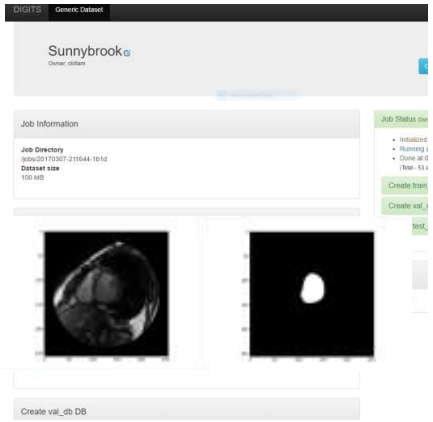
MEDICAL IMAGE SEGMENTATION WITH DIGITS

Hyungon Ryu, Jack Han
Solution Architect
NVIDIA Corporation

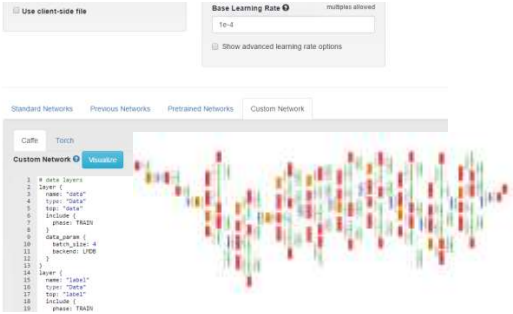
MEDICAL IMAGE SEGMENTATION WITH DIGITS

Overview

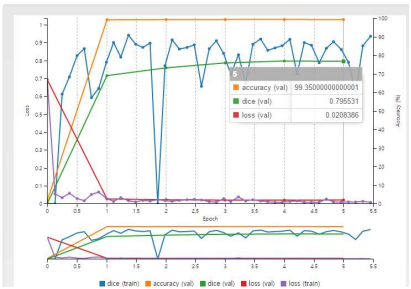
Prepare Dataset



Configure DL Model



DL Training



Segmentation

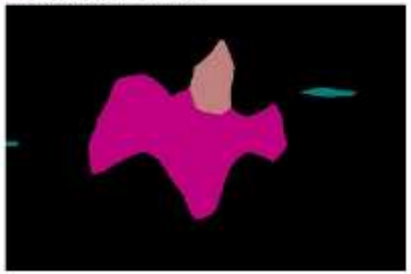
Segmentation

SEGMENTATION

Pascal VOC 2012 dataset

Source Image

Inference visualization



Legend

#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

http://host.robots.ox.ac.uk/pascal/VOC/

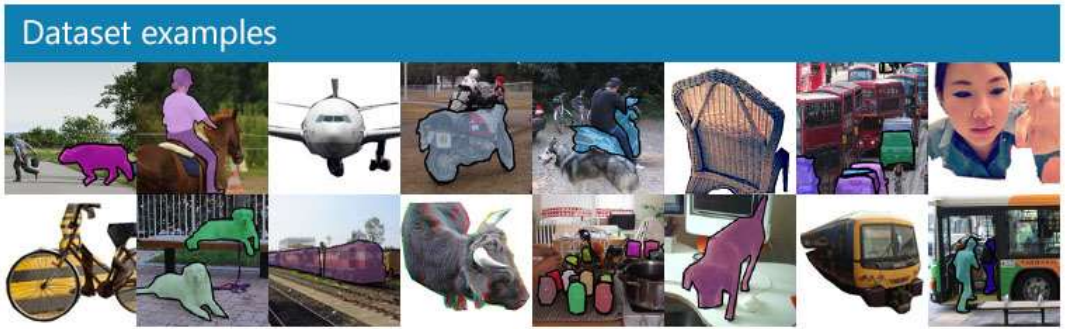
SEGMENTATION

MS COCO dataset

What is COCO?

COCO is a new image recognition, segmentation, and captioning dataset. COCO has several features:

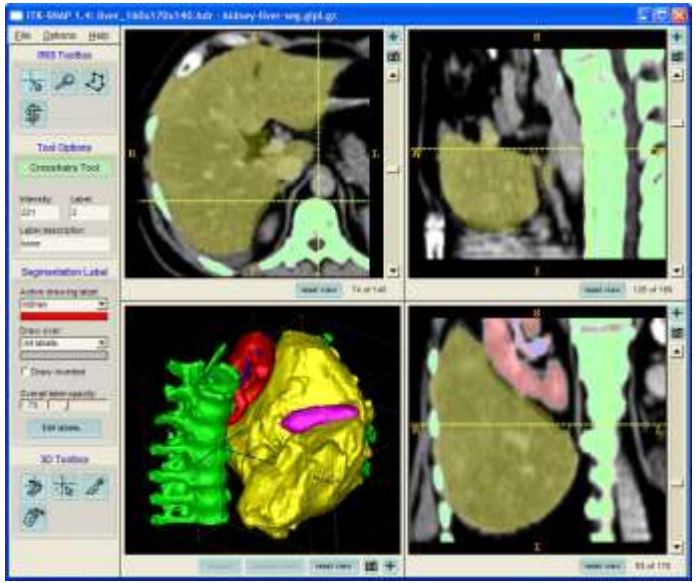
- Object segmentation
- Recognition in Context
- Multiple objects per image
- More than 300,000 images
- More than 2 Million instances
- 80 object categories
- 5 captions per image
- Keypoints on 100,000 people



