

EE/CprE/SE 491 WEEKLY REPORT 03

10/4/24 – 10/10/24

Group number: 11

Project title: Slowpitch Softball Pitch Detector

Client &/Advisor: Nick Fila

Team Members/Role:

Andrew Vick - Machine Learning Integration

Casey Gehling - Client Interaction

Sullivan Fair - Individual Component Development

Ethan Gruening - Team Organization

Josh Hyde - Research

Cameron Mesman - Testing

○ **Weekly Summary**

This week, we developed a much better model that can more accurately determine the height of the ball. Additionally, we enhanced some machine-based AI learning models that scanned thousands of pictures to more accurately identify the softball consistently. Lastly, we started developing a rough sketch idea of what we might want our flutter app to look like.

- **Past week accomplishments**

- **Andrew Vick:**

- This week, I worked on training and testing a non-linear regression model to predict the ball's distance from the camera accurately. In my limited testing, I was able to get accurate readings as long as the model had been trained on data within that region of the screen. On top of this, I could use the ball position on the screen to find how high/low the ball was from the center of the camera's frame. To find the ball height in the real world, we can then add the camera height to the ball's relative height.
 - The ML model to predict the ball's distance will have to be well-trained and properly tested to ensure there is no overfitting to guarantee accurate predictions.
 - The training of the prediction model will need to be done in Python, but the model itself can be exported for use in C++ applications.

- **Sullivan Fair:**

- This week, I attempted to continue to develop my Python script to detect the softball; however, I was unable to make any significant progress in tracking the height of the ball. With that in mind, I decided to shift my focus to the app side of the project as a result of other team members focusing on height detection. During this week we found that a Flutter app would be more suited to our project as we can embed C++ code into it which should improve the overall runtime of our detection. So, I created a default Flutter app, pushed it to our repository, and developed screen sketches for the app before committing to full development.
- TLDR
 - Tried to develop my ball detection script, but shifted focus to the app side
 - Switched our app to use the Flutter framework
 - Created screen sketches to prototype the app

- **Casey Gehling:**

- This week, I experimented with Flutter and worked on building a basic app that interfaces with a camera on an IOS device. Experimenting with different ways of integrating C++ with Flutter, and found various ways to include C++ libraries that integrate with Dart, the language

Flutter is primarily written in.

- Also did some additional research into Flutter itself as well as best design practices for future project hygiene
- Visited a Iowa State recreational softball game on Sunday and collected more video information as well as talked to some players about camera placement strategies.
- Cleaned up the gitlab repository (removed redundant expo project, did some gitignore proofing) and worked on setting up an issue board corresponding to the two separate categories of work we have defined, the Flutter app and OpenCV detection strategies.
 - Note: these categories are fluid, meaning as a team we will most likely be moving between them quite often.
- **Ethan Gruening**
 - This week, I compiled our videos of softball pitches into frames where the softball is visible. 60 frames were captured for each 1 second of the video, creating a set of over 6,000 photos. After the photos were isolated, I used a larger YOLO model trained with the COCO dataset for sports tracking to identify the softball and make its coordinate points. With 6,000 annotated photos of softballs, I then added to the image collection by making variants of the annotations: stretching the images, applying a color filter, and resizing. With over 36,000 images of various lighting, size, and camera quality, the model was ready to be trained with a variety of annotated images. I began training a YOLO machine learning model to be specialized in softball-specific tracking. The 36,000 images were trained and a configuration file was created to implement the detection system within either C++ or Python. I also launched the model on a web service to test the tracking using the system's webcam.
 - Additionally, I organized our groups' empathy map data into a presentation of user needs and how they should affect our design.
- **Josh Hyde**
 - This week I worked on making an alternate object detection code that could potentially be used or used in tandem with the other object detection-based code that we already have and to slightly get a better understanding of it as well. However, this code is slightly inconsistent and only works well at close ranges, leaving the other object detection based code much more reliable moving forward in the future.

- Additionally, I worked on getting flutter, the potential app software we are going to use for our project downloaded. I also tried looking into flutter a little bit and trying to see how it is used to better understand it and get used to potentially having to use it. I also thought of the different potential screens we would have to have for our app to make it working efficiently, and have all of the necessary requirements that it should do, such as a settings page, calibration page of some kind, and some way to look at previous recordings/pitches to be able to rewatch the pitch and see why the system chose to call it illegal or not.
- Cameron Mesman
 - This week, I started designing screens for a potential mobile app. These were pretty rudimentary since we haven't talked much as a group about what we want in our app or even if we're for sure going to make an app. The main idea was to get an initial idea of what screens would likely be on the app and what is the most important information that will be included on each screen. We can make visual tweaks to the screens later on once we have the basic idea of each screen down.
 - I did additional research into photogrammetry to try to find new ways to find the height of an object in an image. This research was unsuccessful for the most part but still provided good discussions amongst the group as we look for the best way to find a height moving forward.
 - I did some additional tuning up on my react skills in anticipation of developing an app in the near future.
- **Pending issues**

We will need to work on converting our detection and tracking logic over to C++, and find a framework to develop an Android and IOS app that can run C++ code on the phone. We are also still running into issue with getting accurate distance data and then using that to find the height of the ball.

○ **Individual contributions**

<u>NAME</u>	<u>Individual Contributions</u> <i>(Quick list of contributions. This should be short.)</i>	<u>Hours this week</u>	<u>HOURS cumulative</u>
Andrew Vick	OpenCV testing, Object detection, photogrammetry research	6	18
Casey Gehling	Flutter research, softball networking, gitlab organization	6	25

Sullivan Fair	Screen sketches, pushed default Flutter app	6	24
Josh Hyde	OpenCV work, Flutter initial work/preparation	6	24
Ethan Gruening	Sketchnotes, PocketRadar research, team contract, team coordination	6	25
Cameron Mesman	Object height/OpenCV research, product research, react refresher	6	12

○ **Plans for the upcoming week**

- Andrew Vick
 - Continue working with OpenCV
 - Test the margin of error when using one camera to calculate depth
 - Port my code from python to C++
- Casey Gehling
 - Finish app camera implementation, start designing app based on designs
 - Experiment with ways to create a minified OpenCV C++ library so that our application size isn't too bloated with unnecessary features
- Ethan Gruening
 - Test the machine learning models on a softball field.
 - Integrate the model into the current GitHub's detection script.
 - Explore if the python script should be rewritten in C++ or create a translator to function within the Flutter application.
- Josh hyde
 - I want to look at potentially non-machine-based learning solutions to the project just to make sure machine based learning is the more reliable solution
 - I want to be able to help the flutter app
 - I want to maybe help with the height detection solution
- Sullivan Fair
 - Review screen sketches with the team
 - Test Andrew's height detection script on a field
 - Research ways to develop the UI of our app
 - Research ways to utilize a camera plugin with Flutter to feed it into our height detection script
- Cameron Mesman
 - Further develop the screens for the app to include more detail and possibly more screens.
 - Continue researching ways we could find the height of an object and help determine if we want to do it using machine learning or not.

