Finding NFL Helmet Impacts with Object Detection

by David Bartholomew

The Need for Object Detection

- From 2012-2019, overall concussions have not changed significantly.
- There is still a need to effectively reduce the amount of concussions per year.

Incidence of Concussion – 2012-2020

Year	Preseason			Regular Season			Preseason + Regular Season		
	Practice	Game	Total	Practice	Game	Total	Practice	Game	Total
2012	42	43	**85**	3	173	**176**	45	216	**261**
2013	39	38	**77**	4	148	**152**	43	186	**229**
2014	42	41	**83**	8	115	**123**	50	156	**206**
2015	29	54	**83**	9	183	**192**	38	237	**275**
2016	26	45	**71**	6	166	**172**	32	211	**243**
2017	45	46	**91**	12	178	**190**	57	224	**281**
2018	45	34	**79**	8	127	**135**	53	161	**214**
2019	30	49	**79**	9	136	**145**	39	185	**224**
2020	30	N/A*	**30***						

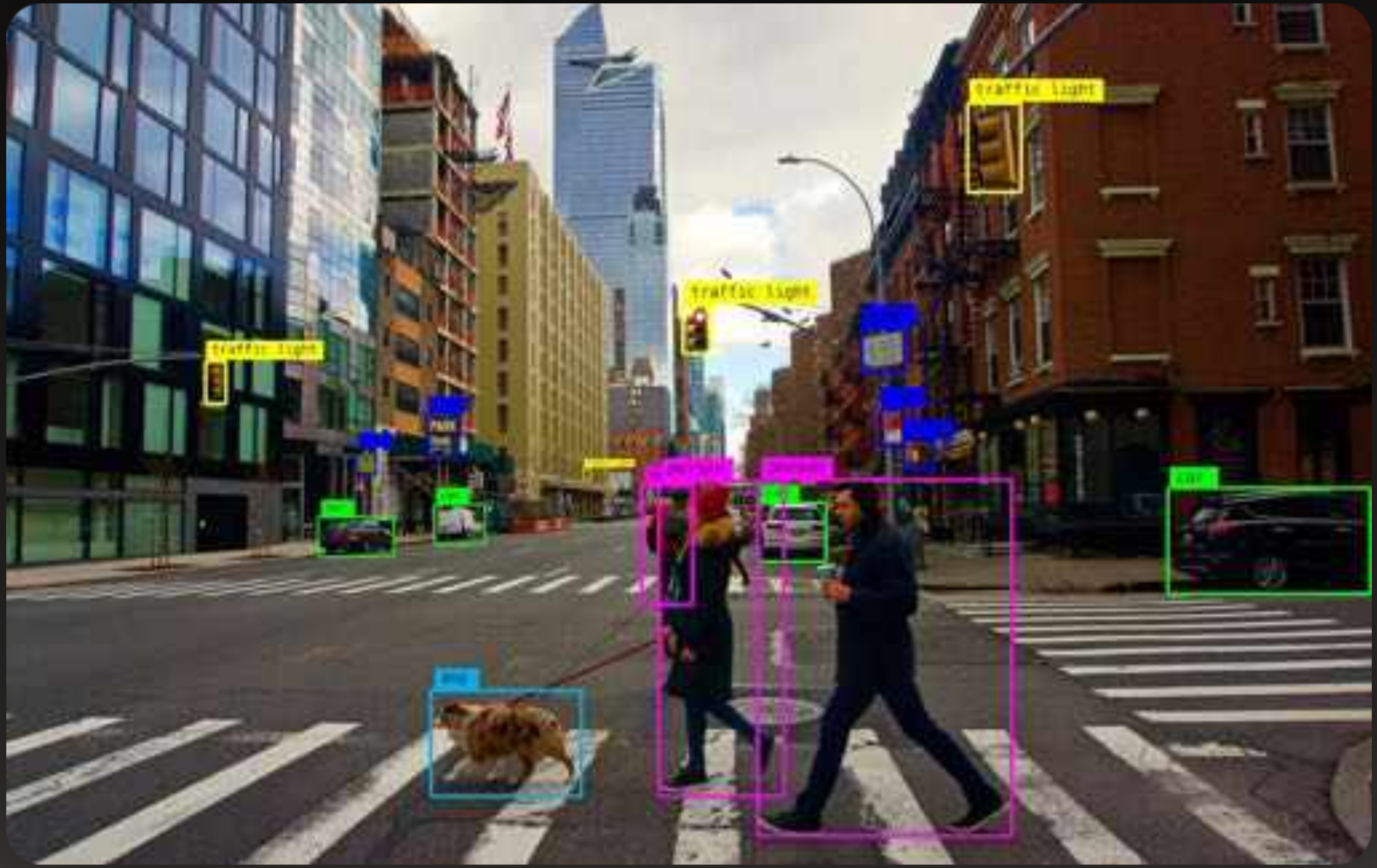
Injuries reported during conditioning, weightlifting, run sessions, or individual training, as well as chronic injuries or those with insidious or unknown onset are not included.