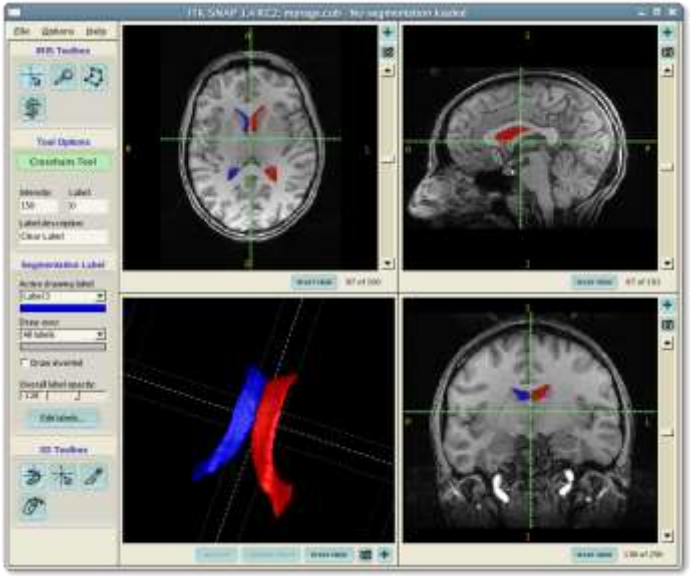
http://mscoco.org/

SEGMENTATION

ITK-SNAP



Kidney-Liver Segmentation

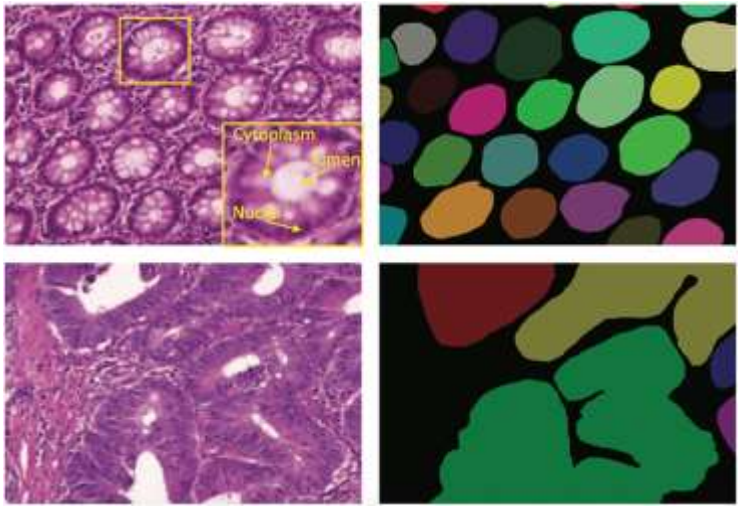


Brain Segmentation

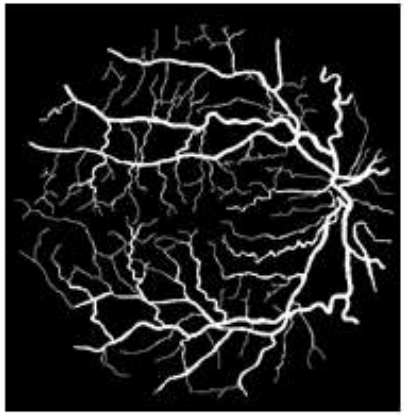
http://www.itksnap.org/

SEGMENTATION

Medical Image



Cancer Cell



Vessel Segmentation

Dataset

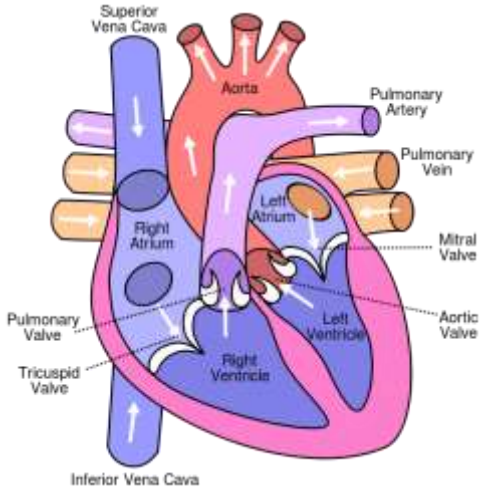
CARDIAC MR LEFT VENTRICLE SEGMENTATION

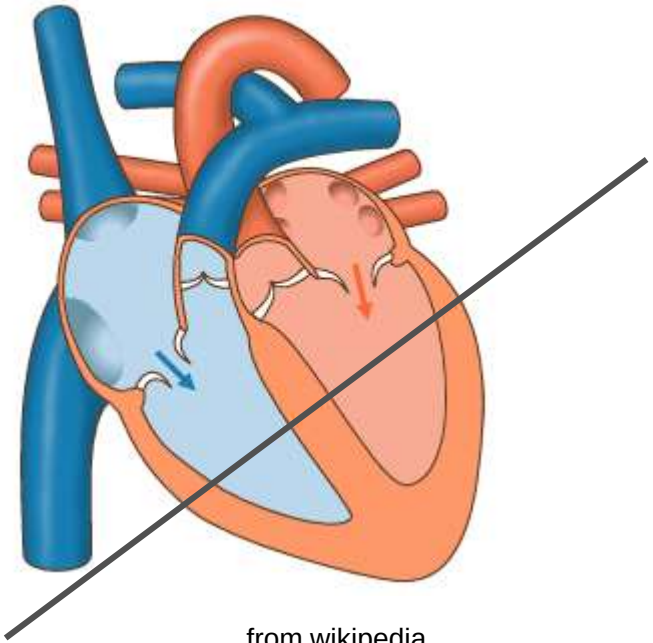
MIDAS Journal – Cardiac MR Left Ventricle Segmentation Challenge

