Motorized System for Plant Root Research

Jimmy He (miaoh2) Nachiket Joshi (njoshi4)

Feb 7 2019

Introduction

Objective

This proposal covers a motorized imaging system that takes photos of a tall plant's root by transporting a camera down a standard, transparent observation tube into the soil, taking multiple pictures from the ground up, and outputting a panoramic image of the entire root, which is used for scientific research in the field of agriculture. The imager device is comprised of a base station on the ground resembling a hoist, and a suspended camera placed into the observation tube, which itself has motors that centers it laterally in the tube so it always takes images facing up. A central control server serves to control, manage and collect images from a fleet of imagers, and also to present a GUI to the user with live progress and diagnostics data from each imager.

Background

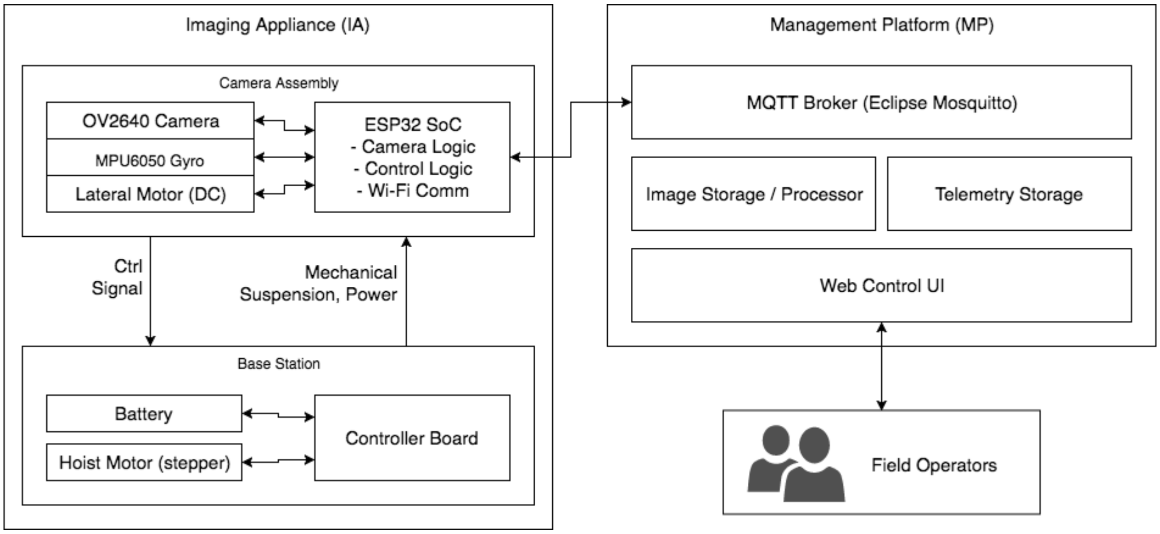
This project is done in collaboration with the College of ACES and SoyFACE Farm (The collaboration has been confirmed.) The SoyFACE Farm contains a corn research facility, where an observation tube is installed by each of the over 1000 corn plants, that goes 5ft deep. Each week, researchers would collect a panoramic image of the roots of each plant to access its health condition. The imaging process is implemented with a bulky camera mounted on a 5ft-long rigid stick. The operator mounts the base of the stick on the observation tube on a fixed mounting point, and inserts the stick deep into the tube. The camera is connected to an equally bulky control box consisting of a laptop, a large car battery and control circuitry of the camera crammed inside of a Pelican case. The camera depth can be read from a ruler on the stick, which the operator needs to input to the laptop, and invoke the "Start" command. The laptop will then verbally instruct the operator to pull the camera up centimeter-by-centimeter at a set interval (usually 1 second), taking a picture at each instruction until the camera is completely out. Any non-compliance of the verbal instruction will ruin the image and require a restart. The set of images are then taken to an external program to be stitched into a panoramic image. The same time-consuming and strength-demanding exercise is carried out over each of the 1000+ corn each week, and the research group demands for an automated solution.

High-level Requirements List

- The system must be fully autonomous, performing all functions without human intervention beyond placing the system in the observation tubes.
- The system must be scalable, allowing multiple imaging appliances to operate at the same time while being managed by an operator.
- The system must be as low cost as possible, with a target cost of $100 maximum for each imaging appliance.

