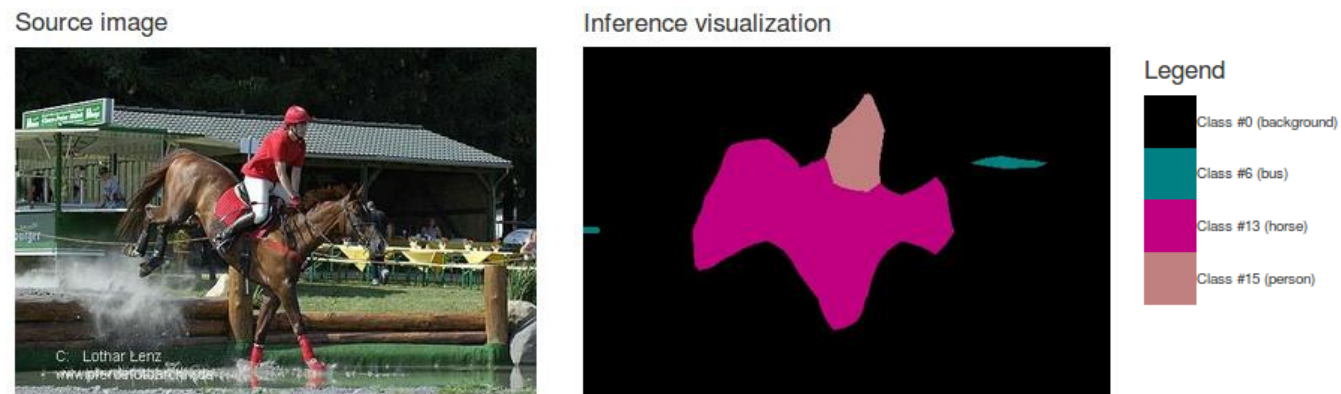


Using DIGITS to train a Semantic Segmentation neural network

Segmentation

PASCAL VOC 2012 data : 2GB tar file.



Use `prepare_pascal_voc.sh` script to create a train/val split of the labelled images:

```
$ ./prepare_pascal_voc_data.sh /data/VOCtrainval_11-May-2012.tar ./voc-data
Expanding /data/VOCtrainval_11-May-2012.tar
Copying data into ./voc-data
Processing train data
Processing val data
Done!
```

labels

#0: background

#1: aeroplane

#2: bicycle

#3: bird

#4: boat

#5: bottle

#6: bus

#7: car

#8: cat

#9: chair

#10: cow

#11: diningtable

#12: dog

#13: horse

#14: motorbike

#15: person

#16: pottedplant

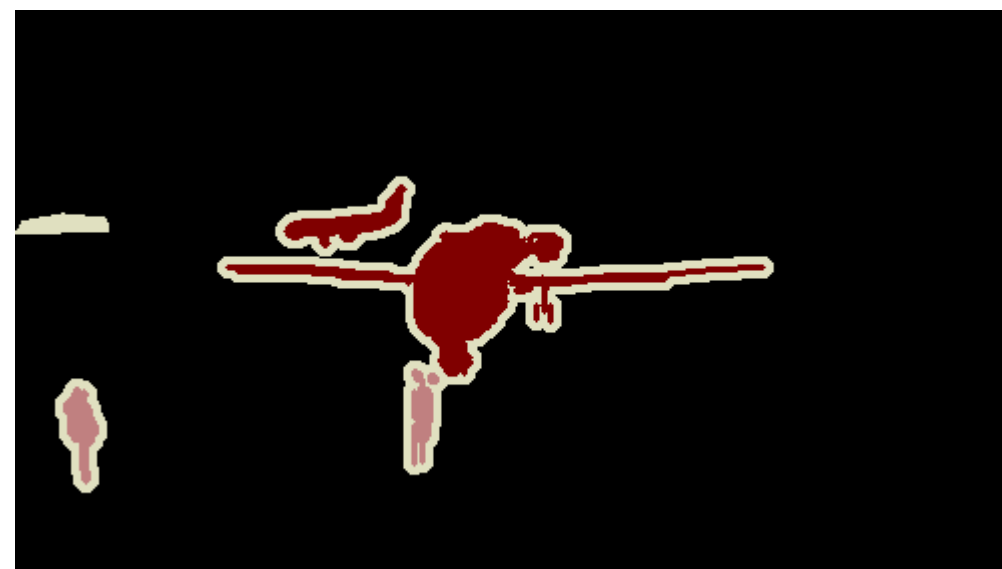
#17: sheep

#18: sofa

#19: train

#20: tvmonitor

#255: undefined/don't care



Loading dataset to DIGITS

Use the `prepare_pascal_voc.sh` script to create a train/val split of the labelled images:

```
$ ./prepare_pascal_voc_data.sh /data/VOctrainval_11-May-2012.tar ./voc-data
Expanding /data/VOctrainval_11-May-2012.tar
Copying data into ./voc-data
Processing train data
Processing val data
Done!
```