http://hdl.handle.net/10380/3070

http://smial.sri.utoronto.ca/LV_Challenge/Home.html

SLICE VIEW

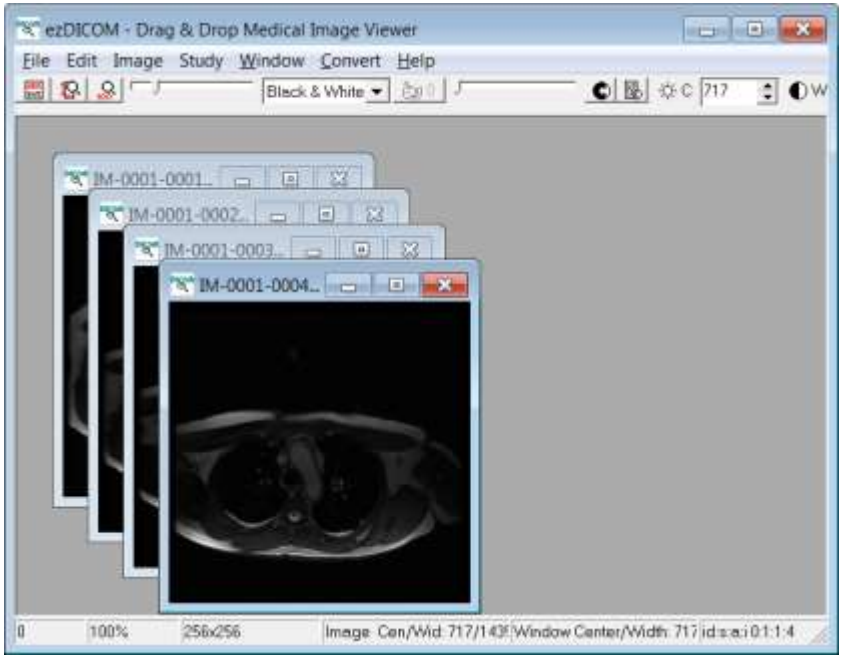


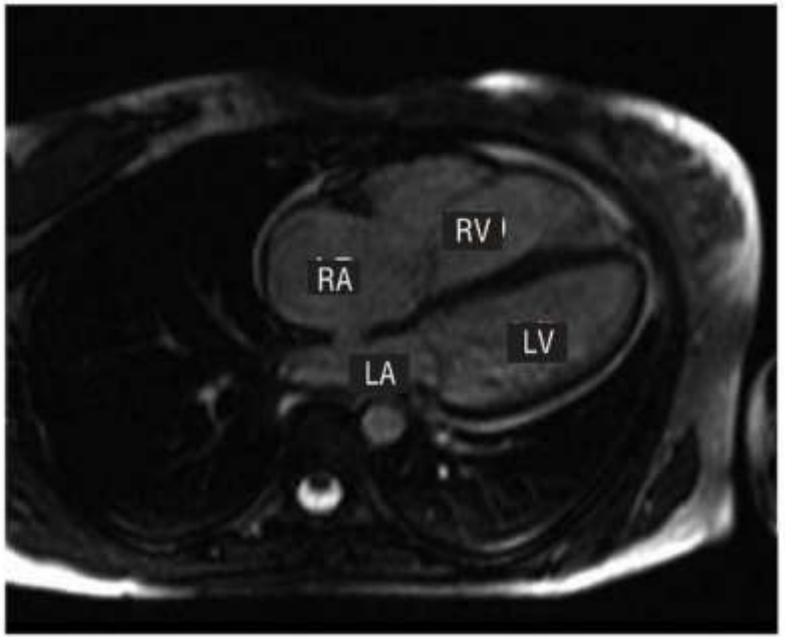


from wikipedia

DICOM VIEWER

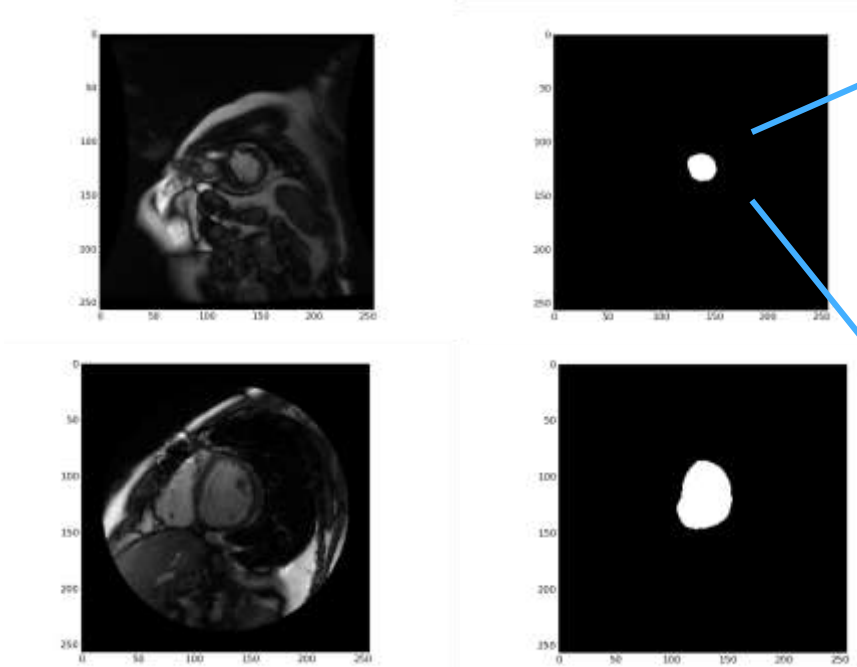
ezDICOM





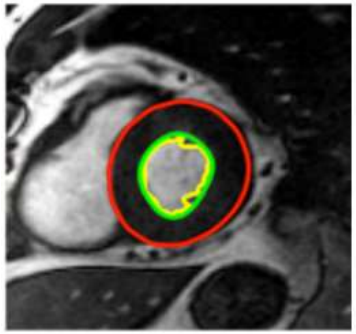
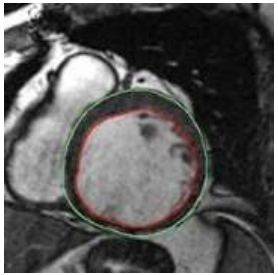
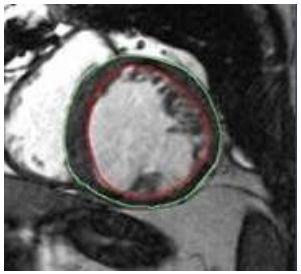
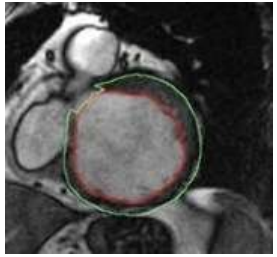
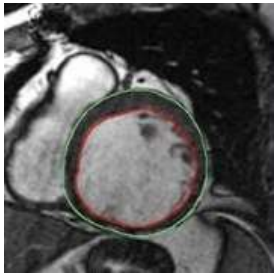
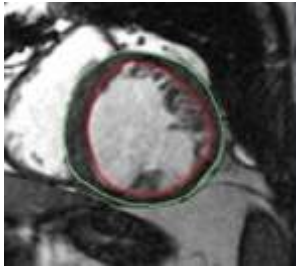
https://sourceforge.net/projects/ezdicom

DATASET WITH CONTOUR



X	Y
136.00	120.00
136.50	120.00
137.00	120.00
137.50	120.50
138.00	120.50
138.50	121.00
139.00	121.50
139.50	121.50
140.00	121.50
140.50	122.00
140.50	122.50
141.00	123.00
141.50	123.50
142.00	124.00
142.50	124.50
142.50	125.00
143.00	125.50
143.50	126.00
143.50	126.50
144.00	127.00
144.50	127.50
144.50	128.00
145.00	128.50
.	.
.	.
.	.

DATASET WITH CONTOUR

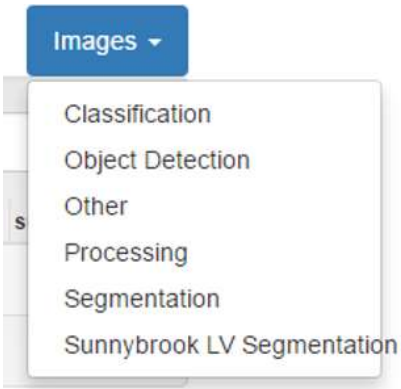







LAB

DIGITS PLUGINS

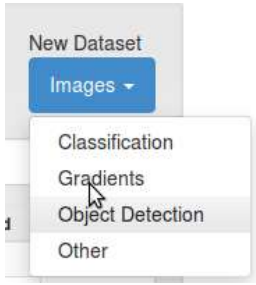
DIGITS Plugins
Image : Sunnybrook LV Segmentation