*2020 preseason games were canceled due to the impacts of COVID-19.

Image owned by NFL and available here.

Object Detection Overview

- Localization: identify where objects are located in an image
- Classification: identify which objects are in those locations



Benefits of Object Detection

Increasing Player Awareness

The NFL created the Use of the Helmet rule, which results in a 15 yard penalty for players who lower their heads to initiate contact. Some impacts go unseen by referees. Detecting impacts could potentially flag these scenarios, increasing player awareness.

Quantifying Impacts

Detecting impacts can help quantify impacts for manufacturers and insurance companies, helping answer questions like, “How much damage can a helmet sustain?”

Further Data Analysis

Detecting where impacts occur can provide data for further analysis of additional rule changes. It can also potentially help manufacturers understand how to better pad helmets for specific positions and for different players.

The Data

- ✓ 120 labeled videos with end zone and sideline views (matching plays for each)
- ✓ Over 10k corresponding images to the video data
- ✓ A CSV file with labels of bounding boxes noting the locations of helmets in video frames as well as pertinent impact data, and a corresponding CSV file with bounding box labels for the image data
- ✓ A CSV file for player tracking data, noting the x, y coordinates of player location over time, their speed, acceleration, direction, orientation, and other related data (tracking sensors are in each helmet).

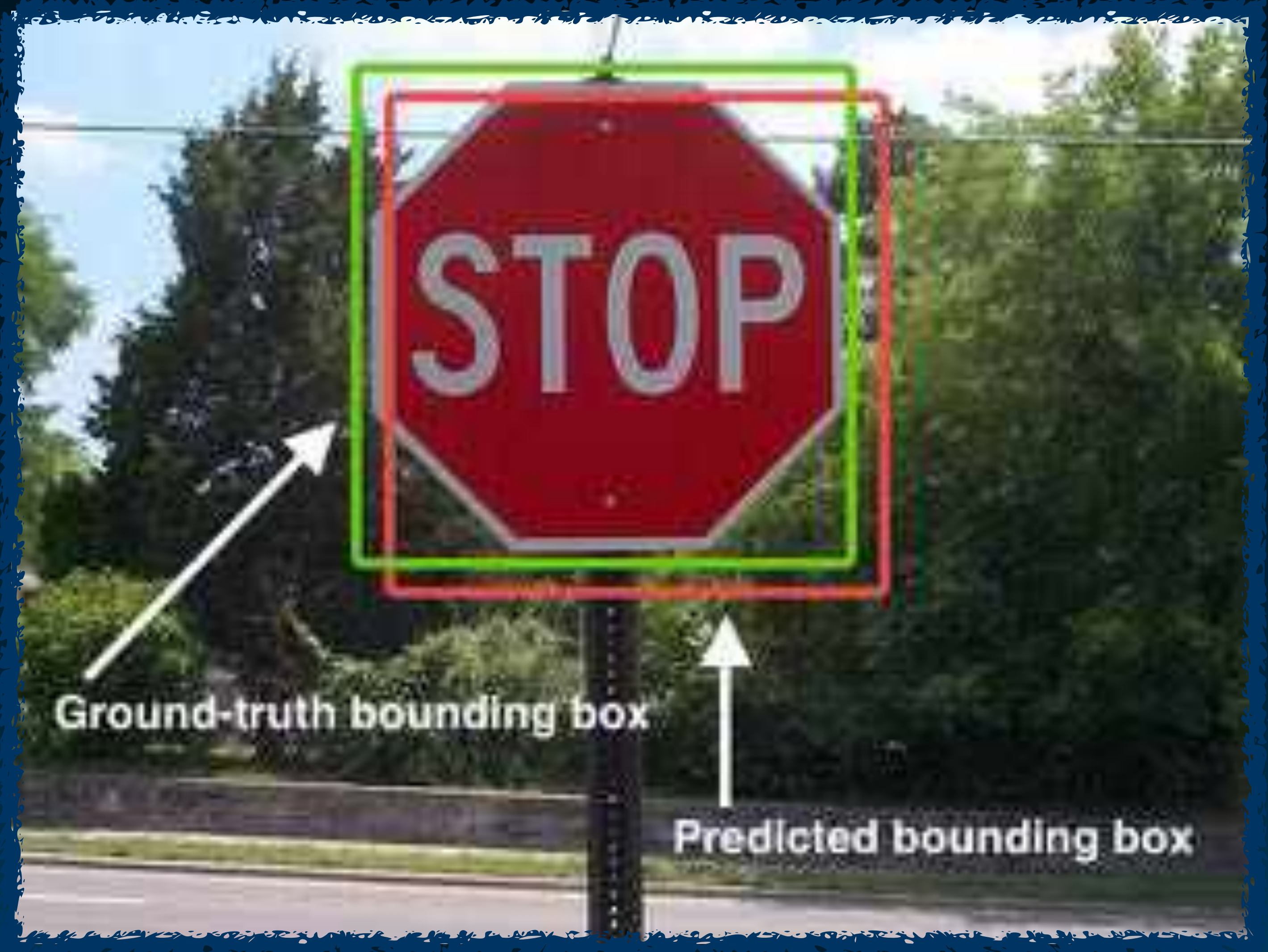
left	top	right	bottom	width	height
647.0	37.0	691.0	85.0	44.0	48.0
649.0	349.0	668.0	376.0	19.0	27.0
463.0	363.0	506.0	402.0	43.0	39.0
658.0	329.0	701.0	376.0	43.0	47.0
483.0	324.0	530.0	370.0	47.0	46.0
70.0	52.0	105.0	101.0	35.0	49.0
744.0	348.0	782.0	388.0	38.0	40.0
435.0	245.0	479.0	291.0	44.0	46.0
60.0	421.0	103.0	462.0	43.0	41.0
384.0	77.0	429.0	120.0	45.0	43.0
573.0	317.0	619.0	361.0	46.0	44.0
986.0	424.0	1030.0	465.0	44.0	41.0
56.0	608.0	103.0	657.0	47.0	49.0
374.0	155.0	419.0	200.0	45.0	45.0
197.0	233.0	246.0	279.0	49.0	46.0