Design

Block Diagram



Physical Design

The following are drawings of the mechanical construction of the devices.

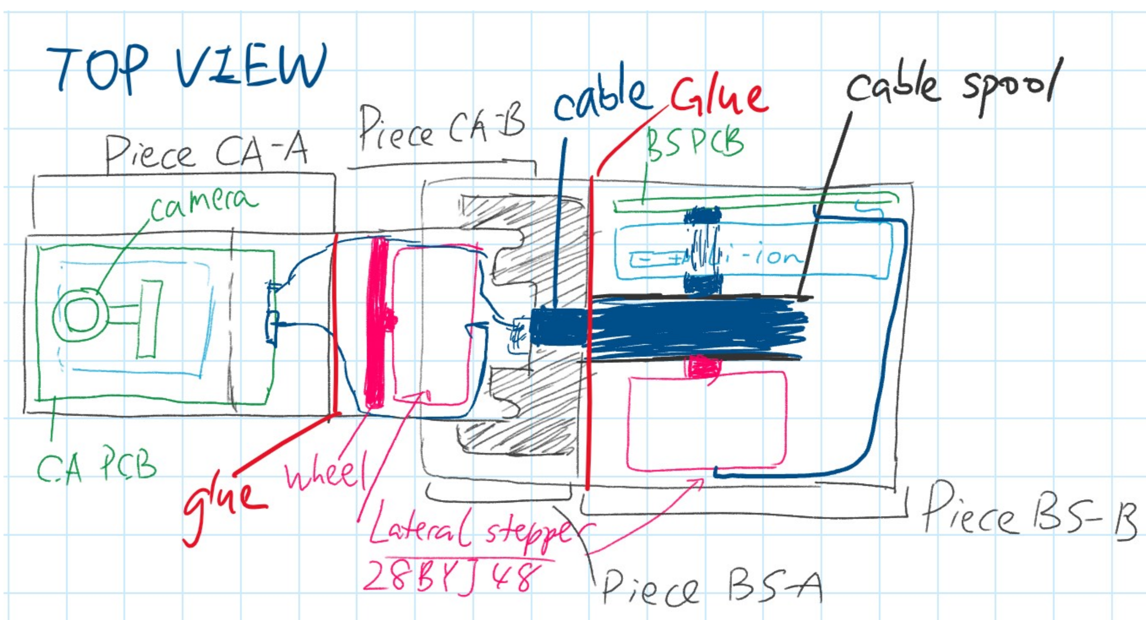


Figure 1. Top view of the design

Functional Overview

The larger system can be split into two components, a Management Platform (MP) and an Imaging Appliance (IA). The IA can be further split into the Camera Assembly and the Base Station. The following