plugins/data/sunnybrook



DIGITS Plugins
Image : Regression

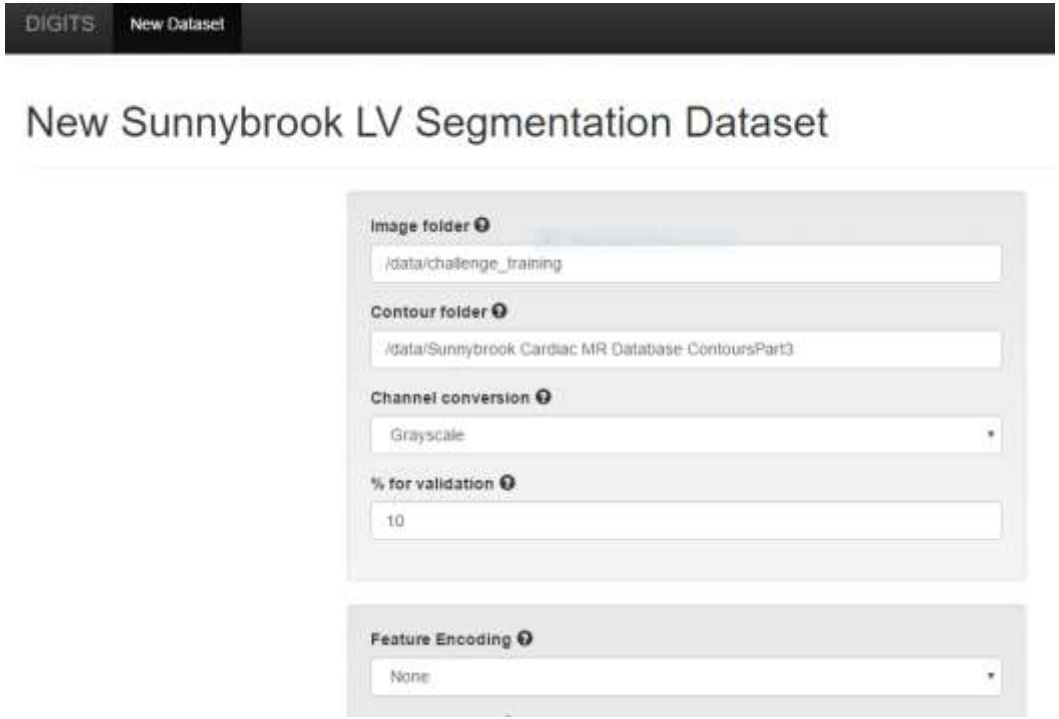
plugins/data/imageGradients

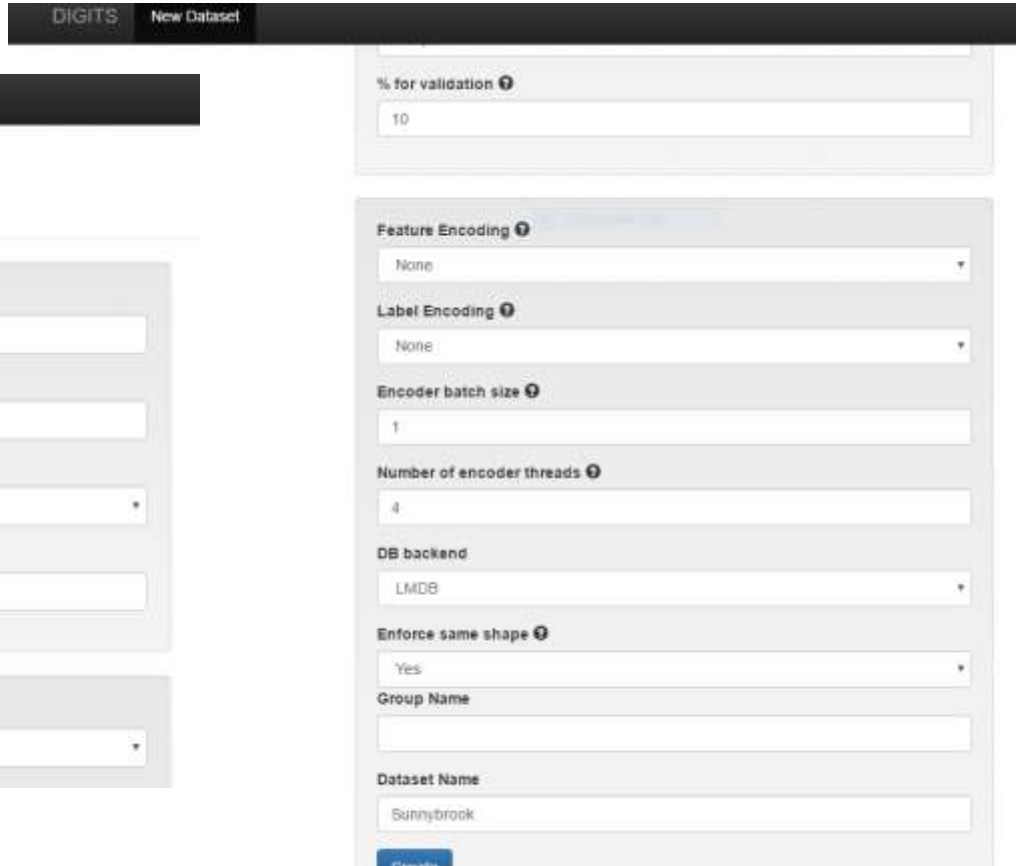


DIGITS Plugins
Text

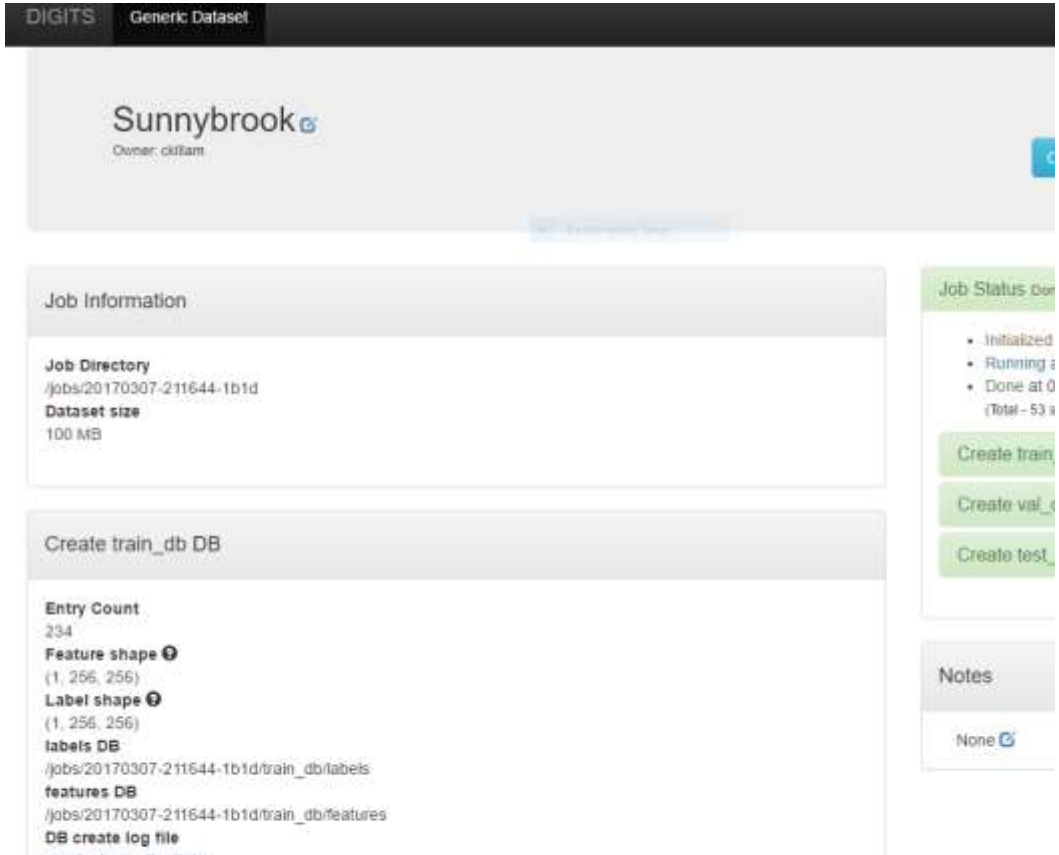
plugins/data/textClassification

PREPARE DATA

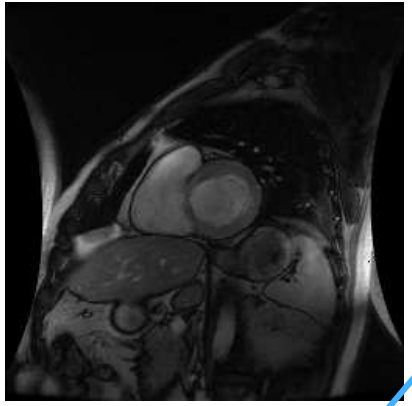




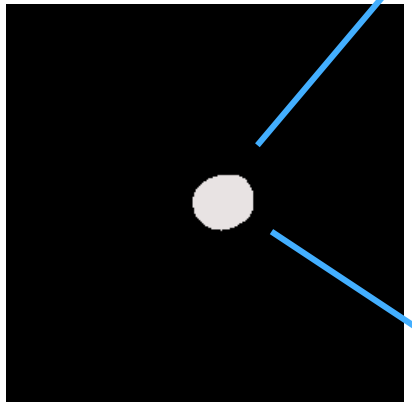
DATASET



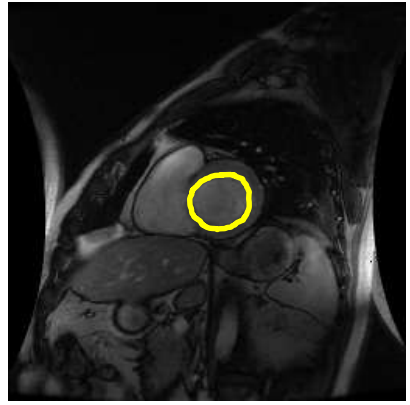
Image



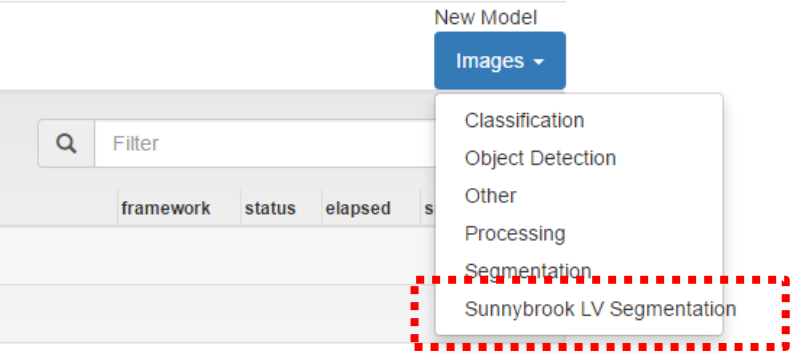
cantour