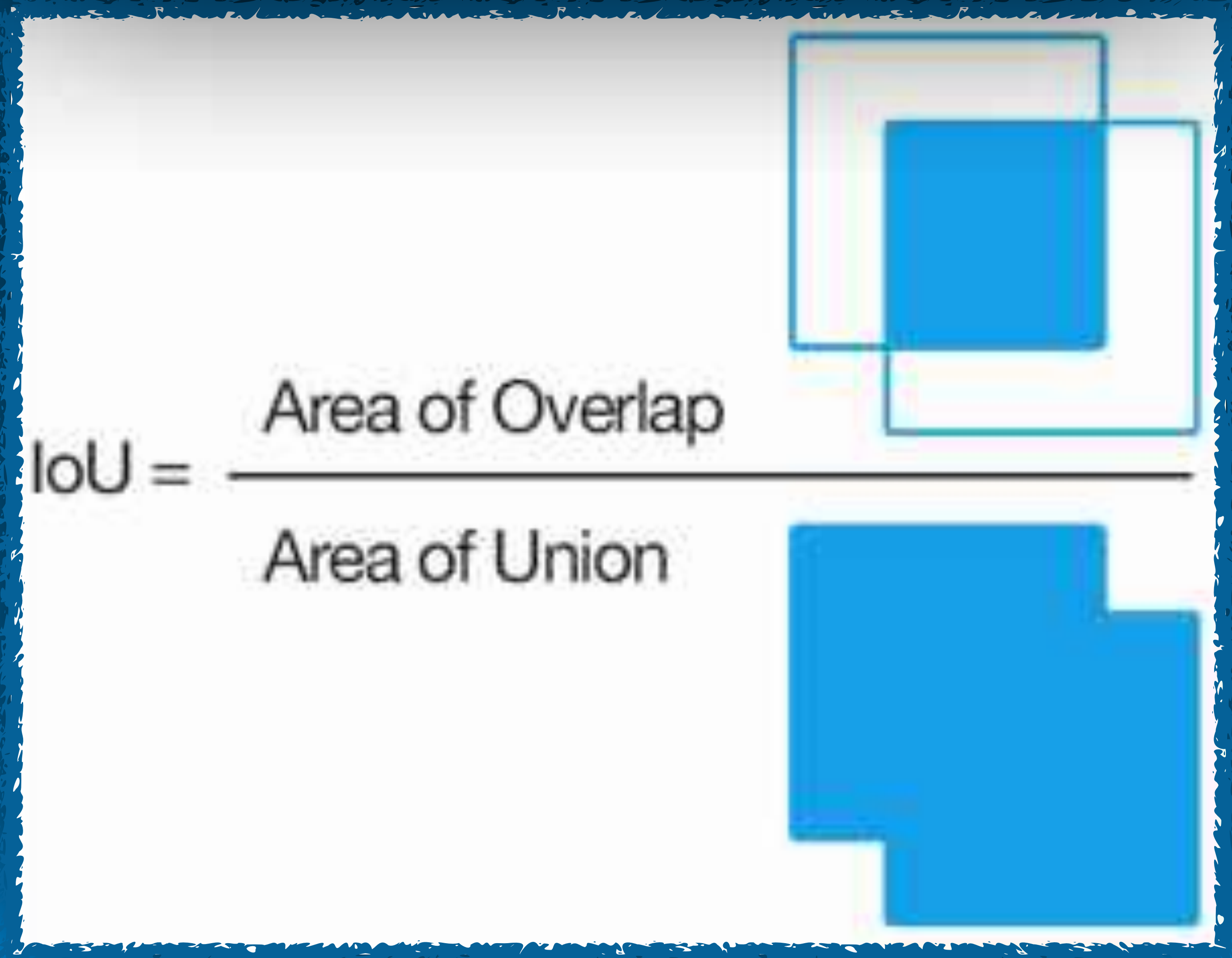
The Data (continued)