Create train_db DB

Entry Count
1464

Feature shape ⓘ
(3, 281, 500)

Label shape ⓘ
(1, 281, 500)

labels DB
/home/greg/ws/digits/digits/jobs/20160808-152315-5e02/train_db/labels
[Explore the db](#)

features DB
/home/greg/ws/digits/digits/jobs/20160808-152315-5e02/train_db/features
[Explore the db](#)

DB create log file
[create_train_db_db.log](#)

Feature image folder ⓘ
/workspace/digits/examples/semantic-segmentation/voc-data/train/images

Label image folder ⓘ
/workspace/digits/examples/semantic-segmentation/voc-data/train/labels

% for validation ⓘ
10

☒ **Separate validation images**

Validation feature image folder ⓘ
/workspace/digits/examples/semantic-segmentation/voc-data/val/images

Validation label image folder ⓘ
/workspace/digits/examples/semantic-segmentation/voc-data/val/labels

Class labels (optional) ⓘ
/workspace/digits/examples/semantic-segmentation/pascal-voc-classes.txt

Color map specification ⓘ
From label image

Color map file ⓘ
file

Channel conversion ⓘ
None

Note: the recommended label encoding is PNG.

Feature Encoding ⓘ
PNG (lossless)

Label Encoding ⓘ
PNG (lossless)

Encoder batch size ⓘ
32

Number of encoder threads ⓘ
4

DB backend
LMDB

Enforce same shape ⓘ
No

Group Name

Dataset Name
PASCAL-VOC-Segmentation-Class

[Create](#)

Feature / label database

Exploring PASCAL-VOC-Segmentation-Class (/home/greg/ws/digits/digits/jobs/20160808-152315-5e02/train_db/features) images

[Show all images](#)

Items per page: 10 - **25** - 50 - 100

« 0 1 2 3 4 5 ... 58 »

