

Daily Log

Tuesday October 15

I realized I forgot to add some methods that I needed such as the non-max suppression method. I had to spend a decent amount of time researching how it worked and how to implement it. I also realized that my IOU calculation method (Intersection over Union) method was a bit off which was why I was getting strange results the first time I tried to test the method.

Thursday October 17

I spent time cleaning up the methods and researching some more and looking at the original YOLO paper. I also tried looking at other people's implementations of YOLO to get a better idea of what was wrong with my code.

Monday October 21

Same as Thursday, mostly small edits to my code and some tests, didn't really have a ton of time to test on real images yet.

Tuesday October 22

I realized that the YOLO classifier layer in my ResNet50 network was a bit off, but since I already have everything else put together, it wasn't hard to retrain.

Thursday October 24

Spent time trying to figure out how to add the bounding boxes into each individual frame of a video. Starting with images first.

Timeline

Date	Goal	Met
9/30	reach acceptable accuracy (over 95 percent) on data with a validation set and augmentations	no - not yet (80-85 percent)
10/07	develop object detection framework	yes
10/15	develop a running object detection framework	yes
10/22	develop a running object detection framework able to detect logos in real-time	no - not yet
10/28	develop a running object detection framework able to detect logos in real-time at least 50 percent accuracy	no - not yet
11/4	develop a running object detection framework able to detect logos in real-time at least 80 percent accuracy	no - not yet

Reflection

I think I've accomplished a lot in the past two weeks. Looking back at my original timeline, I think I have a good bit of flexibility going forward because I am a good bit ahead of schedule. Thankfully, I've have a good amount of time to flesh out some small problems with my network and really make sure it's running well. I'm anticipating this is going to require a really good amount of fine-tuning, given that my dataset is fairly small and in order for the program to be effective for commercial use, all visible/recognizable logos have to be blurred.

#YOLO CLASSIFIER

```
x = Flatten()(x)
```

```
x = Dense(1024, activation='relu')(x)
```

```
x = Dropout(0.3)(x)
```

#OUTPUT TENSOR CALCS

```
x = Dense(13*13*(27+2*5), activation='linear')(x)
```