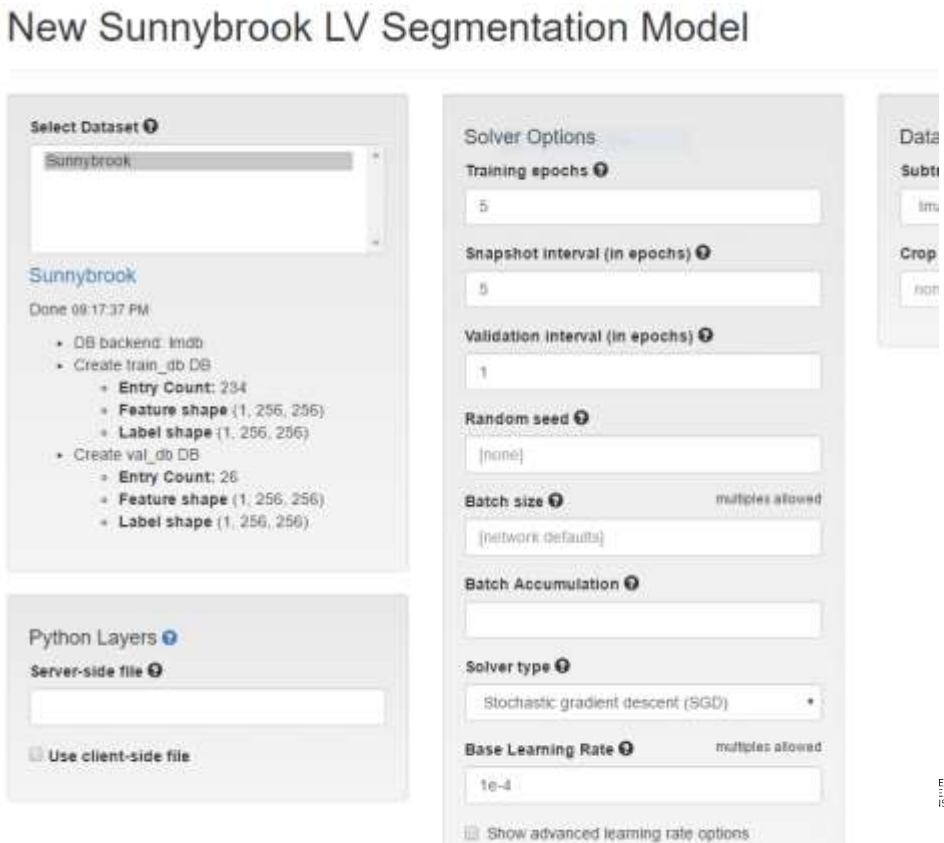


X	Y
130.50	118.00
131.00	118.00
131.50	117.50
132.00	117.50
132.50	117.50
133.00	117.50
133.50	117.50
134.00	117.50
134.50	117.50
135.00	117.50
135.50	117.50
136.00	117.50
136.50	117.50
137.00	117.50
137.50	118.00
138.00	118.00
138.50	118.50
139.00	118.50
139.50	119.00
140.00	119.00
140.50	119.50
141.00	119.50
141.50	120.00
142.00	120.00
142.50	120.50
143.00	121.00
143.50	121.50
144.00	121.50
144.50	122.00
145.00	122.50
.	.
.	.
.	.