- Rather than using the 10k images, the images used with this project were captured from specific video frames for both end zone and sideline views, sorted by impacts to ensure most images included impacts.
- A DataFrame was built with unique image ids. 1,081 images in total were used.
- The images were split into a training set and testing set along with their corresponding labeled data in the DataFrame to evaluate the final model. The total objects/helmets in the 1,081 images used was 21,843.
- The player tracking information was not used for this project, but there is opportunity to implement in future work.

The Model: Key Parameters

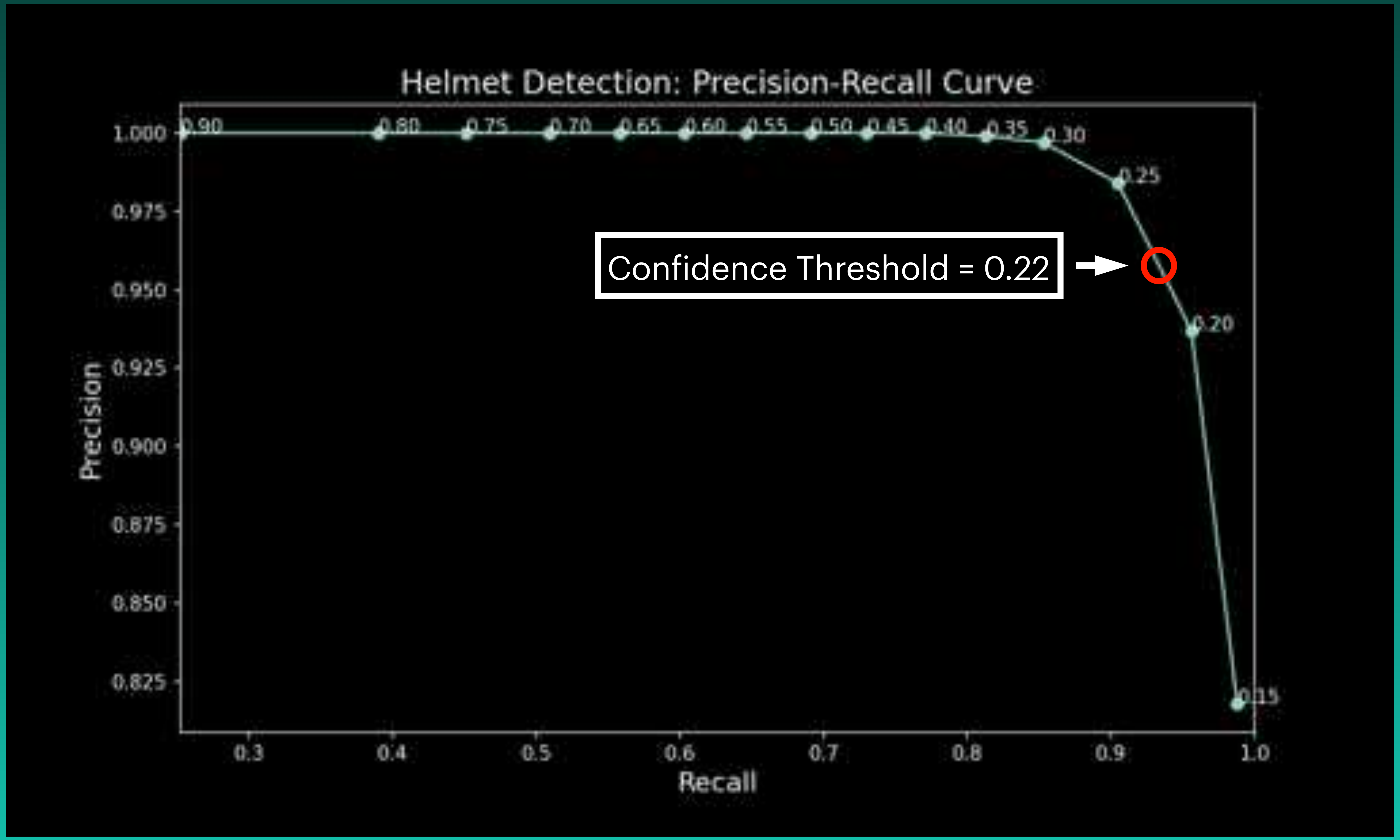
- RetinaNet: a state-of-the-art object detection model. Performs both localization and classification.
- Anchor Boxes: fixed sized boxes that the model uses to predict bounding boxes. The model generates thousands of anchor boxes.
- Intersection over Union (IoU): calculation of the overlap of anchor boxes and ground truth boxes. Area where the anchor box and ground truth box intersect divided by the total combined area of both boxes.
- An IoU threshold is specified to determine how the model learns which anchor boxes are closest to the ground truth boxes.
- Confidence score: the confidence score is a value between 0 and 1 that is equivalent to the predicted class probability. The model is built to select anchor boxes with the highest confidence score.

