

# Preliminary Design Review

Team 1  
Preliminary Design Review  
October 12, 2016



# The Team



Timothy Adams  
CSE



Justin Marple  
CSE



Jason Camiel  
EE



Matt Barnes  
EE



# The Problem

- In baseball, the strike zone is from the batter's knees to halfway up the torso
- Determining strikes and balls, without a professional umpire, is difficult and inaccurate
- Strikes and balls are a core aspect of the game of baseball
- Incorrectly called pitches lead to angry players



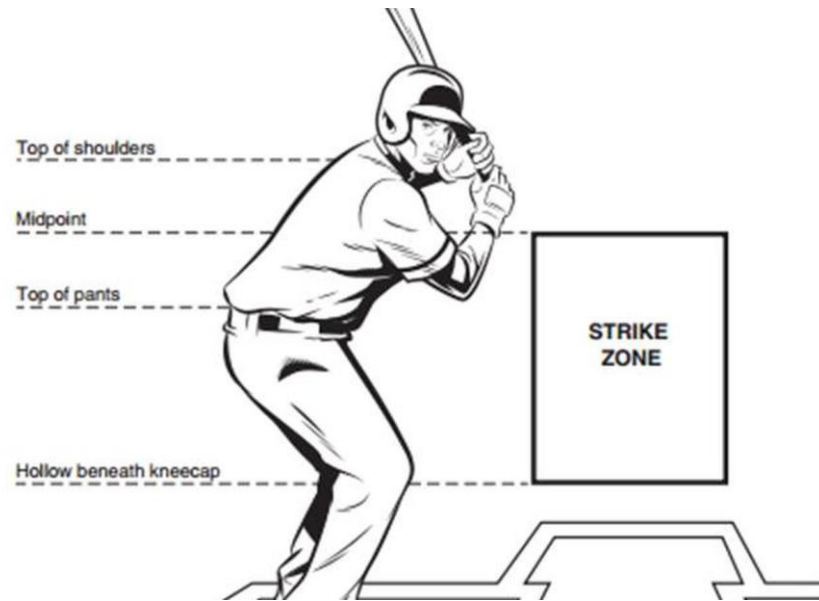
*Photo courtesy of nextlevelballplayer.com*

# Current Solutions

- Expensive
  - Multiple High Speed Camera Systems in Major League stadiums
- Inaccurate
  - Coach “umping” from behind mound
  - Catcher “umping”
  - Chair in Whiffle ball

# Solution

Design a system contained in the home plate that automatically determines balls and strikes by utilizing sensors to determine the baseball's location.



*Photo courtesy of lpdpt.com*

# System Requirements

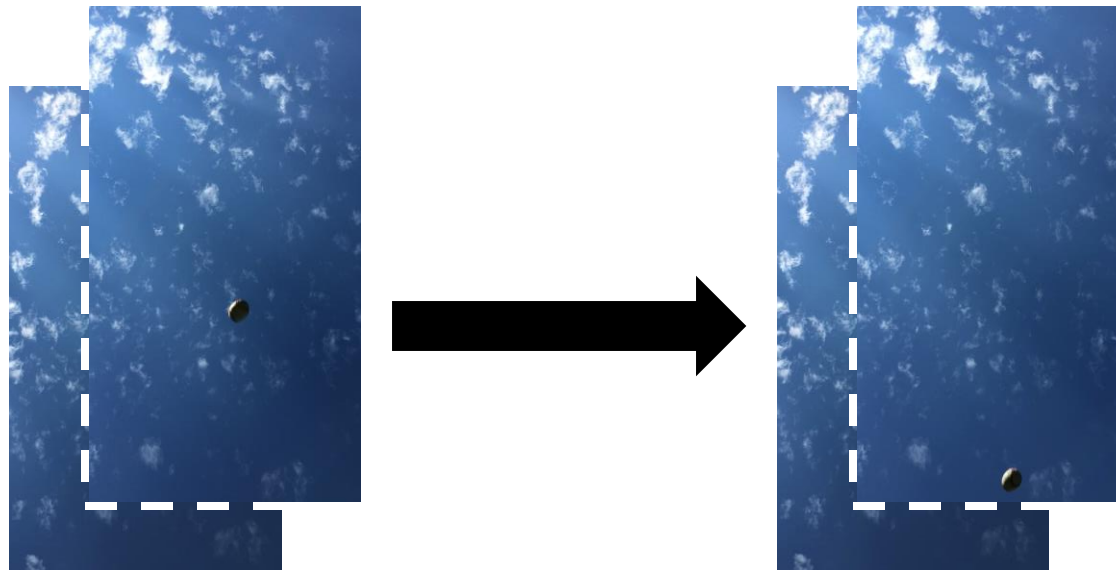
- Determine strikes with 15% error or less when pitch is within one ball length of edge of strike zone
- Accurately detect pitches up to 2 meters high, 1.5 meters wide
- Detect pitches at speeds up to 70 mph
- Battery lasts for 1 game (3 hours)
- Must give strike/ball feedback in real time (within 2-3 sec)
- Self-contained (no external components)
- Perform correctly in regular weather conditions (e.g. cloudy, sunny)
- Physically robust (withstand impacts of normal play)

