Homework 08 – Vision

Arthur J. Redfern
arthur.redfern@utdallas.edu

---

0 Outline

1 Reading

1. Vision
   Motivation: understand common uses of xNNs in vision applications
   https://github.com/arthurredfern/UT-Dallas-CS-6301-CNNs/blob/master/Lectures/xNNs_080_Vision.pdf

   Complete

2. [Optional] Faster R-CNN: towards real-time object detection with region proposal networks
   Motivation: key 2 stage object detection paper that cleanly puts together a number of ideas to create a structure that forms the basis for many modern object detectors
   https://arxiv.org/abs/1506.01497

   Complete

3. [Optional] Mask R-CNN
   Motivation: a similar idea applied to object segmentation
   https://arxiv.org/abs/1703.06870

   Complete

4. [Optional] Feature pyramid networks for object detection
   Motivation: key method for cleanly handling different scales of objects applicable to many different object detection and object segmentation methods
   https://arxiv.org/abs/1612.03144

Complete

5. [**Optional**] Focal loss for dense object detection
   Motivation: important observation on class imbalance, it's negative effect on training and how to address it
   https://arxiv.org/abs/1708.02002

   Complete

2 Theory

None

3 Practice

Previous homework assignments gave you confidence in your ability to train classification networks and an understanding that (unfortunately) training takes a long time with modest resources. A few items that challenge us with respect to developing and training networks for more complex vision problems:

- High resolution images (like you would like for typical vision applications) result in a high input data volume and a proportional increase in computation and memory.
- Detection, segmentation, depth estimation and motion estimation networks tend to have a lot of pieces to them which makes their implementation more time consuming.
- Training routines tend to be more complex with a combination of pre trained and un trained network pieces, multiple loss functions, class imbalances and a larger number of hyper parameters.
- Accuracy evaluation procedures, required to compare to reference implementations, are more involved.

Can you overcome all of these issues, implement and train standard vision networks and eventually try out your own ideas? **Yes**. However, Is all of this practical within the time period of a 1 week homework assignment using Google Colab? Probably not.

As such, for this and some of the subsequent homework assignments we'll be more of a user of existing network implementations. However, always look for places to experiment beyond the given model where practical.

6. Pixel segmentation with a modified UNet. Understand the following example (https://www.tensorflow.org/beta/tutorials/images/segmentation) and run it in Google Colab.

Complete

7. Pixel segmentation with DeepLab. Understand the following example (https://colab.research.google.com/github/tensorflow/models/blob/master/research/deeplab/deeplab_demo.ipynb) and run it in Google Colab.

Complete

8. Object detection with pre trained TF-Hub models. Understand the following example (https://colab.research.google.com/github/tensorflow/hub/blob/master/examples/colab/object_detection.ipynb) and run it in Google Colab.

Complete

9. Object segmentation with Mask R-CNN. Understand the following example (https://colab.research.google.com/github/tensorflow/hub/blob/master/examples/colab/object_detection.ipynb) and run demo.ipynb in Google Colab.

Complete