Automatic Follow Focus

Ethan Lerner, Gabriel Beal, Dayyan Sisson
University of California, San Diego
{elerner, gmbeal, dsisson}@ucsd.edu

June 9, 2021

Abstract

Professional-grade consumer cameras are used by more and more cinematographers for a variety of projects, ranging from amateur shoots, to mid-size productions. While these cameras are developing better digital autofocus, these systems only work with compatible lenses. Our solution is an automatic follow focus system. It would operate much like an industry standard manual follow focus system, but would automatically keep a target subject in focus without the need of human intervention. To create an automatic follow-focus, our system will need to be broken down into three main components: the follow-focus-rig, the depth-sensor, and the computer. The follow-focus rig would provide the mechanism for focusing a manual camera lens, and will follow the same structure and components of industry standard manual follow-focuses.

1 Introduction

One of the most difficult things to do when shooting video is keeping your subject in focus. Low-end consumer cameras (like the ones found on your phone) mitigate this by using lenses with fixed lenses that have use deep depth of field (see Figure 2) and autofocus. Consumer grade professional cameras (dubbed "pro-sumer" cameras) such as DSLRs and mirrorless cameras with interchangeable lens mounts do not control explicitly for the depth of field, but oftentimes still come with autofocus that is reliable enough for still photography and simple videos like vlogs. Narrative filmmaking is the one domain that lacks severely in terms of keeping reliable focus. Larger setups make use of manual follow focuses. Still, the current solutions suffer from a few major issues:

- Many narrative cinematic styles rely on shallow depth of field (see Figure 1), which requires constant and subtle refocusing with moving subjects (relative to the camera).
- Narrative filmmakers often use vintage/cinema lenses which do not come with autofocus capabilities. Even if they do, higher end cinema cameras do not come with reliable autofocus either.
- Manual follow focuses often require an additional dedicated and highly skilled operator (called focus pullers) that must move along with the camera operator. This makes mobile setups incredibly cumbersome and in many cases, impractical.

To solve this issue, our paper proposes an automatic follow focus system. An automatic follow focus would consist of three main components: the follow-focus-rig, the depth-sensor, and the on-board computer. The depth sensor would be responsible for determining how far away a subject is from the camera. The focus-rig would be responsible for actually pulling lens focus. The on-board computer would be responsible for controlling the focus rig based on depth information from the sensor. For this paper, we used a LiDAR time-of-flight sensor for our depth sensor, a manual follow focus system retrofitted with a stepper motor and motor controller for the focus rig, and an Arduino Uno as the on-board microcontroller. Additionally, our system uses a calibrated focus-distance (FD) algorithm to translate subject-distance to steps (where steps are the amount of steps the stepper motor needs to take). Our paper proposes a way of determining an FD algorithm for any lens and focal length. Finally, we built and tested our system on a Panasonic Lumix GH3 camera with a Nikon 50mm f/1.8 prime lens.

Figure 1: Shallow Depth-Of-Field, [1]

Figure 2: Deep Depth-Of-Field, [1]

2 Technical Work

2.1 List of Components

- 3D printed Carbon Fiber frame and attachments
- TF-LUNA LiDar Sensor with 8m range
- 28BYJ-48 ULN2003 5V Stepper Motor with Driver Board
- Adjustable Mechanical Follow Focus
- Arduino UNO

2.2 Constraints

Our main constraint is the fact that one of our team members is located out of state so his contributions on the hardware side are limited however to compensate for this he will be leading most of the software development. He is also the member with the most knowledge of camera systems so naturally this has led to minor slow down in assembly of the embedded system. Once the system was fully assembled we ran into some setbacks. The most notable being that we had to manually approximate the focus-distance function. This function maps a distance from the lens to an amount that the lens has to rotate to bring a subject at that distance into focus. To do this we manually placed objects at a known distance and counted the number of motor steps required to bring the subject into focus. We were able to approximate the function however it is not perfect.