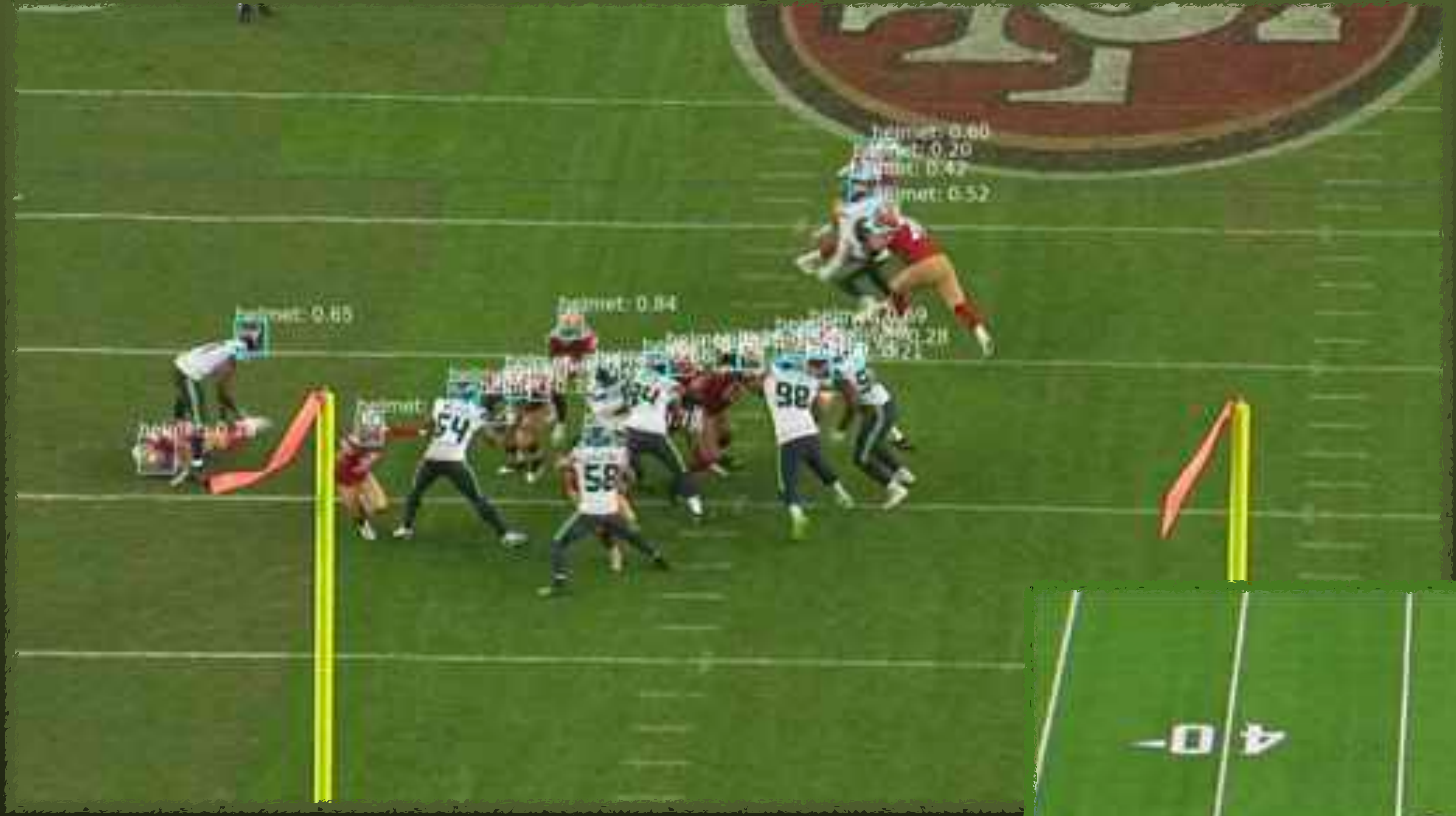
