KR													
Order parts and supplies for mount	JG													
Constructing Equatorial Mount	JG + KR + YS													
Obtain Camera Adapter + Tripod	JG + KR + YS													
Circuitry														
Motor Driver+Gyroscope Circuit Design	YS													
Motor Driver+Gyroscope Board Layout + Fab	YS													
User interface board layout + Fab	YS													
CV System and Interface														
Polar alignment algorithm	KR													
Interface PA with mount circuitry	JG + YS													
Object Mapping	JG + KR													
Object tracking prototype (with video)	JG													
Integrate with Mount Movement	JG + KR													
Calibrate for mount and camera zoom	JG + KR													
GUI														
Software implementation	KR													
Obtain and install peripherals	JG + KR + YS													
Testing of software implementation	KR													
Verification														
Skychart laser array circuit design	YS + KR													
Skychart laser array board layout+fab	YS													
Assembly of Skychart laser array	JG + KR													
Sky tracking and object tracking tests	JG + KR + YS													
Integration and Additional Testing	JG + KR + YS													

Part Name	QTY	Cost ($)	QTY x Cost	Notes
400-step stepper motor	3	$17.95	$53.85	https://www.sparkfun.com/products/10846
Gyroscope Sensor	1	$8.95	$8.95	https://www.pololu.com/product/2129
Camera Tripod	1	$43.99	$43.99	https://www.amazon.com/Victiv-Camera-Aluminum-Monopod-72-inch/dp/B07JCG1BKY?ref_=RAsinC_Ajax&pf_rd_r=6SZY7V5NNMFRDZXWYJ38&pf_rd_p=318500f8-69d0-541c-ad18-da9e4a3216a6&pf_rd_s=merchandised-search-10&pf_rd_t=101&pf_rd_i=499306&pf_rd_m=ATVPDKIKX0DER
Tripod Camera phone adapter	1	$6.89	$6.89	https://www.amazon.com/AILUN-Rotatable-Adjustable-Compatible-Camcorder/dp/B072KNBV21/ref=sr_1_3?keywords=samsung+galaxy+s9+tripod+mount&qid=1580944043&sr=8-3
Telescope Finder Scope	1	$13.79	$13.79	https://www.amazon.com/Celestron-51630-Pointer-Telescope-Finderscope/dp/B00009X3UU/ref=sr_1_6?keywords=finder+scope&qid=1583169301&sr=8-6
Pi-compatible display	1	$47.99	$47.99	https://www.amazon.com/UCTRONICS-Raspberry-Portable-Capacitive-Touchscreen/dp/B07VV7RL7Y/ref=sr_1_16?keywords=raspberry%2Bpi%2Bscreen&qid=1583167946&sr=8-16&th=1
Square turntable bearing (6031K160)	2	$2.40	$4.80	https://www.mcmaster.com/6031K160
Round turntable bearing (6031K21)	1	$7.76	$7.76	https://www.mcmaster.com/6031K21
4ft aluminum U-channel	1	$6.79	$6.79	https://www.mcmaster.com/1630T292
2 Universal mounting hubs	2	$7.49	$14.98	https://www.sparkfun.com/products/10006

2ftx2ft 1/2" MDF	1	$4.70	$4.70	https://www.homedepot.com/p/1-2-in-x-2-ft-x-2-ft-MDF-Project-Panel-205641/205922607
8mm shaft ballbearing (608ZZ) (PKG 10)	1	$6.00	$6.00	https://www.amazon.com/Shielded-Skateboard-Bearings-Longboard-Scooters/dp/B07DZDLB3N
(PKG of 10) Stainless Steel Phillips Flat Head Screws, 1/4"-20 Thread Size, 4" Long	1	$11.03	$11.03	https://www.mcmaster.com/91500A363
Worm Gear Gearbox NMRV-030 Speed Reducer Ratio 80:1 or 20:1 for Stepper Motor	1	$59.96	$59.96	https://www.amazon.com/NMRV-030-Reducer-NEMA23-Stepper-Gearbox/dp/B06XGX19BK
Motor driver components	3	$18.76	$56.28	https://www.digikey.com/short/zq123q
Motor power supply	3	$8.18	$24.54	https://www.digikey.com/short/zq1231
Clear Cast Acrylic (2ft X 2ft X 6mm)	1	$15.70	$15.70	IDeATe Lending
ABS filament	1	$10	$10.00	Approximate, given that 1kg of ABS filament is around $30
#8 bolts (40mm) and nuts	16		$0.00	
M6 bolts (40mm) and nuts	8		$0.00	
#4-40 screws, nuts for mounting hubs	12		$0.00	
Circuit Board Fabrication	3	$20	$60.00	oshpark.com
8v Power supply (Voltage Regulator + Caps)	3	$0.79	$2.37	https://www.digikey.com/short/zq1293
Laser Diodes	12	$2.49	$29.88	https://www.goldmine-elec-products.com/prodinfo.asp?number=G19031
Laser Power supply	12	$0.79	$9.48	https://www.digikey.com/short/zq1293
5v Raspi Power supply	1	$18.50	$18.50	https://www.digikey.com/short/zq1wch
TOTAL COST			$518.23	

Tools	Price	
OpenCV	$0	https://opencv.org/
libgphoto2	$0	http://www.gphoto.org/proj/libgphoto2/
Microcap Circuit Simulator	$0	http://www.spectrum-soft.com/index.shtm
KiCad	$0	https://kicad-pcb.org/
LtSpice	$0	https://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html
SolidWorks	$0	https://www.solidworks.com/
Qt	$0	https://www.qt.io/