# Sensors

- Ultrasonic transmitters and transducer
  - Inexpensive
  - High susceptibility to noise
  - Inaccurate location data due to noise
- Radar
  - Expensive
  - High susceptibility to noise
  - High computational requirements
  - Would require many custom parts
- Stereoscopic cameras
  - Inexpensive
  - High data rate

## 2 Camera System

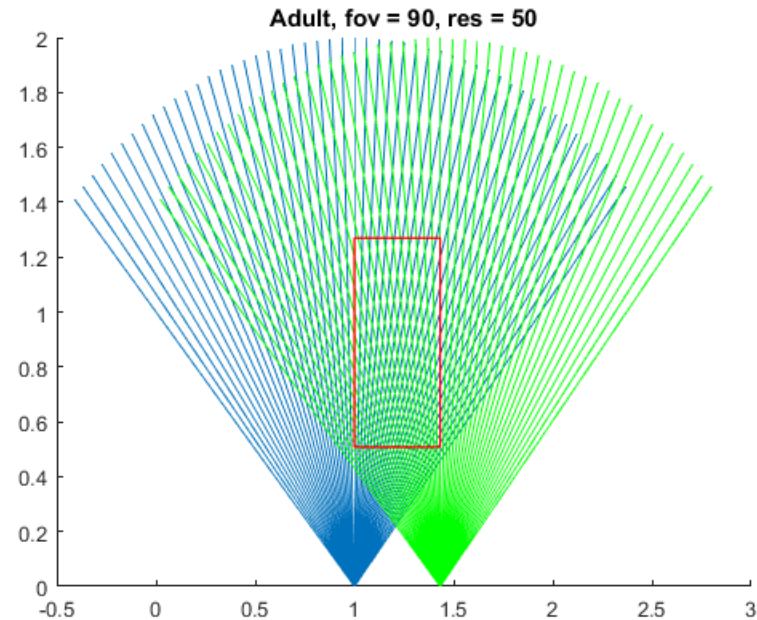
- 2 camera stereoscopic system using image processing to determine 3-dimensional location of ball above plate
- Utilize multiple frames of single pitch to determine path of the baseball through the plane of the strike zone





## 2 Camera System

- Cameras will point directly upwards from front, outer edges of home plate
- High frame rate will ensure at least two frames capture a ball traveling 70 mph at the bottom of the strike zone

