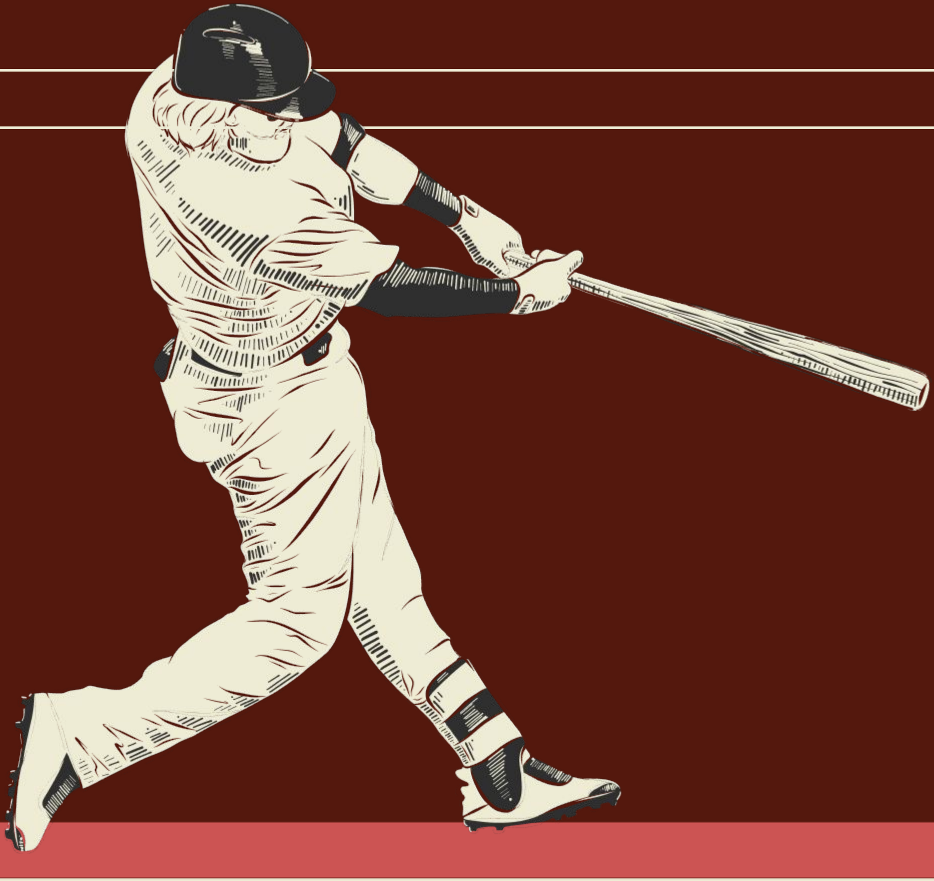


Slow-Pitch Softball Detection

sdmay25-11

Table of Contents

01 **Introduction**

What is our project?

02 **Project Contributors**

Who is this project for? Which faculty will be assisting us?

03 **Functionality**

What are the functional components of our design

04 **Design Model**

What is our current design model?

05 **Technology**

What techniques will we be using?

06 **Developmental Considerations**

What obstacles will we face in the development process?

Our Project

- We are tasked to create a tracker for slow-pitch softball pitches.

- The device will need to track the height of the ball and determine where it falls between a certain height range.
    - Typically 6-10 ft.

- If the ball is not in that range, it will give an audio cue that the pitch was illegal.

- This will help assist umpires call illegal pitches and improve the flow of the game by reducing the number of disagreements on rulings.

Deliverables

- Executable, interactive, tracking software application.
- Our group is making a mobile application.

Client

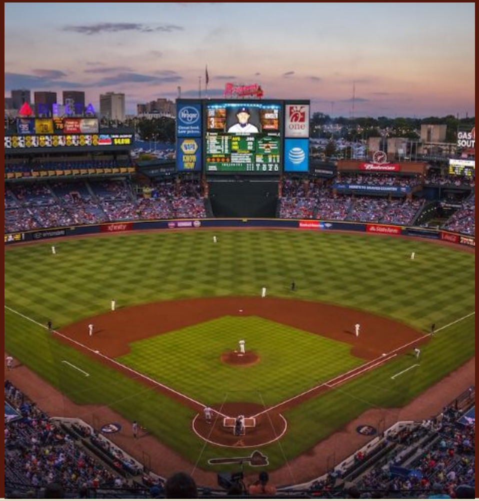
- Dr Nicholas Fila

Faculty Advisors

- Dr Nicholas Fila
- Dr Phillip Jones

Our Design

We plan to implement a mobile application to capture a camera feed and audibly output if an illegal pitch is detected. There are three main factors/segments to our project's design we must consider



The **environment** in which the game is played.

The **object and height detection** given a set of camera grames.

The **user interface** of the mobile application.