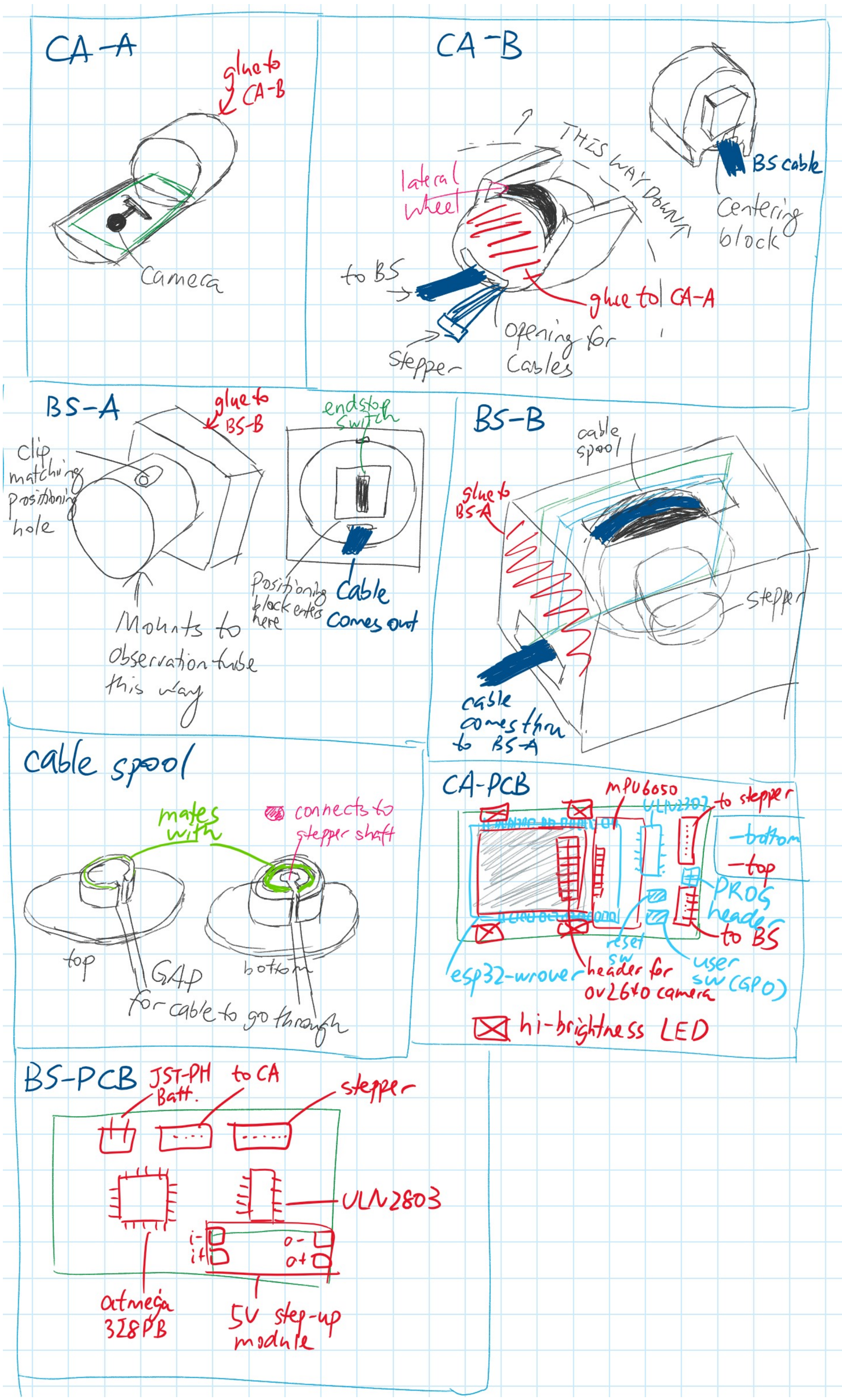


Figure 2. Detailed drawings and sketches

discussion about the components is centered around the block diagram.

The main logic of IA runs on top of an ESP32 SoC inside the camera assembly. It accepts control signal from the operator from the MP, through a MQTT connection over its built-in Wi-Fi. It also has a serial (RS-485) link to the control board in the Base Station, in order to receive battery condition data and transmit signals to control the hoist motor, which in turn manipulates the physical depth of the camera in the tube. The Camera Assembly also has a lateral motor assembly with wheels, that centers the camera view in the tube as the camera could have shifted during the up/down motion. This shift is sensed from the built-in accelerometer and corrected using PID technique. The ESP32 SoC is connected to a OV2640 camera, that captures a 2 mega-pixel image at every run. It is fixed-focus since the distance from the camera to the plant root is known. When imaging, the camera first travels up by a fixed distance (~2cm) such that the view overlaps with the previous, and takes a image. It then compresses the image into JPEG and transmits it to the MP.

Block Requirements

The base station (BS, thereafter) of the IA is placed above ground, preferably mounted on top of the observation tube. It contains a battery to power the entire IA, and a stepper motor that hoists the camera assembly (CA, thereafter) into the tube. The camera assembly sends control signal in terms of distance to move, and the ATmega328 MCU in the ground station translates it into angular motion before commanding the stepper motor. It also has an endstop switch to indicate the home (Z=0) point. The MCU is also connected to a battery charging / monitoring IC that can charge the battery and report battery level to the camera assembly, such that it can home itself and refuse imaging when the battery level drops low.

