

# Deep Learning mit TensorFlow: Entwicklung eines Object Detection Modells

Dominic Regitz,

Forschungsgruppe Data Science, Institut für Digitalen Wandel (IDW)

## Einführung

- Objekterkennung stellt ein Verfahren des supervised learning zur Identifikation bereits bekannter Objekte innerhalb eines spezifischen Objektraumes dar
- Der Einsatz von Object Detection Modellen weist hierbei ein hohes Maß potentieller Einsatzfälle (sowohl in der Industrie als auch im alltäglichen Umfeld) auf

## Zielsetzung

- Erprobung eines ersten Prototyps zur Identifikation vordefinierter Handzeichen
- Schrittweise Identifikation insgesamt sechs unterschiedlicher Objekte
- Umsetzung der Objekterkennung auf Basis von Bildern sowie in Echtzeit durch Nutzung von Kameras bzw. Webcams

## Zusammenfassung

- Entwicklung von Object Detection Modellen wird durch Open-Source-Plattformen wie TensorFlow stark erleichtert
- Das Training solcher Modelle führt auch bei einer begrenzten Menge an Trainingsdaten zu verlässlichen Ergebnissen.

## Methodik

### Datenbeschaffung & Aufbereitung:

- Beschaffung von Bildern für alle zu erkennenden Objekte
- Umfang: ca. 60 Bilder pro Kategorie



- Labeling der Bilder durch die Angabe von Wertebereichen und zugehörigen Bezeichnungen -> supervised learning

### Training des Modells:

- Aufteilung der Bilder in Trainings- und Testdaten (Verhältnis: 80/20) und anschließendes Ausführen der Trainingsschritte

```

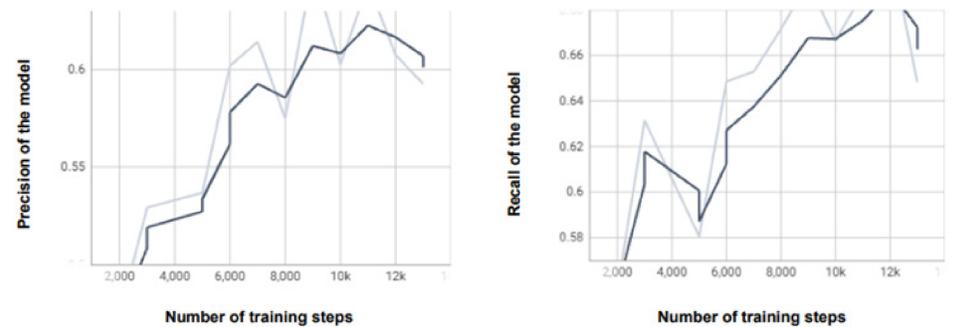
Anaconda Prompt (miniconda3)
INFO:tensorflow:Step 10100 per-step time 4.343s
I0615 11:25:51.068831 8696 model_lib_v2.py:705] Step 10100 per-step time 4.343s
INFO:tensorflow: {'Loss/classification_loss': 0.07369863,
'Loss/localization_loss': 0.020239739,
'Loss/regularization_loss': 0.111774,
'Loss/total_loss': 0.20571238,
'learning_rate': 0.07338293}
I0615 11:25:51.068831 8696 model_lib_v2.py:708] {'Loss/classification_loss': 0.07369863,
'Loss/localization_loss': 0.020239739,
'Loss/regularization_loss': 0.111774,
'Loss/total_loss': 0.20571238,
'learning_rate': 0.07338293}
INFO:tensorflow:Step 10200 per-step time 4.119s
I0615 11:32:42.921210 8696 model_lib_v2.py:705] Step 10200 per-step time 4.119s
INFO:tensorflow: {'Loss/classification_loss': 0.07046095,
'Loss/localization_loss': 0.013657033,
'Loss/regularization_loss': 0.11122487,
'Loss/total_loss': 0.19534285,
'learning_rate': 0.073240966}
I0615 11:32:42.921210 8696 model_lib_v2.py:708] {'Loss/classification_loss': 0.07046095,
'Loss/localization_loss': 0.013657033,
'Loss/regularization_loss': 0.11122487,
'Loss/total_loss': 0.19534285,
'learning_rate': 0.073240966}
INFO:tensorflow:Step 10300 per-step time 4.245s
I0615 11:39:47.438078 8696 model_lib_v2.py:705] Step 10300 per-step time 4.245s
INFO:tensorflow: {'Loss/classification_loss': 0.034308314,
'Loss/localization_loss': 0.021576138,
'Loss/regularization_loss': 0.11072304,
'Loss/total_loss': 0.1666075,