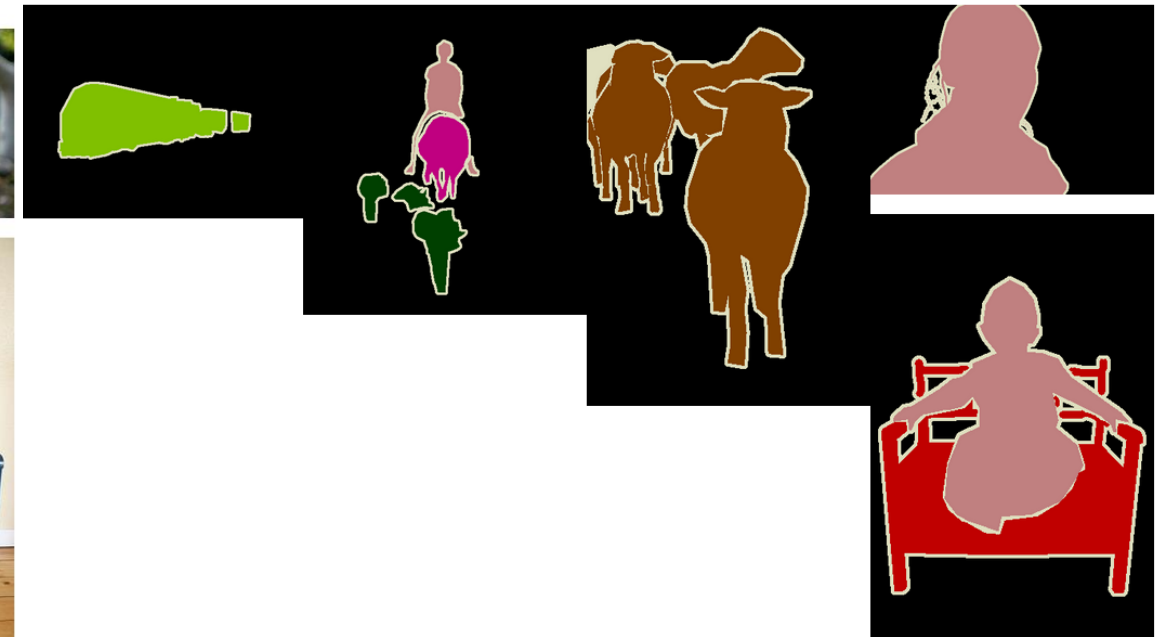


Exploring PASCAL-VOC-Segmentation-Class (/home/greg/ws/digits/digits/jobs/20160808-152315-5e02/train_db/labels) images

[Show all images](#)

Items per page: 10 - **25** - 50 - 100

« 0 1 2 3 4 5 ... 58 »



Model creation

- Use the `net_surgery.py`

```
$ ./net_surgery.py
Downloading files (this might take a few minutes)...
Downloading https://raw.githubusercontent.com/BVLC/caffe/rc3/models/bvlc_alexnet/deploy.prototxt...
Downloading http://dl.caffe.berkeleyvision.org/bvlc_alexnet.caffemodel...
Loading Alexnet model...
...
Saving FCN-Alexnet model to fcn_alexnet.caffemodel
```

- Set Subtract mean to none
- Select the dataset that was created in the previous section
- Set the base learning rate to 0.0001
- Select the Custom Network tab
- Make sure the Caffe sub-tab is selected
- Copy/paste this `prototxt`
- In Pretrained model(s) specify the path to the pre-trained FCN-Alexnet
- Note that since we are using a batch size of 1 it is not possible to train this network on multiple GPUs

Model creation form

Select Dataset ?

PASCAL-VOC-Segmentation-Class
kitti-data

PASCAL-VOC-Segmentation-Class
Done Tue Jan 17, 11:19:43 AM

- DB backend: lmdb
- Create train_db DB
 - Entry Count: 1464
 - Feature shape (3, 281, 500)
 - Label shape (1, 281, 500)
- Create val_db DB
 - Entry Count: 1449
 - Feature shape (3, 366, 500)
 - Label shape (1, 366, 500)

Python Layers ?

Server-side file ?

☐ Use client-side file

Solver Options

Training epochs ?
11

Snapshot interval (in epochs) ?
1.0

Validation interval (in epochs) ?
1.0

Random seed ?
[none]

Batch size ? multiples allowed
[network defaults]

Batch Accumulation ?

Solver type ?
Stochastic gradient descent (SGD)

Base Learning Rate ? multiples allowed
0.0001

☐ Show advanced learning rate options

Data Transformations

Subtract Mean ?
None

Crop Size ?
none

Standard Networks Previous Networks Pretrained Networks Custom Network

Caffe Torch

Custom Network ? Visualize

```
1 # data layers
2 layer {
3   name: "data"
4   type: "Data"
5   top: "data"
6   include {
7     phase: TRAIN
8   }
9   data_param {
10    batch_size: 1
11    backend: LMDB
12  }
13 }
14 layer {
15   name: "label"
16   type: "Data"
17   top: "label"
18   include {
19     phase: TRAIN
20   }
21   data_param {
22    batch_size: 1
23    backend: LMDB
24  }
25 }
26 layer {
27   name: "data"
28   type: "Data"
29   top: "data"
30   include {
31     phase: TEST
32   }
33   data_param {
34    batch_size: 1
35    backend: LMDB
36  }
37 }
38 layer {
39   name: "label"
40   type: "Data"
```

Pretrained model(s) ?
/workspace/digits/examples/semantic-segmentation/fcn_alexnet.caffemodel

