References

Week 1:

- [The Sequential model](#) (TensorFlow Documentation)
- [The Functional API](#) (TensorFlow Documentation)

Week 2:

- [Deep Residual Learning for Image Recognition](#) (He, Zhang, Ren & Sun, 2015)
- [deep-learning-models/resnet50.py/](#) (GitHub: fchollet)
- [MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications](#) (Howard, Zhu, Chen, Kalenichenko, Wang, Weyand, Andreetto, & Adam, 2017)
- [MobileNetV2: Inverted Residuals and Linear Bottlenecks](#) (Sandler, Howard, Zhu, Zhmoginov &Chen, 2018)
- [EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks](#) (Tan & Le, 2019)

Week 3:

- [You Only Look Once: Unified, Real-Time Object Detection](#) (Redmon, Divvala, Girshick & Farhadi, 2015)
- [YOLO9000: Better, Faster, Stronger](#) (Redmon & Farhadi, 2016)
- [YAD2K](#) (GitHub: allanzelener)
- [YOLO: Real-Time Object Detection](#)
- [Fully Convolutional Architectures for Multi-Class Segmentation in Chest Radiographs](#) (Novikov, Lenis, Major, Hladůvka, Wimmer & Bühler, 2017)
- [Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks](#) (Dong, Yang, Liu, Mo & Guo, 2017)
- [U-Net: Convolutional Networks for Biomedical Image Segmentation](#) (Ronneberger, Fischer & Brox, 2015)

Week 4:

- [FaceNet: A Unified Embedding for Face Recognition and Clustering](#) (Schroff, Kalenichenko & Philbin, 2015)
- [DeepFace: Closing the Gap to Human-Level Performance in Face Verification](#) (Taigman, Yang, Ranzato & Wolf)
- [facenet](#) (GitHub: davidsandberg)
- [How to Develop a Face Recognition System Using FaceNet in Keras](#) (Jason Brownlee, 2019)
- [keras-facenet/notebook/tf_to_keras.ipynb](#) (GitHub: nyoki-mtl)
- [A Neural Algorithm of Artistic Style](#) (Gatys, Ecker & Bethge, 2015)
- [Convolutional neural networks for artistic style transfer](#)
- [TensorFlow Implementation of "A Neural Algorithm of Artistic Style"](#)
- [Very Deep Convolutional Networks For Large-Scale Image Recognition](#) (Simonyan & Zisserman, 2015)
- [Pretrained models](#) (MatConvNet)