```

- Ermittlung der Performance des Modells (Evaluation)
- Fine-Tuning durch das Hinzufügen:
  - zusätzlicher Datensätze
  - weiterer Trainingsschritte

## Ergebnis

- Optimum nach 11.000 Trainingsschritten erreicht mit einer Average Precision von rund 65% und einem Average Recall von rund 69%



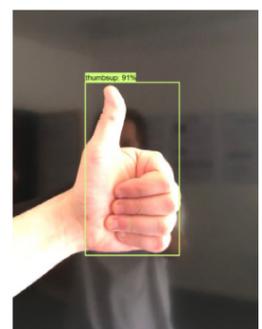
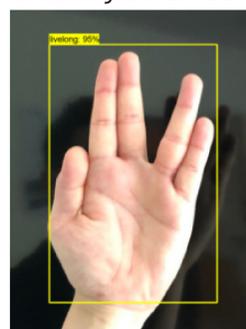
- \*Precision: ratio between True Positives (TP) and the actual results ->  $TP / (TP + False\ Positives)$
- \*Recall: measure of the model of correctly identifying True Positives ->  $TP / (TP + False\ Negatives)$

```

Anaconda Prompt (miniconda3) - python Tensorflow\models\research\object_detection\model_main_tf2.py --mod...
INFO:tensorflow:Performing evaluation on 36 images.
I0615 12:55:53.442982 25464 coco_evaluation.py:293] Performing evaluation on 36 images.
creating index...
index created!
INFO:tensorflow:Loading and preparing annotation results...
I0615 12:55:53.442982 25464 coco_tools.py:116] Loading and preparing annotation results...
INFO:tensorflow:DONE (t=0.00s)
I0615 12:55:53.442982 25464 coco_tools.py:138] DONE (t=0.00s)
creating index...
index created!
Running per image evaluation...
Evaluate annotation type *bbox*
DONE (t=0.08s).
Accumulating evaluation results...
DONE (t=0.08s).
Average Precision (AP) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.644
Average Precision (AP) @[ IoU=0.50 | area= all | maxDets=100 ] = 0.872
Average Precision (AP) @[ IoU=0.75 | area= all | maxDets=100 ] = 0.690
Average Precision (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
Average Precision (AP) @[ IoU=0.50:0.95 | area= medium | maxDets=100 ] = 0.775
Average Precision (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.642
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 1 ] = 0.687
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets= 10 ] = 0.754
Average Recall (AR) @[ IoU=0.50:0.95 | area= all | maxDets=100 ] = 0.754
Average Recall (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
Average Recall (AR) @[ IoU=0.50:0.95 | area= medium | maxDets=100 ] = 0.783
Average Recall (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.757
INFO:tensorflow:Eval metrics at step 11000
I0615 12:55:53.583592 25464 model_lib_v2.py:1015] Eval metrics at step 11000
INFO:tensorflow: + DetectionBoxes Precision/mAP: 0.644057

```

- Objekterkennung trotz geringer Anzahl an Bildern und Trainings-schritten in Echtzeit sowie durch Bildübergabe funktionstüchtig



## Literatur

- Abadi, Martin, et al. «TensorFlow: a system for Large-Scale machine learning.» 12th USENIX symposium on operating systems design and implementation (OSDI 16). 2016.
- Abadi, Martin, et al. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467 (2016)

## Kontakt

Prof. Dr. Wolfram Höpken  
Leiter IDW / Forschungsgruppe Data Science  
+49 751 501 9764  
wolfram.hoepken@rwu.de