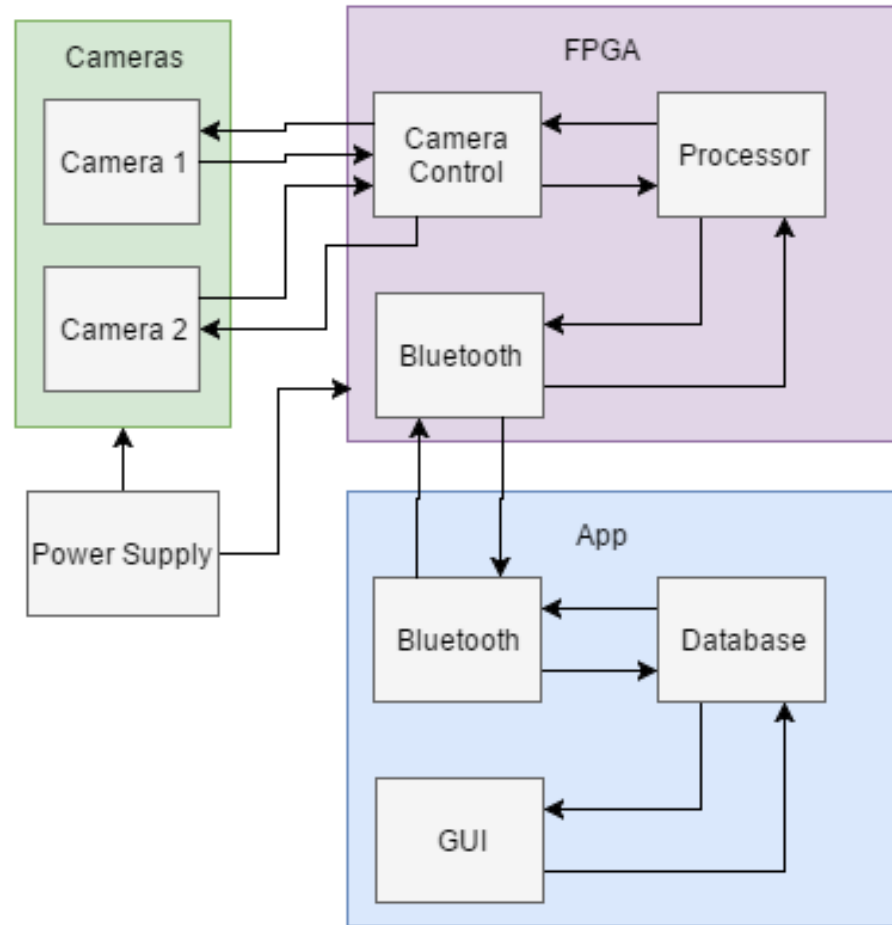


# Challenges

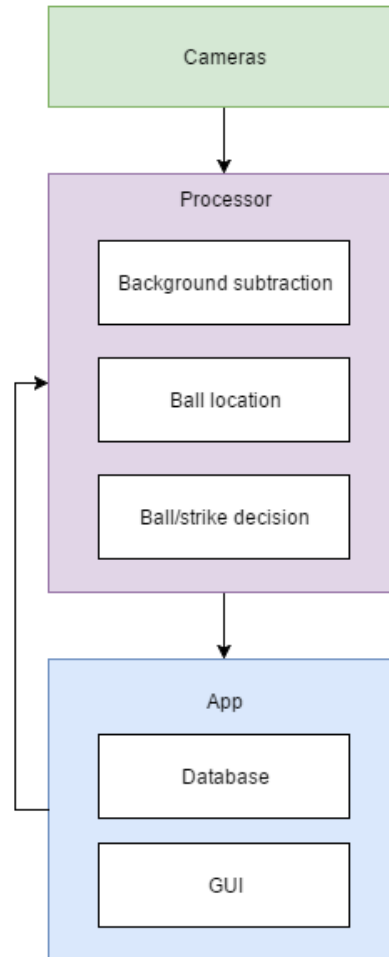
---

- Determine location of ball in 3D space using 2D images
- Locate ball within images given varying backgrounds
- Identify and ignore non-pitches (e.g. a bat, other thrown ball, or a player)
- High computational workload to process many frames per second