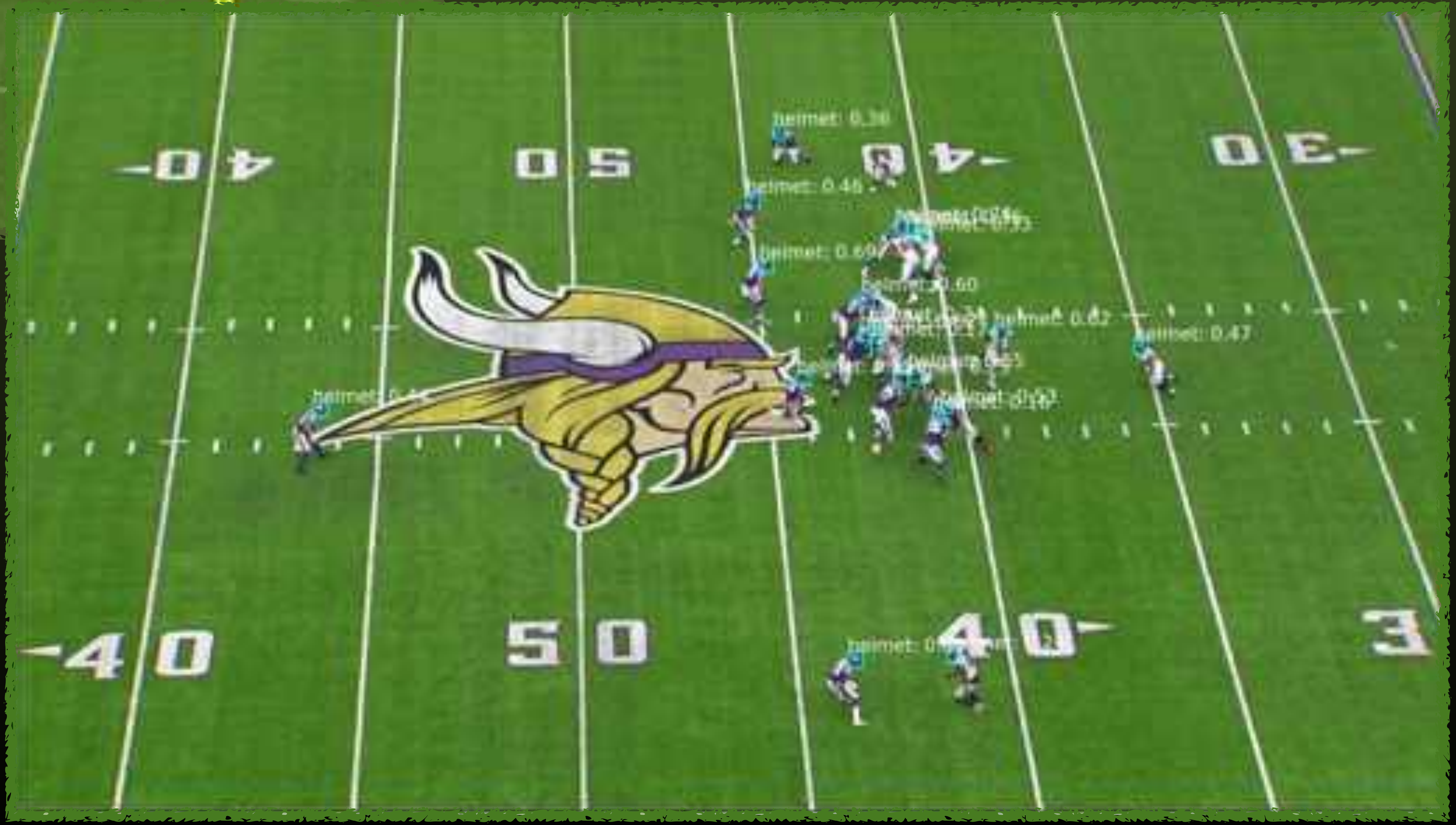


