

COMP3106 Final Project: **FieldVision AI**

Mario Pardo - 101286588

Paul - 101219909

Fabrice - 101196480

Introduction

Inspired by the movie “Moneyball”, we realized that baseball is the perfect sport for which we could create an AI-based tool to help coaches. It’s a sport with lots of metrics and data which we can use to train our models, and the game is static enough such that we can make real-time predictions with our limited model sizes and restrictions.

We believe that we can make a tool which can help baseball enthusiasts (in our case MLB coaches), set the best possible outfielder positions given the ‘current state of the game’.

This is the **general idea**

- 1) The system receives an overhead view (such as an eagle eye) of the game.
From this, the system determines which bases are loaded
- 2) The coach tells the system the current status of the game - score, inning, who’s pitching and who’s batting
- 3) Our model will be trained on large amounts of MLB situational data, so it can predict a field of probabilities on where the batter will hit the ball. Again, given the batter’s history and what they’ve done in similar situations to these in the past.
- 4) The system outputs the grid of probabilities, which the coach can use to set his outfielder’s positions.

Novelty: We believe we are creating something that isn’t publicly available, using novel AI technologies in a sports world that is more and more data-driven. Success in a project like this can have real-world benefits and implications in the world of baseball and sports in general.

Objectives: The main objectives we want our system to accomplish

- Use an API to retrieve the data we want, and then create code to help us sort/parse this data effectively
- Use **computer vision** to understand which bases are loaded from a field view
- Train a model with the mentioned MLB data to predict what a batter is likely to do given the game’s current state

Validation Strategy

Computer Vision: this can be easily tested by manually classifying in-game photos (which bases loaded) and comparing them to the accuracy of the predictions

Model: We will train a model on many MLB games to get a model that can analyze the current game state, given who is pitching and batting, and predict what the batter will do. To test this, we will have to find a subset of this data (games) which we do not train the model with, and use this to test our model.

Dataset

We will be using the the pybaseball library and API to get our data

To train the model, for each batting event, we will be getting

Current game score, bases loaded

Pitcher: Pitching stats including what they like to throw, speed, location, etc

Batter: Stats and preferences

Batting event: Type of bat (punt, ground ball,etc), location of ball landed

The specific data used is subject to change due to availability of data, and how relevant it is due to our model - this will be calculated as we go

Weekly Schedule

Dates	Milestones	Tasks
Oct 27 - Nov 2	Model selection and data gathering	<ul style="list-style-type: none">• Select suitable MLB dataset for model training• Select best model given dataset
Nov 3 - Nov 9	Model training	<ul style="list-style-type: none">• Implement classification and validation strategy
Nov 10 - Nov 16	Image recognition implementation	<ul style="list-style-type: none">• Gather images• Train a machine vision model
Nov 17 - Nov 23	Model validation	<ul style="list-style-type: none">• Test model accuracies
Nov 24 - Nov 30	Environment demo setup	<ul style="list-style-type: none">• Make necessary UIs• Code for running application as a whole
Dec 1 - Dec 6	Final report submission	

GPU Usage

Yes

Availability for Demo

Monday, Dec 2nd - 2:30 pm to 3:30 pm

Tuesday, Dec 2nd - 2:30 pm to 3:30 pm

Friday, Dec 6th - 11 am to 12 pm, 12 pm to 1 pm, Anytime before 3:30 pm