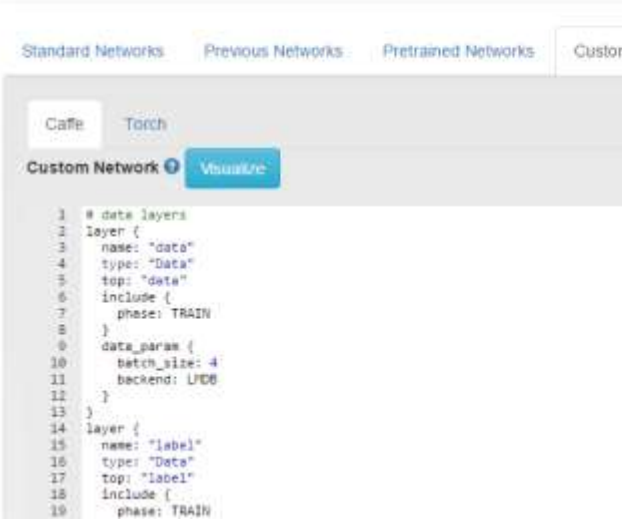


CONFIGURE DL MODEL

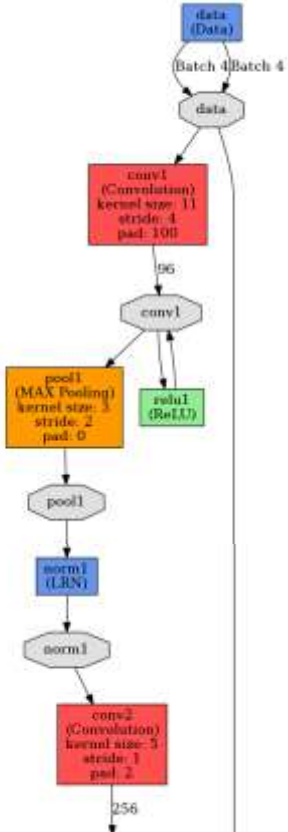


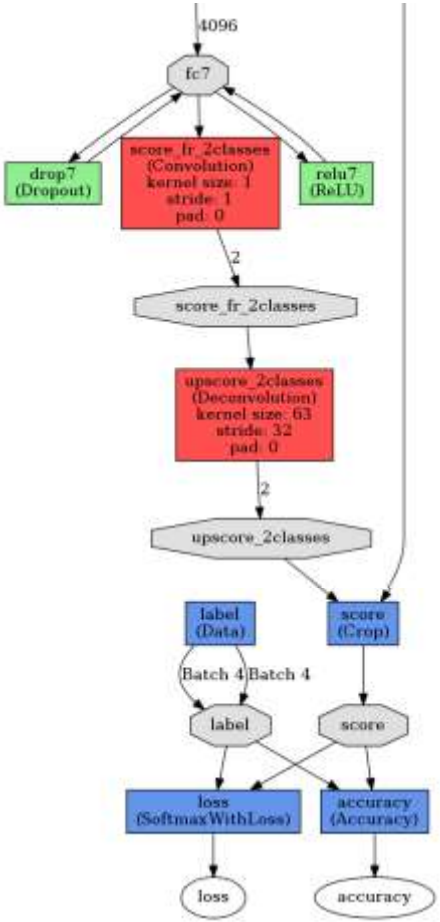


CONFIGURE DL MODEL

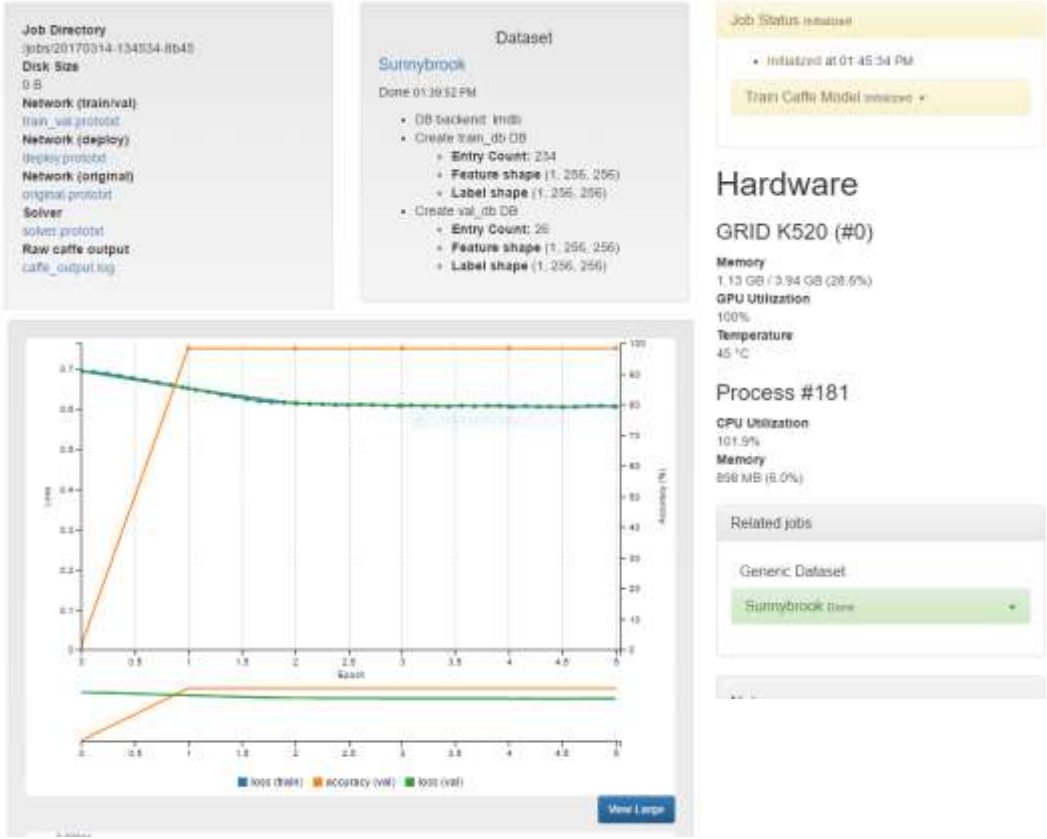


VISUALIZE

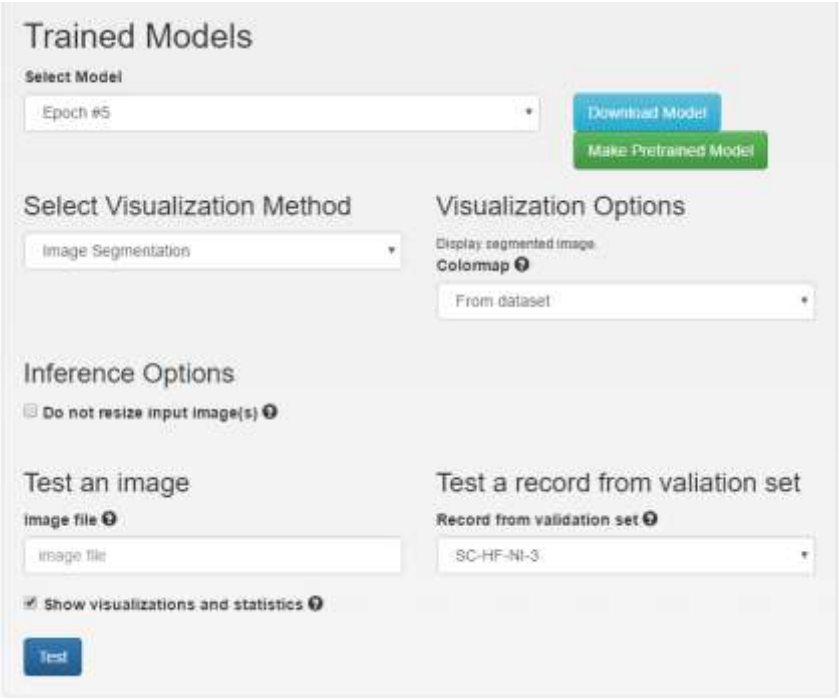


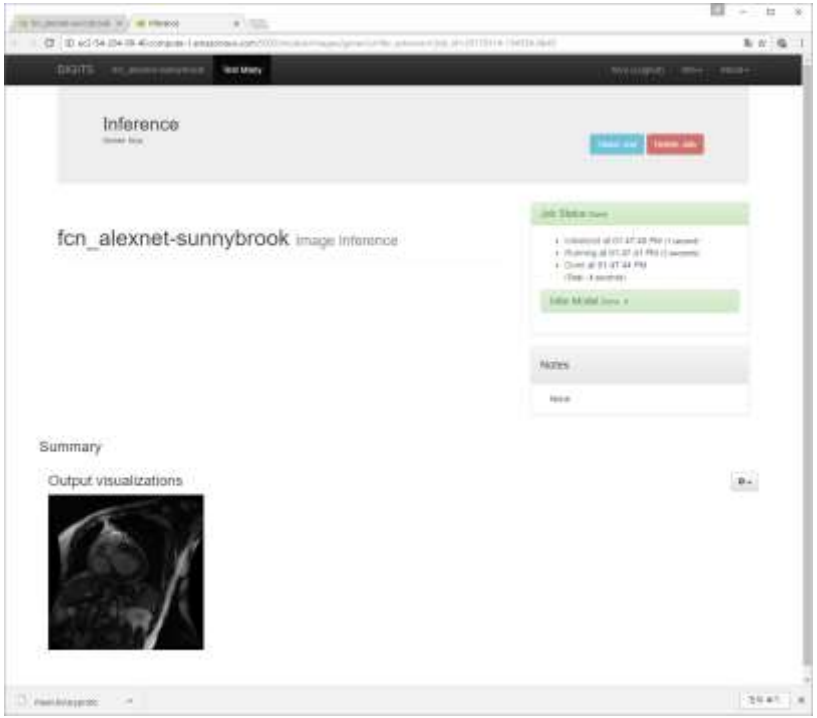


MONITOR TRAIN



TEST

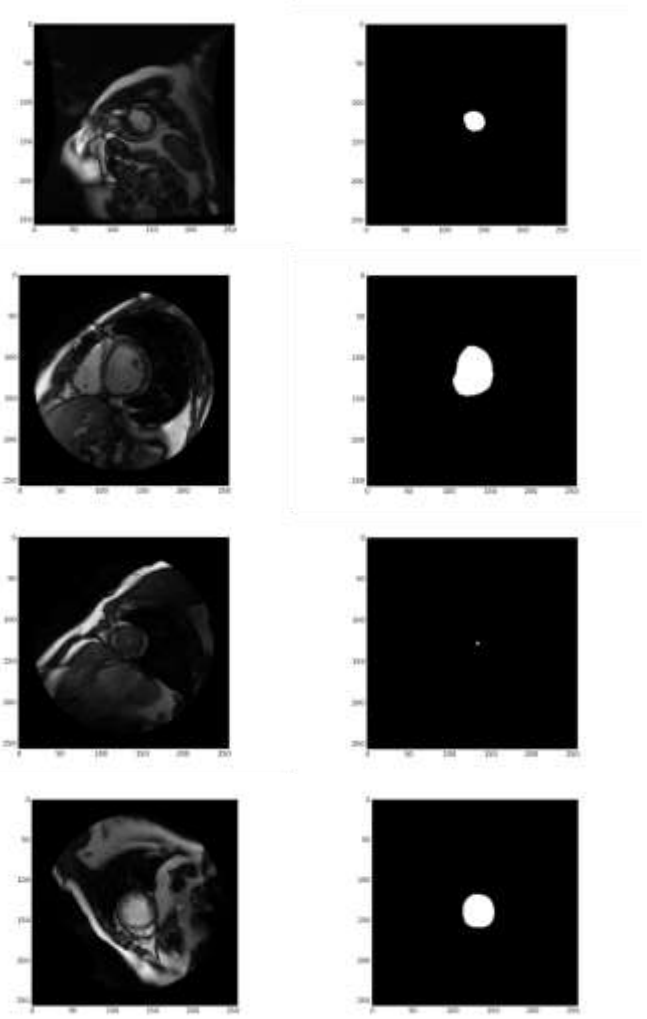

Trained Models
Select Model
Epoch #5
Download Model
Make Pretrained Model
Select Visualization Method
Image Segmentation
Visualization Options
Display segmented image
Colormap
From dataset
Inference Options
Do not resize input image(s)
Test an image
Image file
Image file
Test a record from valiation set
Record from validation set
SC-HF-NI-3
Show visualizations and statistics
Test


Inference
fcn_alexnet-sunnybrook Image Inference