Functionality

Environment

- Understand your surroundings and sensors.
- Calibrate lens distortion.
- Calibrate softball color.
- Obtain visual field data (height markers)

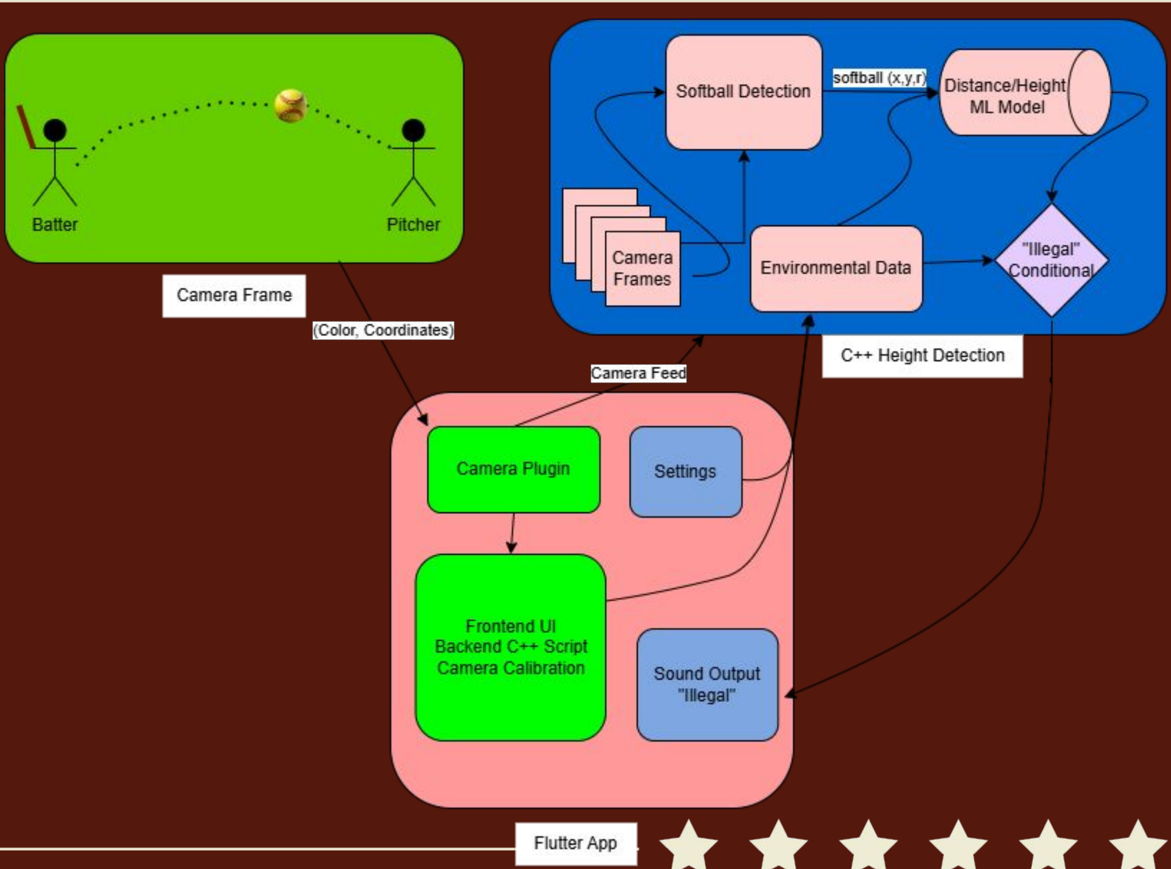
Height Detection

- Input camera feed
- Detect a softball
- Use the environmental variables and softball location to calculate height
- Trigger an “Illegal” condition if needed.

User Interface

- Gain access to the camera
- Run and display the object detection scripts
- Interact with user for calibration steps

Our Design Model



Technologies

Object Detection

- **OpenCV -** OpenCV manages the camera frame-by-frame, allowing easy manipulation, processing, and analysis. It facilitates color detection and identification.
- **OpenCV** is both a Python and C++ library. Our project will use C++ for mobile compatibility.
- **OpenCV** also has tools to remove lens distortion and read in several different color spaces.

User Interface

- **Flutter -** Flutter is an UI toolkit by Google to build natively compiled applications for both iOS and Android from a single codebase. It uses a the language Dart to provide a sets of customizable widgets, for high-performance, visually appealing cross-platform apps.

Development Considerations

Our challenges:

- Object detection using a Machine Learning approach is slow. (5-10fps)
- User experience must be seamless and promote uninterrupted play
- Game environment is not constant (lighting issues, camera placement)

How we will mitigate:

- Exploring other non-ML softball detection strategies
    - Hybrid approaches
- Intuitive UI design, Flutter for cross-platform development
- Refining object detection strategies, camera calibration methods, exhaustive testing

Conclusion

- Practicing user-centered design is critical when developing an application to be used in real time and by many people.
- While designing, relaxing certain constraints in “what if” scenarios may lead to generating better solutions overall.
- Taking into account design requirements while planning for certain features has helped us to work more efficiently and prioritize implementations.

Questions?