The Management Platform (MP) runs on a physical machine of any platform (such as an x86 Linux server) and optionally also act as the Wi-Fi AP for each IA to connect to. The heart of the MP is an MQTT broker collecting telemetry and images from and emitting control signals to one or more IA'ds. Upon reception of images, it processes the image set and stores a panoramic image into its image storage. Upon reception of telemetry, it stores the telemetry in a volatile database for tracking. It exposes a Web UI to the frontend users, such that the users can view and download the images, as well as monitor and control the IA's. When used, the operator enters the name of the image (matching the current date and ID of the observation tube) and presses "Start", starting the automatic imaging sequence of the selected IA. The operator then monitors the Z depth of the camera in real time as an indicator of the imaging process. When done, the operator may download the panoramic image already stitched from the camera.

The below is an elaborated specification and verification methodology for each requirement in the block level.

Imaging Appliance - Base Station

The base station is an equipment fixed to the observation tube. It is a passive device communicating over a serial interface with the Imaging Appliance. It supports the following single-byte commands:

- `H` for HOME, which moves the camera assembly up into a home position. It returns a byte `K` for `OK`, when fully homed.
- `S` for STEP, which moves the camera down for just a set distance and stops. It returns a byte `K` for `OK`, when motion stops. It returns a byte `E` for `ERROR` if the number of STEPs has exceeded the limit.
- `C` for COUNT, returns the number of STEPs the camera has performed since last homed.
- `P` for POWER, returns the current battery voltage as a floating point number.
- `X` for CANCEL. Stops whichever activity the base station is doing.

Requirement	Verification
BS performs HOME per spec.	Send H over serial. Check that BS begins motion and stops when CA is fully homed. At which point, expect K received over serial.
BS knows that a device is already HOME'ed.	Send H over serial. Expect K received over serial immediately, and no motion has occured over the motor.
BS performs STEP per spec.	Send S over serial. Expect BS to begin motor motion downwards. Expect BS motor motion to eventually terminate and receive K when it happens.
BS performs STEP repeatably, with a distance error of max <=30%.	Place a ruler along the tube the camera travels down. Send multiple consecutive S commands, each waiting for the motion to complete and a K return to be issued.
BS returns E for STEP when step count exceeds limit.	Set a hypothetical limit (10) in software. Send 11 S's over serial. Expect only the first 10 commands to result in motion and a K return. Expect the 11th command to result in no motion and a E return.
BS reports the correct number of STEPs when received COUNT.	Randomly assign an integer N <= 10. Issue N S commands, each waiting for the previous motion to complete. At the completion of Nth command, issue a C command. Expect to receive an integer over serial, and the integer is equal to N.
BS clears COUNT when HOME is issued.	Perform the above verification, and issue an H command. Wait for motion to complete. Now issue a C command. Expect to receive 0 over serial.
BS reports the correct voltage for POWER.	Disconnect the battery, and place a voltage-dividing potentiometer that ranges from 0.0V to 3.3V. Set the potentiometer to a fixed value, and read the voltage from a voltmeter. Issue a P command and expect a floating-point integer to be returned on serial with a newline character. Expect this number to match the voltmeter reading.
BS ignores commands when in motion, except the X command.	Issue multiple S commands to BS in rapid succession, before the motion resulted from the first S command had stopped. Record the distance moved as D1. Issue a second S command, wait for motion to end and record the distance moved as D2. Expect D1 to be within 20% of D2. Issue an S command again to start the motion. Issue a X command right after. The motion should stop immediately, and a K will be received.
After getting the X command, BS ignores all other commands until successfully homed.	Issue an S command. While in motion, issue X. Expect to receive a K and motion stops. Now issue S multiple times, expect E return and no motion each time. Issue H and expect BS to home the CA. Once fully homed and K received, issue S commands and BS should generate motions accordingly.

Requirement	Verification
The power supply provides a stable 5V voltage rail with an accuracy of 0.3V.	Measure the 5V rail using a voltmeter, from the VCC and GND lines of the interconnect between the BS and the CA. Expect the reading to be between 5.3V and 4.7V. Repeat the measurement with lithium battery cells measured at 3.0V, 3.7V and 4.2V.

