

# CSC486B/CSC586B Final Project Feedback

Kwang Moo Yi  
University of Victoria  
Victoria, BC, Canada  
kyi@uvic.ca

## Abstract

*The feedback is written in two parts. I will first discuss about the report, and then the code.*

### 1. Thanks!

First of all, thanks for surviving the semester! As learning continues even after the course ends, here's a final feedback on your final project.

### 2. Report (95/100)

Overall, paragraphs are a bit long, with multiple subjects discussed in a single paragraph.

#### 2.1. Introduction

Up to point and well written. For the sake of feedback, I think you can split the paragraph into two, and maybe add another paragraph summarizing your findings as well, although I did not ask for it.

#### 2.2. Anchor Boxes

You can split the first paragraph into multiple paragraphs. As noted above, ideally you want to discuss one subject in each paragraph. Also, are you sure it should be the lowest IoU? I'm a bit confused there.

#### 2.3. Learning rate and Adam

ADAM does not mean the learning rate decays. To do this you would need to use a learning rate schedule. Also the learning rate that was used is way too small. Default parameters typically work well. And with batch normalization, it's often good to use high learning rates.

### 3. Code (85/100)

Awesome job! The code base is quite clean and easy to follow. Also brave of you guys to go for TFRecords. Here are some tips that might help you in the future.

### 3.1. Debugging deep networks

**Making it manageable.** To debug deep networks, you want to use a small subset of your dataset and first overfit to it. For example, you can take only two classes from the VOC dataset, and then train YOLO to detect only those. Moreover, at first, you want to train/validate/test on the same set, so that you are sure your network is behaving as intended. For example, using a single batch for everything is actually a good idea.

**One step at a time.** You also want to validate one thing at a time. For example, the loss function of YOLO can be tested one at a time. Although this seems time consuming, it's a good idea to do this in a single batch setup. For example, you first verify that the classification is working, then go to the grid localization, then then anchor, then the anchor adjustments. Baby-steps.

### 3.2. Data augmentation

I see that you guys also did some data augmentation by randomly flipping. A good idea is to also randomly apply small affine transformations. Also, you are resizing the image with interpolation, but you might want to crop first to keep the aspect ratio. Sounds unimportant, but could make a big difference!