Detecting tennis players using the yolo model

Objectives:

The goal of this project was to use the yolo model to detect the tennis players and tennis in a video of a tennis point.

A introduction of the yolo model:

Yolo is a machine learning object detection model. It generates bounding boxes and calculates the probability of each one being a particular object. The model is trained by repeatedly making predictions and adjusting the model based on how accurate the predictions are. Eventually, the model can make a very accurate prediction of the location of each object in an image.

Project overview:

1. Each frame of the video was labeled with the boxes of each player.
2. The yolo model was trained to detect the 2 players using the labeled images.
3. Test the trained model on the rest of the video to see how well it can detect the players.

Project details:

I labeled the first 400 frames of the video by drawing boxes around each player and the tennis ball in the frame. Then I trained the yolo model to detect the 2 players and the tennis ball using the prewritten training code. The model learns by repeatedly making predictions about where the objects are and adjusting based on the accuracy of those predictions.

In the beginning, the model did not successfully detect the ball and players. I tweaked the different parameters of the model such as the batch size, learning rate, and number of epochs. The batch size determines how many groups the images are split into every epoch. The learning rate determines how quickly the model changes each iteration.

I also tried ignoring the tennis ball and trained the model to only detect the players. This was because the size of the tennis ball was too small for the yolo model to detect. When the boxes for the tennis balls were removed from the annotation file, the performance of the model improved drastically and the model started detecting the players.

This is an example of the model's prediction boxes for the two players.

The loss and precision are used to measure the accuracy of a trained model. The loss curve for the model shows how the training loss always decreases and the validation initially decreases then increases. The training loss is the value of the loss function for the training set,

and the validation loss is the loss value for the testing set. The validation loss starts to go up at the end because the model has started overfitting the training data.

This is the loss curve for the final trained model.
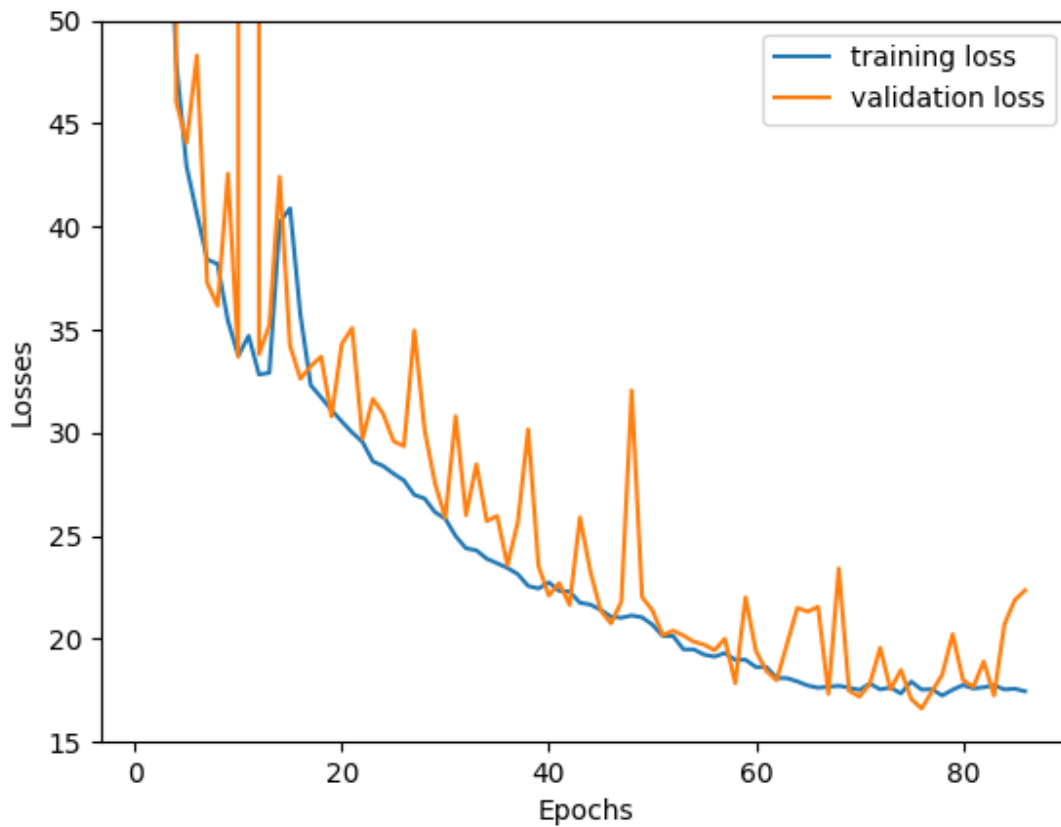
The trained model was able to successfully detect tennis players in a video of a tennis match. The mean average precision of the model was 75.14%. This shows that the accuracy of the trained model is high, and can detect most of the players in the video.