2.3 Implementation

There were several steps to consider when implementing a solution for automatic follow focus. Firstly, the hardware needed to be easily mountable, quiet, and reliably adaptable for many cameras. Next, the software would need several modes as well as a way to calibrate the algorithm in order to account for the many different types of lenses.

2.3.1 Hardware - Mount

The mount went through many iterations shown in Figure 4. First, we used a steel plate to model the basic dimensions of our mount. This served as a good estimation and made structural errors obvious, though the precision was not refined enough to directly model with 3D printing show in Figure 5. Therefore we had to iterate on the design, adding mounting holes and extending the frame where necessary, eventually landing on a finished model. These iterations were done in standard PLA for speed, convenience, and cost. The final product, however, was printed in ONYX material, a nylon-carbon fiber hybrid which is surprisingly light and strong without being too brittle. Though unnecessary, this step turned out to be worthwhile as our heat dispersion was initially non-existent and some PLA pieces melted.

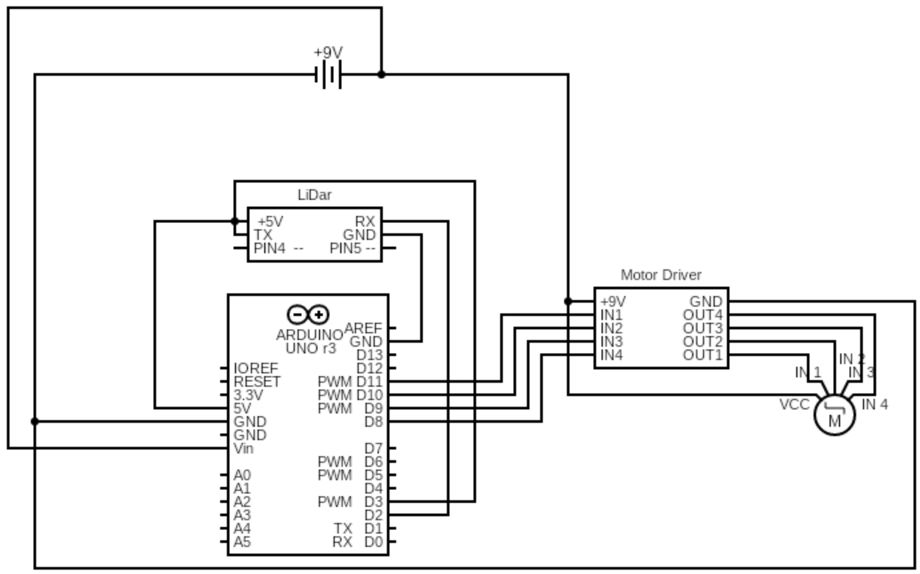

Figure 3: Circuit Diagram

2.3.2 Hardware - Electronics

The majority of our system remained the same and required very little iteration. We began with an Arduino Nano, a stepper motor and driver, and a LiDar. We ended up needing to swap the Nano for an Uno as the former was having port issues when uploading software. The stepper motor and driver turned out to be very effective and simple to use with the one caveat that there was no heat sink and thus caused overheating. This turned out to be simple enough to solve via software by cutting power to the motor when not in use. The whole system is powered by a 9V battery, anything less caused insufficient torque in the motor. The Circuit Diagram in Figure 3 shows all the connections necessary to reproduce this system.