Imaging Appliance - Camera Assembly

The camera assembly consists of the main logic SoC, a lateral alignment stepper motor backed by a MPU6050 IMU and the OV2640 camera module. It performs the following tasks:

Requirement	Verification
Accept an "Image" command over MQTT, and start executing a sequence to collect images.	Send an "Image" command over MQTT to the SOC and observe the following sequence. The verification of the sequence can be done by hooking the CA directly to a computer over serial, bypassing the motor assembly.
During image collection, periodically transmit status information over MQTT. The status information contains battery voltage level, that it is imaging (not idle) and the number of steps so far traversed.	After the "Image" command, observe from the MQTT broker that voltage level, the "imaging" status and number of steps are correctly reported.
Before imaging, issues a `HOME` command to the BS at the beginning of the collection sequence and waits for a `K` response for confirmation. At this point, it zeroes the roll value of the gyroscope.	Observe the outputs over the CA's serial port. Expect an `H` from this port, and no further outputs should be expected until a `K` response is manually sent to the CA.
When imaging, it continuously issues N `STEP` commands to the BS. Each time after a `STEP` command, it waits to receive a `K` response, then commands the lateral motor to move sideways to zero out the gyroscope roll value, takes an image and sends the image over the MQTT broker.	Observe the outputs on the serial port. Expect a `S`. Expect no motion of the lateral motion until a `K` is sent manually. Move the unit laterally and send `K` over serial, and it should correct itself. Also observe the MQTT broker, that before the next `S` command is received on the serial port, it should transmit the image taken just now over MQTT.
After imaging, it issues a `HOME` command again to be transported back to home.	Observe the serial port, expect an `H` command. When a `K` response is manually given, expect the status information sent over MQTT to denote that the unit is idle.

Management Platform - Web UI

The Web UI is an interface to the underlying Image Processor and Telemetry Processor, and it allows the user to perform the following actions:

- Command the IA to start capturing images.
- Command the IA to cancel imaging and return home.
- View the status of the IA, whether it is idle or imaging.
- Check and download the raw images, and the stiched panoramic images.

Since the Web UI is an inherent part of the MP, verifications of the Web UI are combined with those of the underlying components of the MP as follows.

Management Platform - Telemetry Processor

The telemetry processor is able to do the following:

Requirement	Verification
Track and display the current status of the IA. (i.e. whether it is idle or imaging + battery voltage)	Connect a mock IA (software simulated) to the MQTT broker. Under the same protocol of a real IA, periodically send out status packets entailing the information that the IA is idle, with a battery voltage that varies from 3.3 to 0V randomly from packet to packet. Check from the MP that the change in battery voltage is correctly indicated.
Periodically update a flag that tells whether an IA is online	Connect a mock IA to the MQTT broker. Under the same protocol of a real IA, periodically send out status packets like above. Expect that the MP indicates the device is online. Now disconnect the IA from the MQTT broker. Wait for 30 seconds, and expect that the MP indicates that the device is offline now.
Send imaging and cancellation command to the IA.	Connect a mock IA to the MQTT broker. Periodically send out status packet as above, to indicate to the MP that the IA is now online. Expect that the Start button on the MP is now operable. Click on the Start button and expect that the IA receives a command over MQTT to start imaging. Now periodically send out status packets from the IA to indicate that it has started imaging. Expect that MP is also indicated that the IA is in progress imaging. Click on the Stop button on the MP. Now expect that the mock IA receives the Cancel command over MQTT.

Management Platform - Image Processor

The image processor is able to do the following:

Requirement	Verification
Receive images from the IA over MQTT	Connect a mock IA (software simulated) to the MQTT broker. Send over an image of the same format taken by a real IA, under the same transfer protocol. Check that the MP's log indicates successful receiving of the image. Alse check that the raw image appears under the filesystem MP runs on, and that the raw image is identical to the one being sent using diff.