mAP = 75.14%
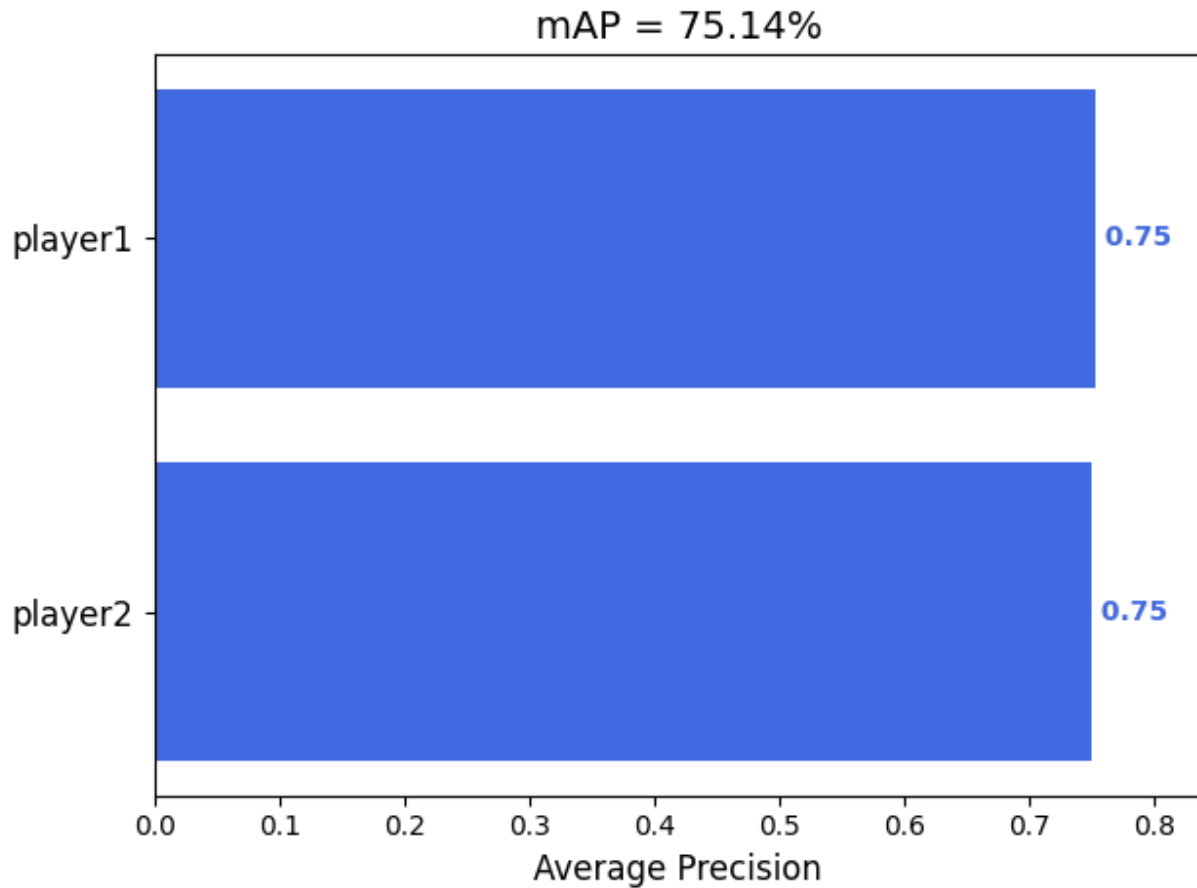
Conclusion:

   Through this project, I've learned about how the yolo model, and image detection algorithms work. I learned about the specific parameters used in the yolo model such as the learning rate and batch size. I also learned how to debug and improve the performance of the model by changing some parameters, and by removing some of the objects. Finally, I have also improved my skills in working with other people.