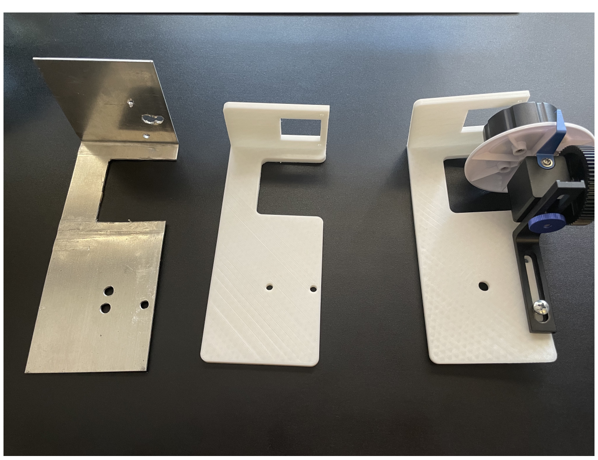

Figure 4: Hardware Mount Iterations

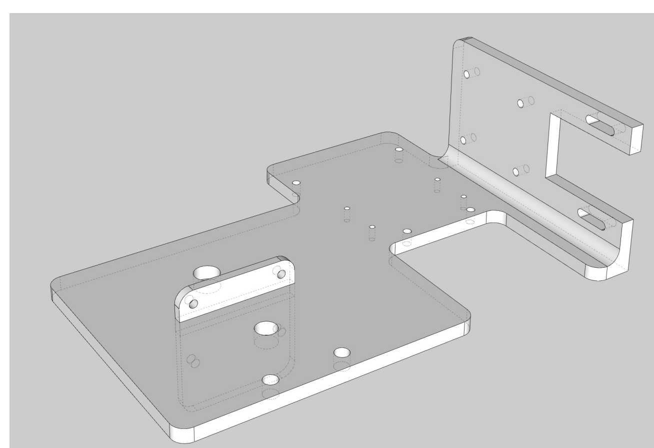

Figure 5: Final 3D Model

2.3.3 Software - System Control

To test individual system components as units and integrated, we wrote a testing suite that allowed us to control the system directly. The testing suite allowed us to control the system in these ways:

- Trigger the motor off/on
- Set the amount of steps we want the stepper motor to take once activated in either direction. This includes large steps and fine-grained incremental controls
- Set the motor focus to the distance the LiDAR sensor currently reads
- Set the motor's RPM
- Turn on/off the LiDAR sensor readings
- Turn on/off Autofocus Mode
- Turn on/off debug mode

2.3.4 Software - Focus Distance Function

For the focus distance function we had to manually approximate the function since it was not listed for the lens. The lens was set to f1.8 which meant that the focus was the most sensitive and would give us the most accurate results. First we had to find the closest focus point of the lens. This is the closest an item can be and still be in focus which varies from lens to lens. In our lens' case this distance was 42cm. That was our starting point and from there we placed objects at 15cm intervals and counted the number of motor steps it took to bring the item into focus. We did this part by giving the motor steps that incremented by 100s and 10s. For example at a distance of 60cm it took 410 motor steps to bring the subject into focus. Below is a table which maps the distances and motor steps that we tested. Through our testing we found that we would need two functions to best approximate the total lens function. At under 2m the function was polynomial with large increments of steps between each distance however the farther a subject was from the lens the more the function became logarithmic. Once we had the measurements in excel we graphed the results and used trend lines to approximate a short range function (42cm-2m) and a long range function (2-8m). Short range: $-1267 + 41.1x + -0.269x^2 + 5.97E - 04x^3$, Long range: $414 + 3.86x + -0.00624x^2$. We had difficulty maintaining the small incremental testing at longer ranges due to the inaccuracy of the LiDar sensor past 2m.

Distance (cm)	Steps
42	0
60	410
75	580
90	680
105	760
120	810
135	850
150	880
165	910
180	930
195	940
300	1390
400	1450
500	1470
577	1480

2.3.5 Software - Autofocus

The software uses the focus-distance function to determine how many steps the motor should take to put a certain subject in focus. Once turned on, the system keeps track of where the motor is at in its rotation, and, using live readings from the LiDAR sensor, determines how manys steps the motor needs to take in either direction to put the subject into focus. The motor then is activated and linearly interpolates to that point. This happens automatically, and without any input from the camera operator.

3 Milestones

Ahead of schedule
On schedule
Behind schedule

3.1 Hardware

3.1.1 Circuit Design

This milestone involved drawing up a wiring schematic to determine which wires connect to which pins. (i.e. the I/O of the different components of the system). This was the first item completed once the hardware items came in.