# Design: Block Diagram (Physical)



# Design: Block Diagram (Logical)

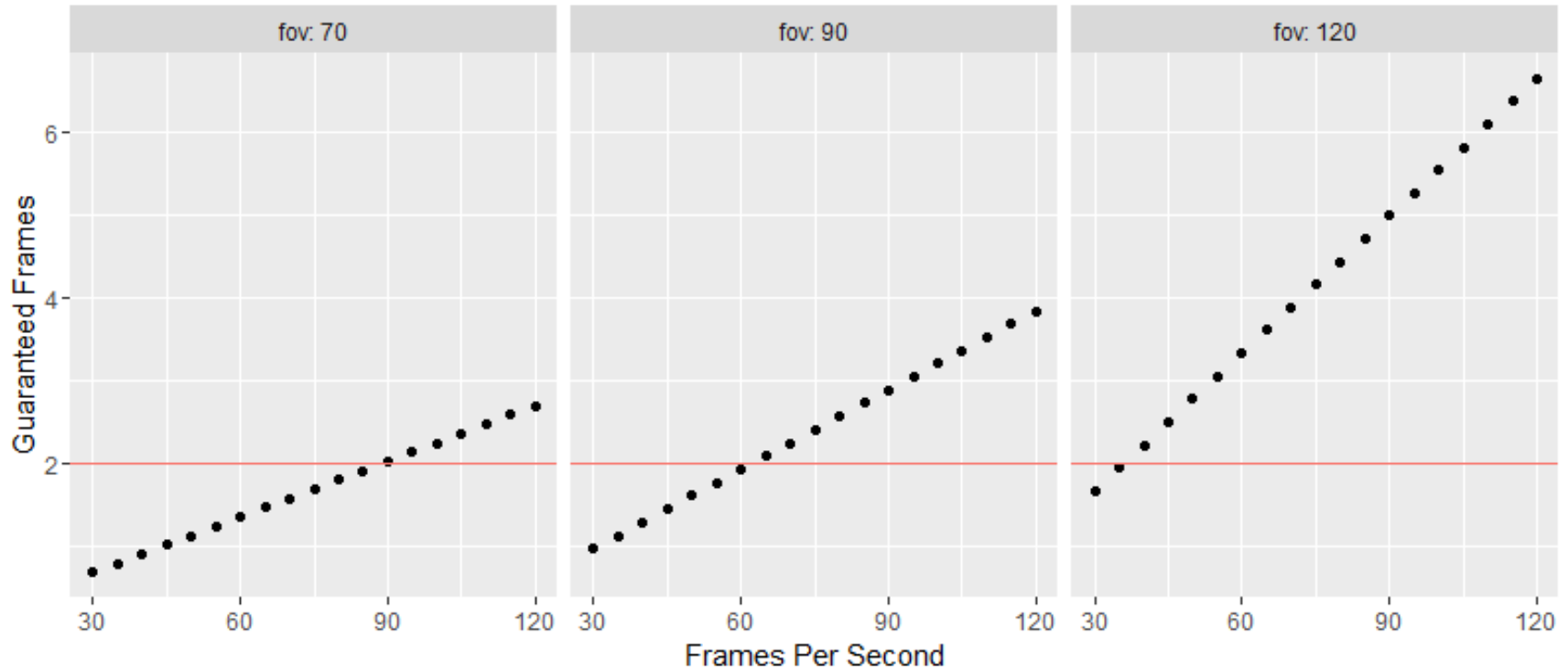


# Design: Hardware selection

- Need cameras that can take pictures at 60fps @ ~480p
- Need a processor who can communicate with the cameras
  - ~2MB of external RAM will be required
  - Algorithms needs to take finish in  $(1 / \text{fps})$  time
    - Ex.  $1 / 60\text{fps} = 16.6\text{ms}$  before next frame.

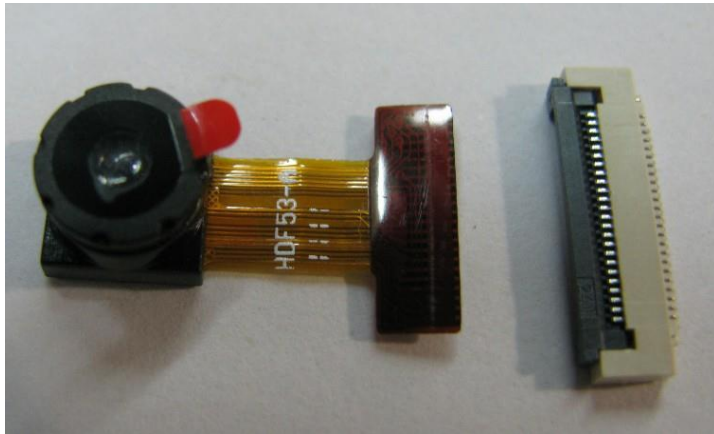
# Design: Camera selection

Frames Seen vs. Frames Per Second for Different Fields of View



# Design: Camera selection

- Camera
  - Omni-Vision has a large selection of cameras
  - Up to a couple MP
  - Many can go up to 120fps



# Design: Processor selection

- Processors:
  - Regular CPU (single-core uProc)
    - Easy to program and test.
    - Easy to add RAM
    - Very hard to implement stereo-cameras
  - XMOS-based CPU (multi-core uProc)
    - Easy to program and test.
    - Reasonable to implement stereo-cameras
    - Difficult to add RAM
  - FPGA
    - Parallel execution is a big plus
    - Reasonable to implement stereo-cameras
    - PCB/Software might be harder to implement.



# Estimated Costs

<b>Unit</b>	Quantity	Unit Cost	<b>Cost</b>
<b>FPGA</b>	1	\$10	<b>\$10</b>
<b>Cameras</b>	2	\$30	<b>\$60</b>
<b>Power Supply</b>	1	\$10	<b>\$10</b>
<b>Bluetooth</b>	1	\$10	<b>\$10</b>
<b>Total Cost</b>			<b>\$90</b>

# Distribution of responsibilities

- Jason: Power supply/Bluetooth
  - Establish stable power supply to all systems
  - Connect and configure Bluetooth module for processor
- Matt: Camera control/synchronization
  - Identify and implement hardware protocol
  - Ensure video feeds are linked frame by frame
- Justin: Low level processing, TX/RX
  - Implement image processing to identify frames that contain ball
  - Handle strict time requirements to match frame rate
- Tim: High level processing, app (GUI/database)
  - Implement image processing to locate ball in 3D space
  - Create mobile app that displays results of pitch analysis to user

## Proposed MDR Deliverables

---

- Demonstration of image collection
  - Create two synchronized video feeds using cameras and collect with processor
- Demonstration of core image processing algorithm in MATLAB
  - Returns  $x, y$  position of ball above plate given two .MOV files