Agenda

- CSBDeep
- Keras / TensorFlow
- Tips & Tricks

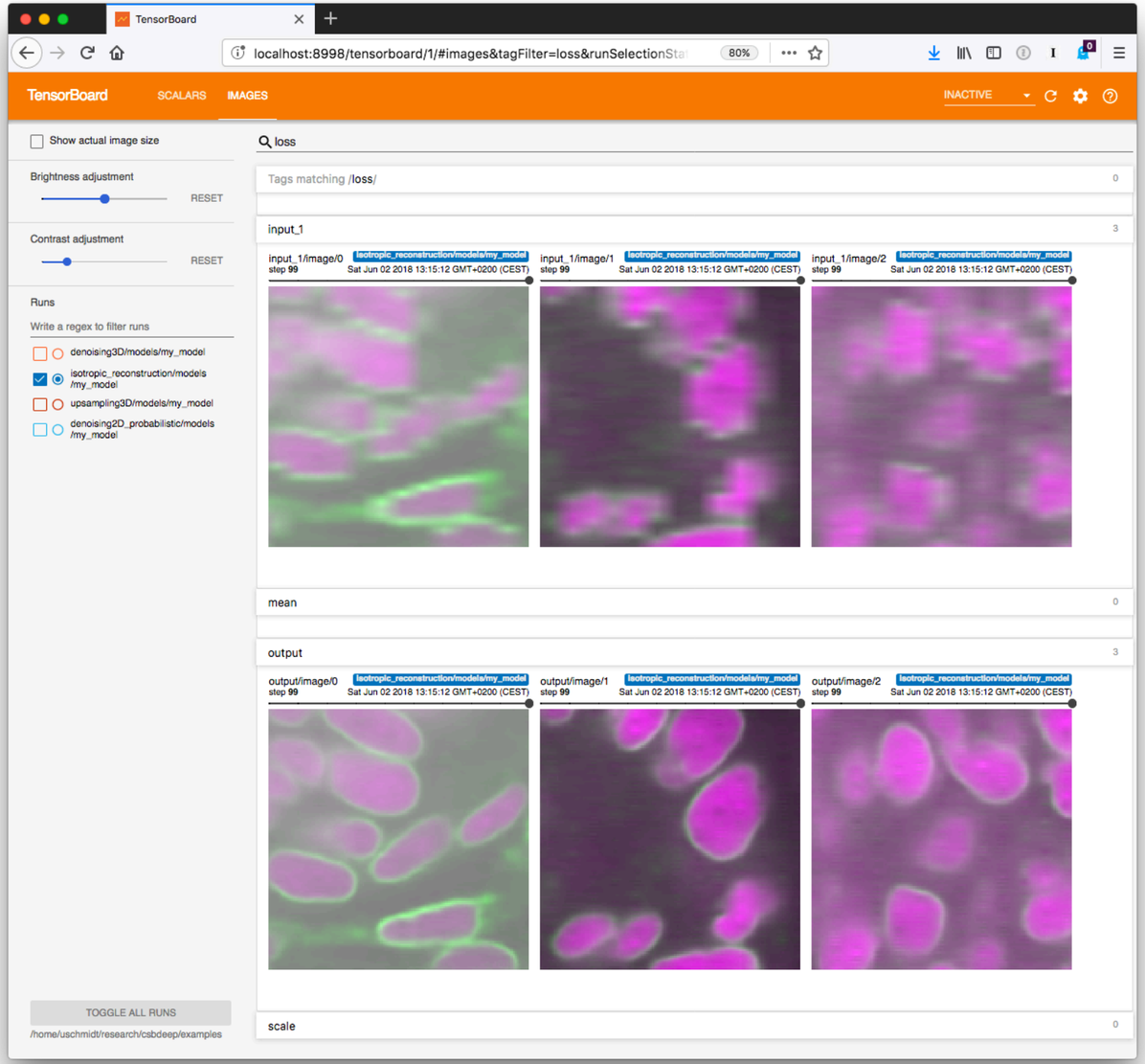
CSBDeep

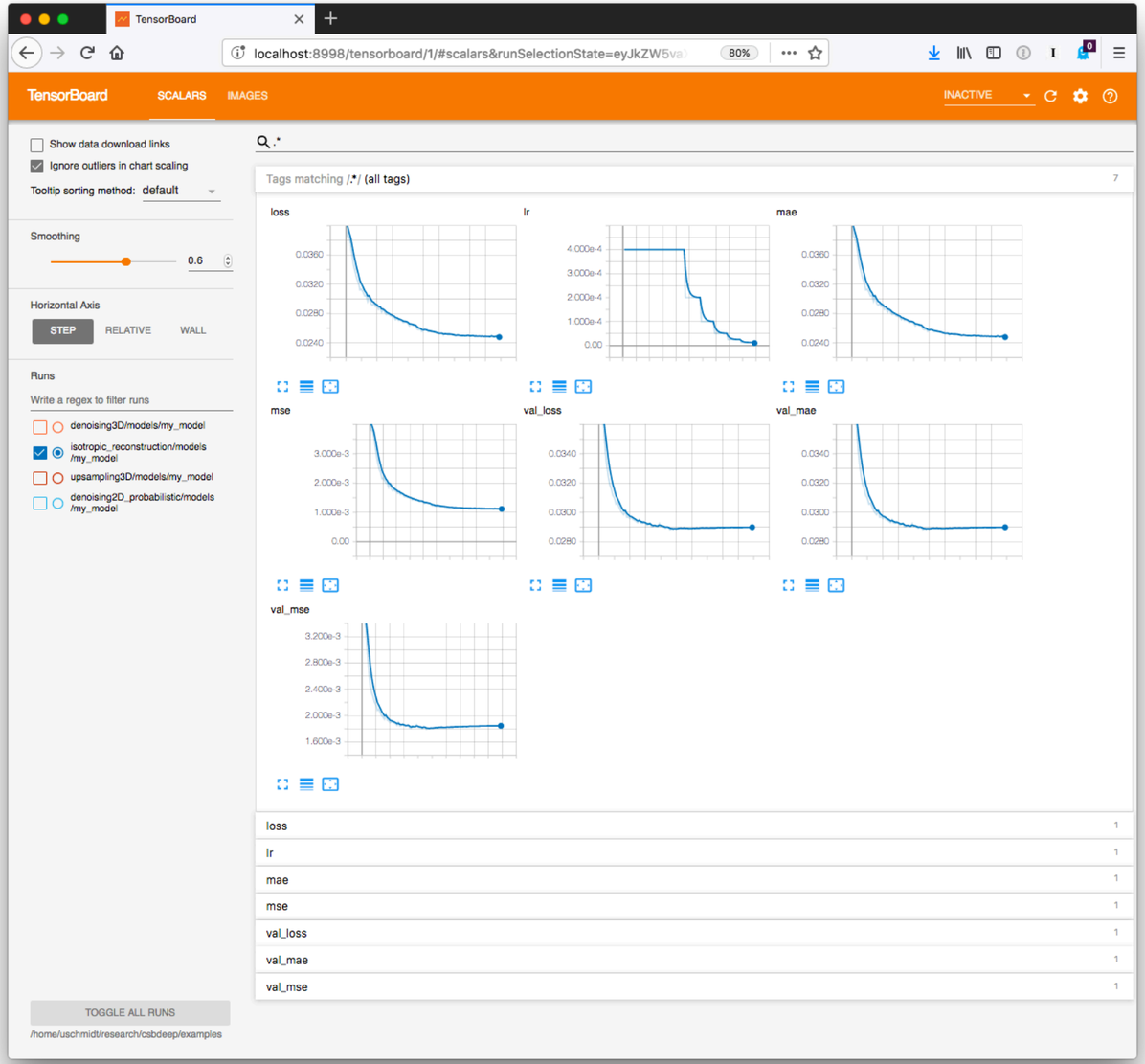
https://github.com/csbdeep/csbdeep

- Training data generation
- Input normalization
- Tiling of big stacks

Keras / TensorFlow

Keras Docs
TensorFlow Docs


TensorBoard
localhost:8998/tensorboard/1/#images&tagFilter=loss&runSelectionSta
80%
TensorBoard
SCALARS
IMAGES
INACTIVE
Show actual image size
Brightness adjustment
RESET
Contrast adjustment
RESET
Runs
Write a regex to filter runs
denoising3D/models/my_model
isotropic_reconstruction/models/my_model
upsampling3D/models/my_model
denoising2D_probabilistic/models/my_model
TOGGLE ALL RUNS
/home/uschmidt/research/csbdeep/examples
loss
Tags matching /loss/
0
input_1
3
input_1/image/0
step 99
Sat Jun 02 2018 13:15:12 GMT+0200 (CEST)
input_1/image/1
step 99
Sat Jun 02 2018 13:15:12 GMT+0200 (CEST)
input_1/image/2
step 99
Sat Jun 02 2018 13:15:12 GMT+0200 (CEST)
mean
0
output
3
output/image/0
step 99
Sat Jun 02 2018 13:15:12 GMT+0200 (CEST)
output/image/1
step 99
Sat Jun 02 2018 13:15:12 GMT+0200 (CEST)
output/image/2
step 99
Sat Jun 02 2018 13:15:12 GMT+0200 (CEST)
scale
0


TensorBoard
localhost:8998/tensorboard/1/#scalars&runSelectionState=eyJkZW5v
80%
TensorBoard
SCALARS
IMAGES
INACTIVE
Show data download links
Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing
0.6
Horizontal Axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
denoising3D/models/my_model
isotropic_reconstruction/models/my_model
upsampling3D/models/my_model