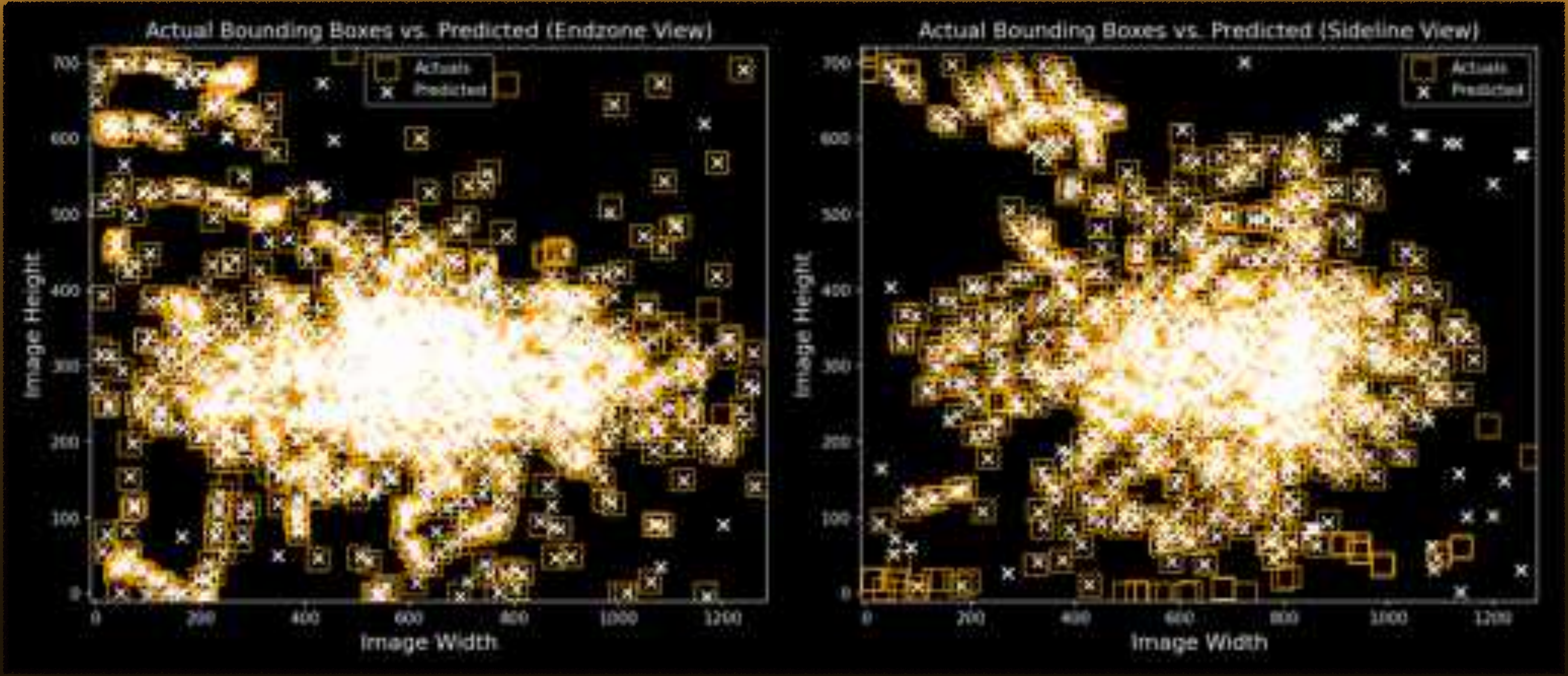


Evaluating Results

- Accuracy is impossible to measure since true negatives can't be quantified (background is considered true negative)
- Compared the results of different confidence thresholds ranging from 0.15 to 0.90
- Classified results above the confidence threshold as true positives and results below the threshold as true negatives
- If the total count of predicted objects was less than actuals, the difference was labeled as false negative/s. If the count was higher, the difference was labeled as false positive/s
- The precision/recall curve highlights the trade-off between eliminating false positives (precision = 1.0) and eliminating false negatives (recall = 1.0) based on the confidence thresholds.
- F1 Score is the harmonic balance between precision and recall. The confidence threshold of 0.22 had the highest F1 score of 0.95.

