

REFERENCES

Abadi, M. *et al.* (2016) ‘TensorFlow: A System for Large-Scale Machine Learning’, in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. Savannah, GA: {USENIX} Association, pp. 265–283. Available at: <https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>.

COCO - Common Objects in Context (no date). Available at: <http://cocodataset.org/#detection-eval> (Accessed: 17 June 2019).

Descending into ML: Training and Loss | Machine Learning Crash Course | Google Developers (no date). Available at: <https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss> (Accessed: 11 June 2019).

Geron, A. (2019) *Hands-on Machine Learning with Scikit-Learn, Keras and Tensorflow*. O’Reilly Media.

Google (no date) *Developer Guide | Protocol Buffers | Google Developers*. Available at: <https://developers.google.com/protocol-buffers/docs/overview> (Accessed: 11 June 2019).

Howard, A. G. *et al.* (2017) ‘MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications’. Available at: <http://arxiv.org/abs/1704.04861>.

Huang, J. *et al.* (2017) ‘Speed/accuracy trade-offs for modern convolutional object detectors’, *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua, pp. 3296–3305. doi: 10.1109/CVPR.2017.351.

Jia, Y. *et al.* (2014) ‘Caffe: Convolutional Architecture for Fast Feature Embedding’. Available at: <http://arxiv.org/abs/1408.5093>.

Laplante, P. A. (2004) *Real-Time Systems Design and Analysis*. Wiley. Available at: <https://books.google.co.id/books?id=1qDuaC17yxAC>.

Liu, W. *et al.* (2016) ‘SSD: Single shot multibox detector’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9905 LNCS, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

Lu, M. *et al.* (2005) ‘Transportation Planning and Technology Technical Feasibility of Advanced Driver Assistance Systems (ADAS) for Road Traffic Safety’, *Transportation Planning and Technology*, 283(February 2013), pp. 167–187. doi: 10.1080/03081060500120282.

Lysukhin (no date) *GitHub -lysukhin/tensorflow-object-detection-cpp: A C++ example of running TensorFlow Object Detection model in live mode*. Available at: <https://github.com/lysukhin/tensorflow-object-detection-cpp> (Accessed: 15 February 2019).

Redmon, J. *et al.* (2015) ‘You Only Look Once: Unified, Real-Time Object Detection’. Available at: <http://arxiv.org/abs/1506.02640>.

Ren, S. *et al.* (2017) ‘Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.’, *IEEE transactions on pattern analysis and machine intelligence*, 39(6), pp. 1137–1149. doi: 10.1109/TPAMI.2016.2577031.

Rosebrock, A. (2016) *Intersection over Union (IoU) for object detection - PyImageSearch*. Available at: <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/> (Accessed: 17 June 2019).

Shanmugamani, R. (2018) *Deep Learning for Computer Vision: Expert Techniques to Train Advanced Neural Networks Using TensorFlow and Keras*. Packt Publishing. Available at: <https://books.google.co.id/books?id=dgdOswEACAAJ>.

Szegedy, C. *et al.* (2015) ‘Going deeper with convolutions’, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

Ting, K. M. (2010) ‘Precision and Recall’, in Sammut, C. and Webb, G. I. (eds) *Encyclopedia of Machine Learning*. Boston, MA: Springer US, p. 781. doi: 10.1007/978-0-387-30164-8_652.

Yosinski, J. *et al.* (2014) ‘How transferable are features in deep neural networks?’, 27. Available at: <http://arxiv.org/abs/1411.1792>.

Ziebinski, A. *et al.* (2017) ‘Review of advanced driver assistance systems (ADAS)’, *AIP Conference Proceedings*, 1906. doi: 10.1063/1.5012394.