denoising2D_probabilistic/models/my_model
TOGGLE ALL RUNS
/home/uschmidt/research/csbdeep/examples
.*
Tags matching /.*/ (all tags)
7
loss
lr
mae
mse
val_loss
val_mae
val_mse
loss 1
lr 1
mae 1
mse 1
val_loss 1
val_mae 1
val_mse 1

Deployment

https://www.tensorflow.org/deploy/

Tips & Tricks

model/network architecture

- start simple
  - establish baseline
  - make model only more complicated if needed
- look for papers / articles that solve similar problem to yours
- code to define network typically the easy part

data preparation/processing

- most time/code typically spent on data preparation
  - data labeling often biggest effort
- problem of imbalanced data
- if you already have good handcrafted features,
  try using them as additional input

problem formulation/evaluation

- model evaluation ideally corresponds to training loss
- adapt problem formulation to exploit strengths of DL
- don't solve harder problem than need be
- is your problem unique in some way?

training

- data shuffling
- checkpointing
- use Adam
- LR (try 2-3 different orders of magnitude, e.g. 0.01, 0.001, 0.0001)
- ReduceLROnPlateau / EarlyStopping
- batch size selection
  (e.g. 8-16 for segmentation/regression, 64-512 for classification)

debugging

- overfit on small data first
- check input normalization
- train and test should be done under same conditions (e.g. data normalization)
- forgot to shuffle data (e.g. train/val split beforehand)
- intuition for fluctuations of losses and metrics comes with experience

Q & A