Summary
Output visualizations

REASON

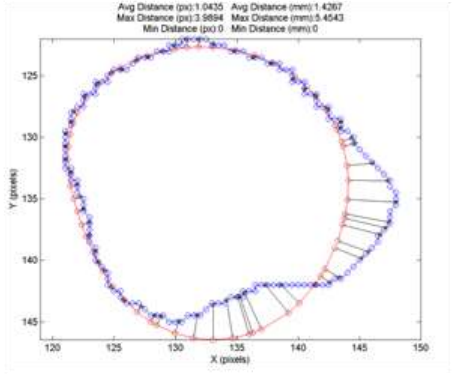
Cine MR

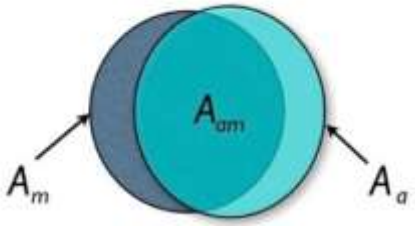
Same Object, time variance



PRACTICE 2
DICE METRIC

DICE METRIC





Manual contour (Expert)

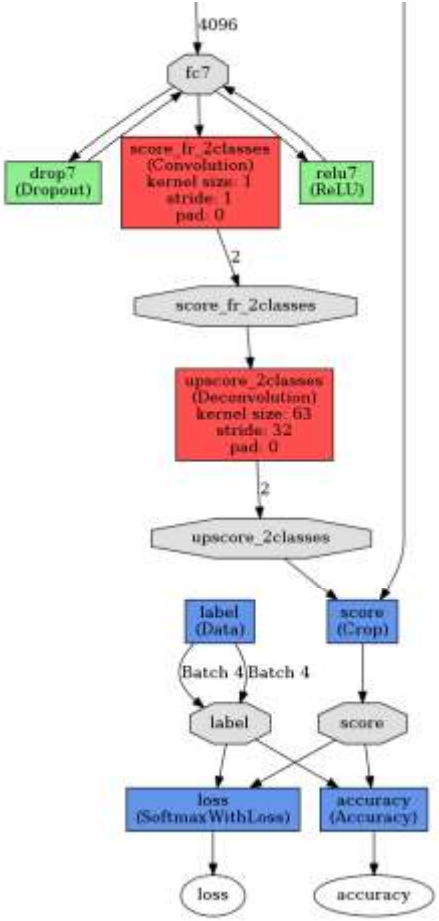
Automatic (DL)

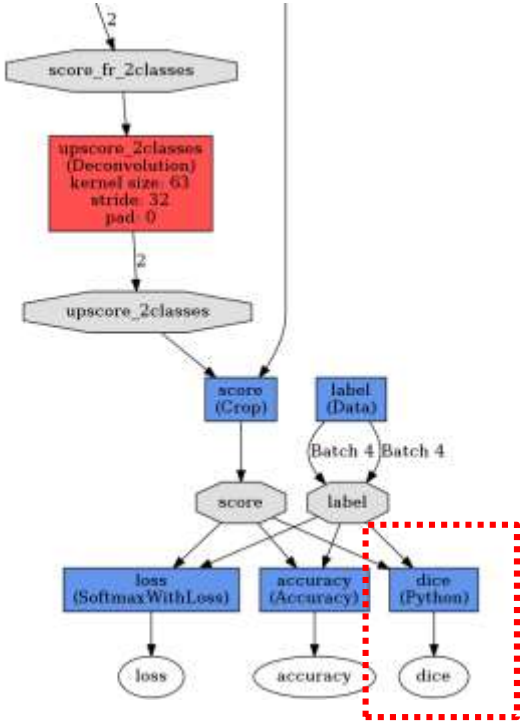
intersection

$$DM = 2A_{am}(A_a + A_m)^{-1}$$

Union

DICE METRIC





PYTHON LAYER

```
layer {
  name: "accuracy"
  type: "Accuracy"
  bottom: "score"
  bottom: "label"
  top: "accuracy"
  include { stage: "val" }
}


# Dice coefficient
layer {
    type: 'Python'
    name: 'dice'
    bottom: 'score'
    bottom: 'label'
    top: 'dice'
    python_param {
      module: "digits_python_layers"
      layer: "Dice"
    }
    exclude { stage: "deploy" }
}
```

