



# Introduction to TensorFlow

Ali Athar & Sabarinath Mahadevan

`<lastname>@vision.rwth-aachen.de`

RWTH Aachen University - Computer Vision Group

`http://www.vision.rwth-aachen.de/`

2019-12-12



# Outline

1. Deep Learning Frameworks
2. TensorFlow Example
3. Tips & Tricks



## Deep Learning Frameworks





## Got TensorFlow?



- ▶ Install from:  
`http://www.tensorflow.org/install/`
- ▶ CPU and GPU version available
- ▶ Performance guide: `https://www.tensorflow.org/performance/performance\_guide`

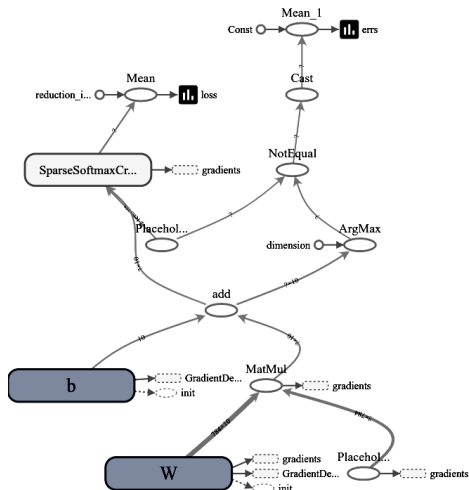


## Computational Graph

- ▶ First, construct graph
- ▶ Graph defines computations
- ▶ Actual computations: later
- ▶ Defining a graph
  - ▶ Placeholder:- serves as an input feed
  - ▶ Variables:- variables that are optimised (eg: weights)
  - ▶ Model:- defines the mathematical functions performed by the network
  - ▶ Loss:- defines the loss function to optimise
  - ▶ Optimiser:- chooses the optimisation strategy



# Computational Graph: Example





## TensorBoard

- ▶ TensorFlow tool for visualization
- ▶ During training create and write out summaries
- ▶ Later look at the data in TensorBoard
- ▶ To run: `tensorboard`  
`--logdir=path/to/log-directory`
- ▶ Then open browser and open `computer_name:6006`
- ▶ Tutorial: [https://www.tensorflow.org/get\\_started/summaries\\_and\\_tensorboard](https://www.tensorflow.org/get_started/summaries_and_tensorboard)



## Tipps and Tricks 1

- ▶ `nvidia-smi`: show GPU load
- ▶ `htop`: show CPU load
- ▶ GPU memory usage: training data + model (keep in mind: batch size is also a hyperparameter, do not compare models with different batch size)
- ▶ Use data augmentations (rotation, translation, mirroring, scaling, gamma, etc.)
- ▶ Normalize your data





## Tipps and Tricks 2

- ▶ Hyperparameters: learning rate, batch size is also a hyperparameter
- ▶ Multiple runs: keep in mind that training is randomized, retrain model multiple times to be sure it is actually better than another model
- ▶ Visualize everything: look at your training data, what is passed into a batch, look at your results, not just final score