3.1.2 Circuit Assembly

This milestone involved connecting the system components together. The connected components were: stepper motor to the motor controller, motor controller to the micro-controller(Arduino UNO), and the LiDar sensor to the micro-controller(Arduino UNO).

3.1.3 Mount Design

This milestone was very iterative. Our original mount was completed soon after the Circuit Assembly milestone but this was a very rough estimation that was done in steel. Once we had an idea of the shape and size we begin to iterate through 3D models to find the one that would work the best for the entire system.

3.1.4 Mount Assembly

This milestone again was iterative along with Mount Design. We went through many iterations which involved mounting the hardware to each one to determine where improvements could be made. The final iteration and mounting was done by week 8.

3.1.5 Full Assembly

This milestone represents a completion of all the hardware milestones. This was accomplished slightly behind schedule due to underestimating how long the mount design would take. Below is a table outlining when each milestone was expected to be completed and when we actually accomplished the task.

Milestone	Expected Completion	Actual Completion
Circuit Design	Week 4	Week 4
Circuit Assembly	Week 4	Week 5
Mount Design	Week 5	Week 7
Mount Assembly	Week 5	Week 8
Full Assembly	Week 5	Week 9

3.2 Software

3.2.1 Motor Controller

Communication between the Ardiuno UNO and the motor driver. Done using the stepper motor libraries from Arduino.

3.2.2 LiDar Controller

Communication between the Ardiuno UNO and the LiDar sensor. Done using the documentation that came with the sensor.

3.2.3 Optical Algorithm

This involed implementing two functions. (1) Function with input: distance from the LiDar, Ouput: focal length. (2) Function with input: focal length, Output: motor control.

Milestone	Expected Completion	Actual Completion
Motor Controller	Week 7	Week 4
LiDar Controller	Week 7	Week 5
Optical Algorithm	Week 7	Week 10

3.3 Testing

3.3.1 Manual Objective Testing

A recorded testing process, which includes manually putting a subject in focus, marking focus ring rotation, subject distance, and then verifying these numbers with the automatic system.

3.3.2 Optical Subjective Testing

Take pictures and video and visually verify subjects remain in focus. This was used to estimate the focus distance function.

Milestone	Expected Completion	Actual Completion
Manual Objective Testing	Week 9	Incomplete
Optical Subjective Testing	Week 9	Week 10

4 Conclusion

4.1 Summary and Contributions

Optical focus in video production is one of the most important issues filmmakers and videographers face today, particularly for manual focus setups common in high end cinema. We were able to build an automatic follow focus system which, using depth information from a fixed subject, was able to automatically pull manual lenses into focus. The system, albeit in its rough stages is a contribution to the small list of effective solutions to the issue of focus in cinema. In addition to the system hardware, we were also able to devise and demonstrate a way to approximate any lens' focus-distance function, which can be used as a calibration step that would allow this and similar systems to be adapted for use with a wide variety of lenses and focal lengths.

4.2 Future Work

For future iterations of this system we have many options that we want to explore. Most importantly we want to research into more accurate methods of obtaining or generalizing the focus-distance algorithm that we explored during this quarter. Without being able to generalize this for use on a variety of lenses the system has minimal viability. We have brainstormed many methods for doing this from an AR calibration walk-through to using machine learning to calibrate the system automatically. We would also like to move away from solely using LiDar to determine distance to a subject that we want to bring into focus. We found that especially with the low-end LiDar sensor that was used in this system there was too much inaccuracy at further ranges where the change in focus became very subtle. In order to solve this we think that with a combination of more accurate hardware (Intel RealSense, Ultra-Wide-Band tags/receivers) and machine learning algorithms trained to recognized levels of focus we could more finely tune the focus to achieve higher accuracy and also allow for multiple subjects. We would also like to develop a companion interface such as an iPad app to allow users to have more direct control over the system for when they want it.

References

[1] Shallow vs Deep Depth Of Field In Photography,
https://www.picturecorrect.com/tips/shallow-vs-deep-depth-of-field-in-photography/