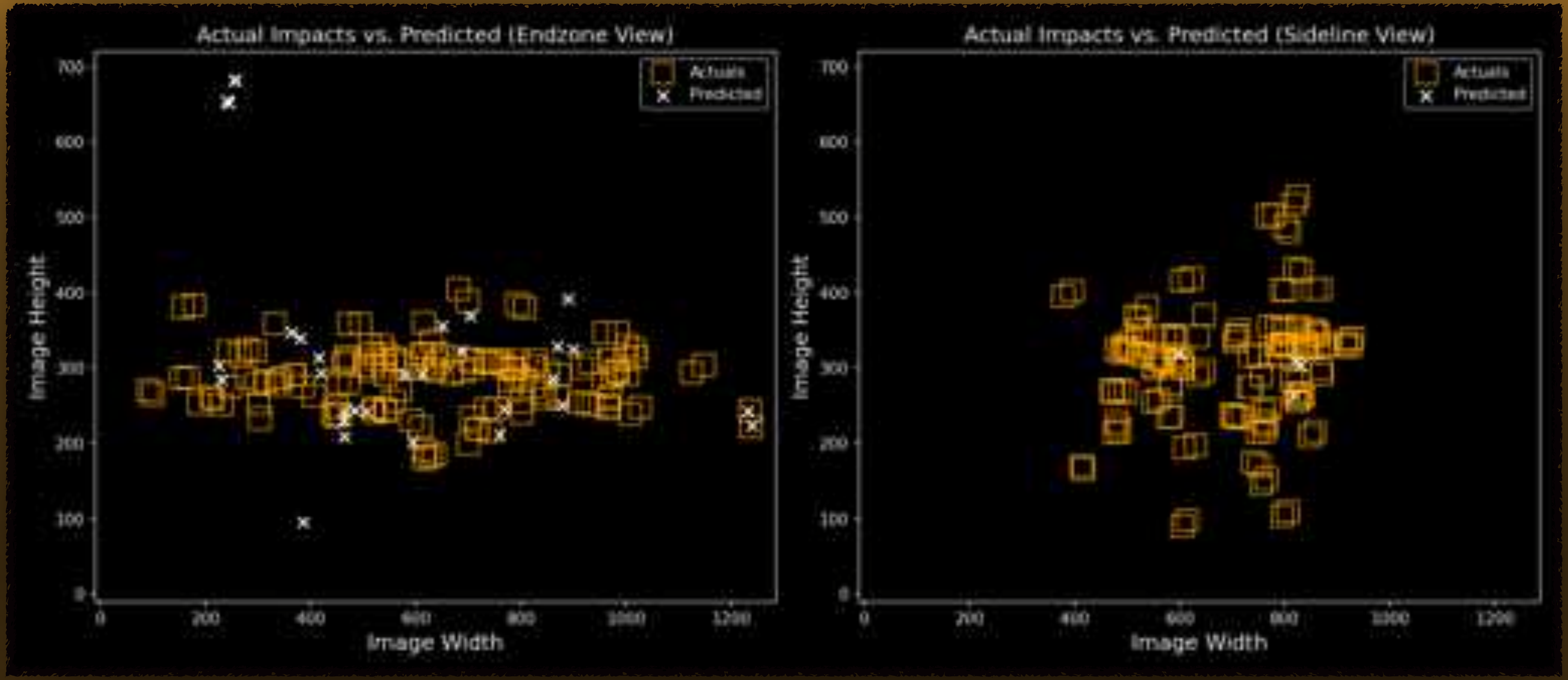



helmet: 0.60
helmet: 0.52
helmet: 0.65
helmet: 0.84

helmet: 0.36
helmet: 0.46
helmet: 0.69
helmet: 0.60
helmet: 0.62
helmet: 0.47
40
50
40
3


Actual Bounding Boxes vs. Predicted (Endzone View)
Actuals
Predicted
Image Height
Image Width
0
100
200
300
400
500
600
700
0
200
400
600
800
1000
1200
Actual Bounding Boxes vs. Predicted (Sideline View)
Actuals
Predicted
Image Height
Image Width
0
100
200
300
400
500
600
700
0
200
400
600
800
1000
1200

Detecting Impacts

- Impacts were defined using a strict IoU filter of 0.1 to 0.4, but this method had its faults.
- It is necessary to consider other dimensions, both the temporal dimension (shift between frames in videos) and depth (from 2D to 3D).



Detecting Helmets/Impacts in Video

Conclusion/Further Work

- Using object detection for finding helmet impacts could potentially assist with increasing player and referee awareness of impacts and additional penalties for violating use of the helmet rule, quantify impacts for insurance and manufacturing companies, and would allow further data analysis for additional rule changes and how helmets are manufactured based on where impacts are occurring.
- The final model did reasonably well detecting helmets with an F1 score of .957 for the end zone video and .96 for the sideline video.
- The final model did not do well detecting impacts due to only considering 2D aspects.
- Recommendations for further work include but are not limited to the following:
  - Combining previous data with player tracking data and potentially using homography to locate players in videos based on tracking key points.
  - Potentially using 3D Convolutional Neural Network models to identify impacts. While this is computationally expensive, it's possible to include crops of areas where impacts occur and attempt to convert these areas from 2D to 3D, giving an added dimension of depth.
  - Using a Temporal Shift Module. Videos have an additional dimension over images, the temporal dimension, which is the shift of objects in images between frames. Using this information, the TSM predicts actions based on a specified range of frames. This is a less computationally-expensive model as it only utilizes 2D data.

Project Link

- Full details of this project are available at this link:

https://github.com/dbarth411/NLF-Object-Detection

References

- https://www.nfl.com/playerhealthandsafety/health-and-wellness/injury-data/injury-data
- https://keras.io/examples/vision/retinanet/
- https://blog.zenggyu.com/en/post/2018-12-05/retinanet-explained-and-demystified/#fn10

Thank you!