Use this many GPUs (next available)
1

or

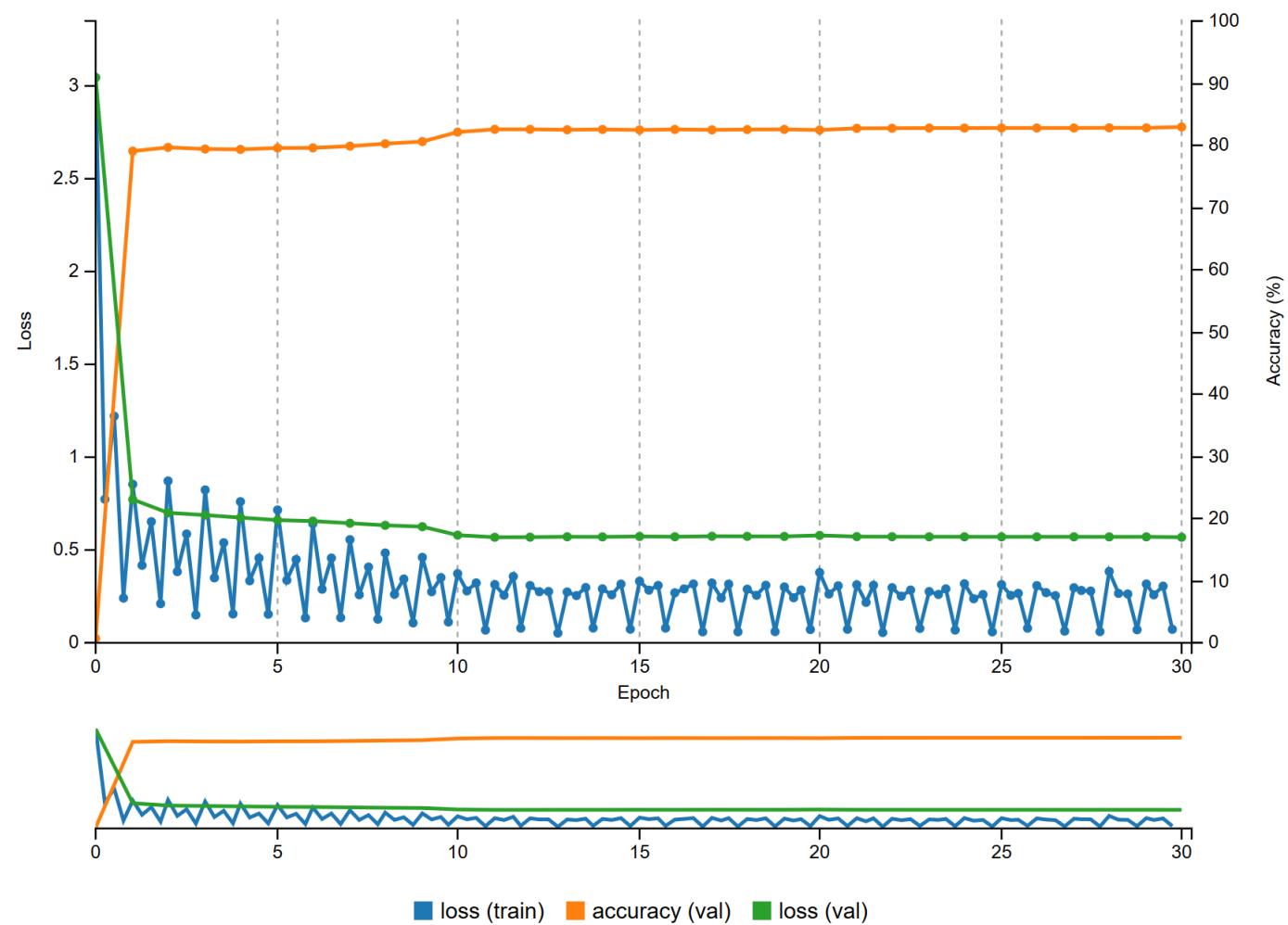
Select which GPU[s] you would like to use ?
#0 - Tesla P100-SXM2-16GB (15.9 GB memory)
#1 - Tesla P100-SXM2-16GB (15.9 GB memory)
#2 - Tesla P100-SXM2-16GB (15.9 GB memory)
#3 - Tesla P100-SXM2-16GB (15.9 GB memory)

Group Name ?

Model Name ?
PASCAL-VOC-Segmentation

Create

Training



Inference

Trained Models

Select Model

Epoch #30

Download Model

Make Pretrained Model

Select Visualization Method

Image Segmentation

Visualization Options

Display segmented image.

Colormap ?

From dataset

Inference Options

☒ Do not resize input image(s) ?

Test a single image

Image Path ?

Upload image

Browse...

2008_000142.jpg

☐ Show visualizations and statistics ?

Test One

Test a list of images

Upload Image List

Browse...

Accepts a list of filenames or urls (you can use your val.txt file)

Image folder (optional)

Relative paths in the text file will be prepended with this value before reading

Number of images use from the file

Source image



Inference visualization



■ cow ■ horse ■ person