Requirement	Verification
Stitch a batch of images into a panorama	Prepare a suite of images captured from a long object. Connect a mock IA (software simulated) to the MQTT broker. Start a periodic transmission of status packets over MQTT indicating that it's idle. Click on the Start button to simulate start of imaging process. Now send in each of the images in the batch in order over MQTT. After sending each of them, transmit the indication that imaging is complete over MQTT. Check the MP log to expect that images received are stitched, and a stitched image appears in the filesystem that the MP runs on. Expect that the stitched image is a valid image file, and it contains the unique geometry of each test image in the batch. Upon visual inspection, no dicernable distortion of geometry should be found in the stitched image compared to raw images.
Serve the stitched image or the raw images over HTTP as a file download	After completing the above verification, expect that the test run conducted shows up in the MP's Web UI. Click on the corresponding buttons to start downloading the zipped raw images and stitched image. Expect that the images are downloaded from the browser. Using the UNIX `diff` tool, compare the downloaded images with the ones in the MP's filesystem, and expect them to be completely identical.

Criterion for Success

The overall effectiveness of the project can be assessed in three aspects: Functionality, Repeatability and Effectiveness.

Functionality can be measured by having an imager device take a panoramic picture from a real plant, and checking the following. First, a valid panorama should be returned, and each image comprising the panorama must not be yawed more than +/-15 degrees from each other to certify the lateral motion compensation. Second, the central management system should report real-time progress from the imager at all times.

Repeatability can be measured by comparing two consecutive images taken from the same plant, and there should not be significant differences in the geometry of the images and features.

Effectiveness can be measured by having the entire imager cost less than $200, without a complex manufacturing procedure that consumes more than 2 man-hours in assembly.

Success of the project can be certified if the above criteria are met.

Risk Analysis

The most risky requirement is the lateral centering capability of the CA. This is a careful choreography between the sensor, the motor, the physical construction of the rotor, the shape of the enclosure as well as the algorithm handling the control. The hardest parts that require tweaking are the following:

- The degree of friction between the rotor (lateral wheel) and the carrying tube, which dictates the effectiveness of lateral control and the tendency to veer off center during downward motion.
- The criterion of "center" of the control algorithm. What is a threshold? How to handle the anomaly case where a single step causes the CA to veer from off the left center to off the right center?
- Tuning of the outer enclosure to make sure no self-spinning of the module occurs around the direction of advance, while the rotor can effectively control the attitude of the camera.

Hence, this requirement poses the most risk among all requirements of this project.

Ethics and Safety

Safety

There is a certain level of safety that we must maintain while developing the hardware for the imaging appliance. First of all, we will be using a lithium ion battery, and lithium ion batteries can be volatile if we use it outside the batteries operating range. Our voltage regulator has to keep itself sufficiently ventilated, and operate within the acceptable temperature range according to its specification sheet, and the charging circuitry must keep the battery from overcharging or overdrawing, both in terms of acceptable current and voltage. In addition, benchmarks of the power-hungry components such as the motors, high-brightness LED lights and the ESP32 SoC must be done assuming worst-case condition, to ensure the battery selected can withstand the sum of current draw from these components.

Since the imaging appliance is designed to go into a tube that goes underground, it should have some level of water resistance. These tubes are covered, so having a minimal level of water resistance should be sufficient for our purposes.

While lithium batteries are a hazard when disposed of in the landfill, the EPA does not regulate the disposal of such in small quantities (40 CFR PART 273). However, the casing design should ensure that the lithium battery is easily removable from the case. Aside from disposal, the lithium battery removed from the appliance, and transported in land-only methods (IEC-61960) with a UN-3480 hazadous material marking on the package.

Ethical considerations

Our project is primarily designed for academic and research purposes. Our ethics standards must therefore comply with the standards that the project itself maintains.

According to the ACM Ethics Guidelines, the purpose of this project is to contribute to the society (1.1) and to be honest and trustworthy (1.3). As a device deployed in a research environment, we pledge to ensure the solution is sufficiently tested and benchmarked, and the accurate performance metrics are reported to the user. Exhaustive verifications are also required to ensure the data produced from the device exhibit the accuracy and precision as specified in this proposal.

As such, our testing procedure evident from the Block-level requirements ensure that the system is both tested under real-life scenarios and close-to-real-life mockups in verification stages, and pilot tested in the actual research facility it is deployed in. The system under test will be subject to a comparative trial, in which both the image produced from this system and the legacy camera device currently in used will be audited both by researchers themselves, and algorithmically compared, to ensure equivalence of the devices during their transition.