Pretrained model(s)

```
import random

import numpy as np
import caffe


class Dice(caffe.Layer):
    """
    A layer that calculates the Dice coefficient
    """
    def setup(self, bottom, top):
        if len(bottom) != 2:
            raise Exception("Need two inputs to compute Dice coefficient.")

    def reshape(self, bottom, top):
        # check input dimensions match
        if bottom[0].count != 2*bottom[1].count:
            raise Exception("Prediction must have twice the number of elements of the input.")
        # loss output is scalar
        top[0].reshape(1)

    def forward(self, bottom, top):
        #print "bottom[0].shape=%s" % repr(bottom[0].data.shape)
        #print "bottom[1].shape=%s" % repr(bottom[1].data.shape)
        label = bottom[1].data[:,0,:,:]
        # compute prediction
        prediction = np.argmax(bottom[0].data, axis=1)
        #print "prediction.shape=%s" % repr(prediction.shape)
        # area of predicted contour
        a_p = np.count_nonzero(prediction)
        # area of contour in label
        a_l = np.count_nonzero(label)
        # area of intersection
        a_pl = np.count_nonzero(prediction * label)
        #print "a_p=%f a_l=%f a_pl=%f" % (a_p, a_l, a_pl)
        # dice coefficient
        dice_coeff = 2.*a_pl/(a_p + a_l)
        top[0].data[...] = dice_coeff

    def backward(self, top, propagate_down, bottom):
        pass
```

PYTHON LAYER

Label Shape (1, 256, 256)

Python Layers

Client-side file

Browse... layers.py

Use client-side file

[network defaults]

Batch Accumulation

Solver type

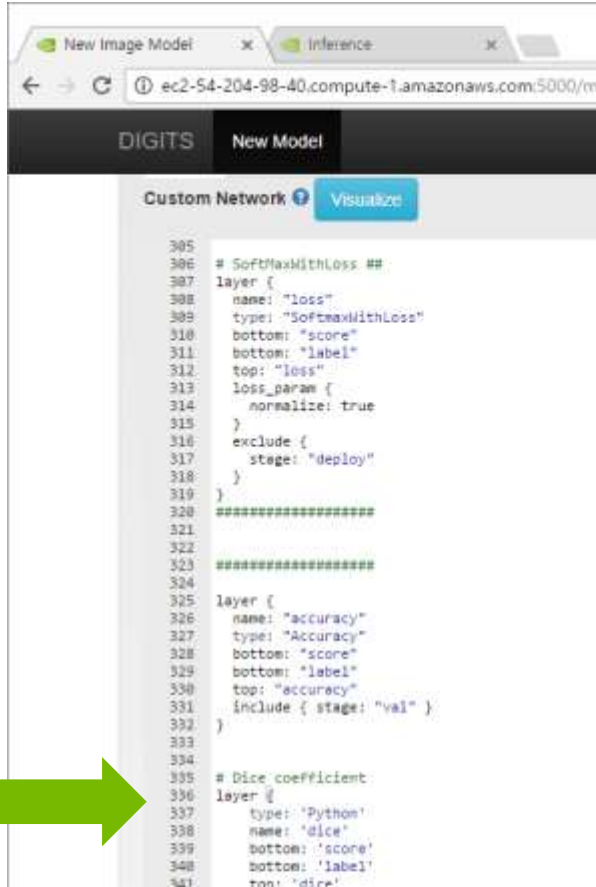
Stochastic gradient descent (SGD)

Base Learning Rate multiples allowed

0.0001

Show advanced learning rate options

Type : Python



ENABLE DICE LAYER

```
layer {
  name: "accuracy"
  type: "Accuracy"
  bottom: "score"
  bottom: "label"
  top: "accuracy"
  include { stage: "val" }
}

# Dice coefficient
#layer {
#    type: 'Python'
#    name: 'dice'
#    bottom: 'score'
#    bottom: 'label'
#    top: 'dice'
#    python_param {
#      module: "digits_python_layers"
#      layer: "Dice"
#    }
#    exclude { stage: "deploy" }
#}
```

```
layer {
  name: "accuracy"
  type: "Accuracy"
  bottom: "score"
  bottom: "label"
  top: "accuracy"
  include { stage: "val" }
}

# Dice coefficient
layer {
    type: 'Python'
    name: 'dice'
    bottom: 'score'
    bottom: 'label'
    top: 'dice'
    python_param {
      module: "digits_python_layers"
      layer: "Dice"
    }
    exclude { stage: "deploy" }
}
```

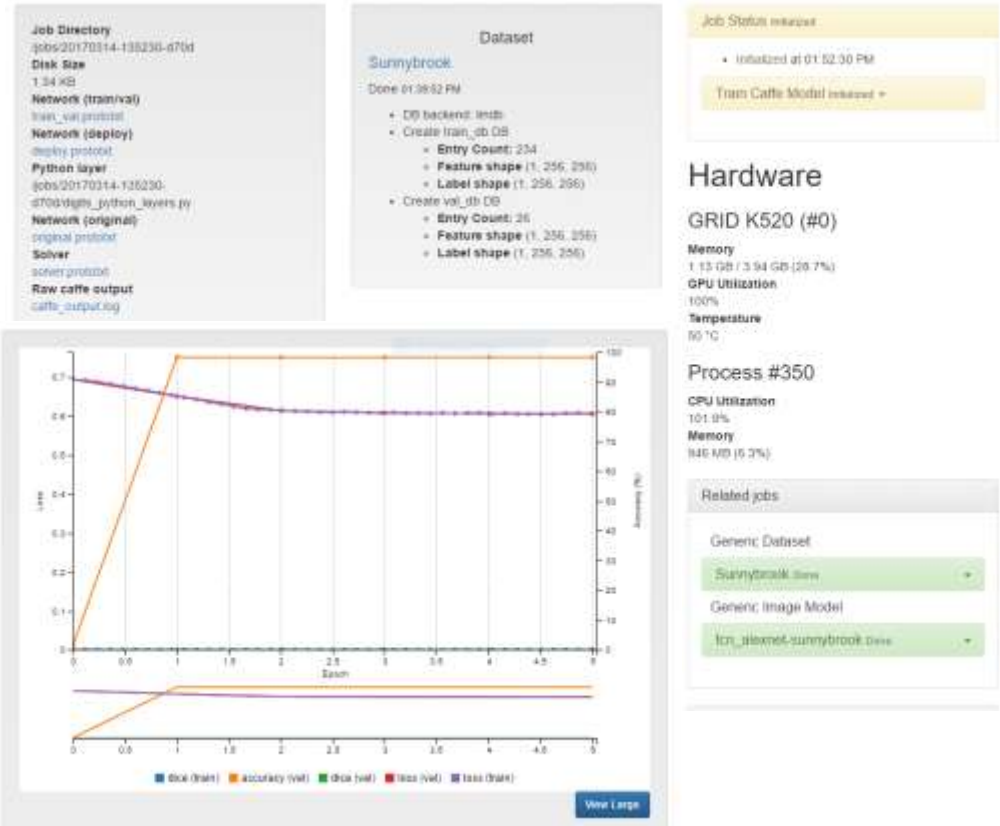
Group Name

Model Name

fc_alexnet-sunnybrook-with_dice

Create

DL TRAINING WITH DICE



RESULT2 WITH DICE

fcn_alexnet-sunnybrook-with_dice Image Inference

Job Status Done

- Initialized at 01:55:00 PM (1 second)
- Running at 01:55:01 PM (3 seconds)
- Done at 01:55:04 PM

(Total - 4 seconds)

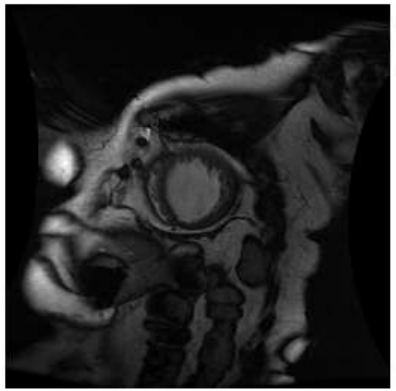
Infer Model Done

Notes

None

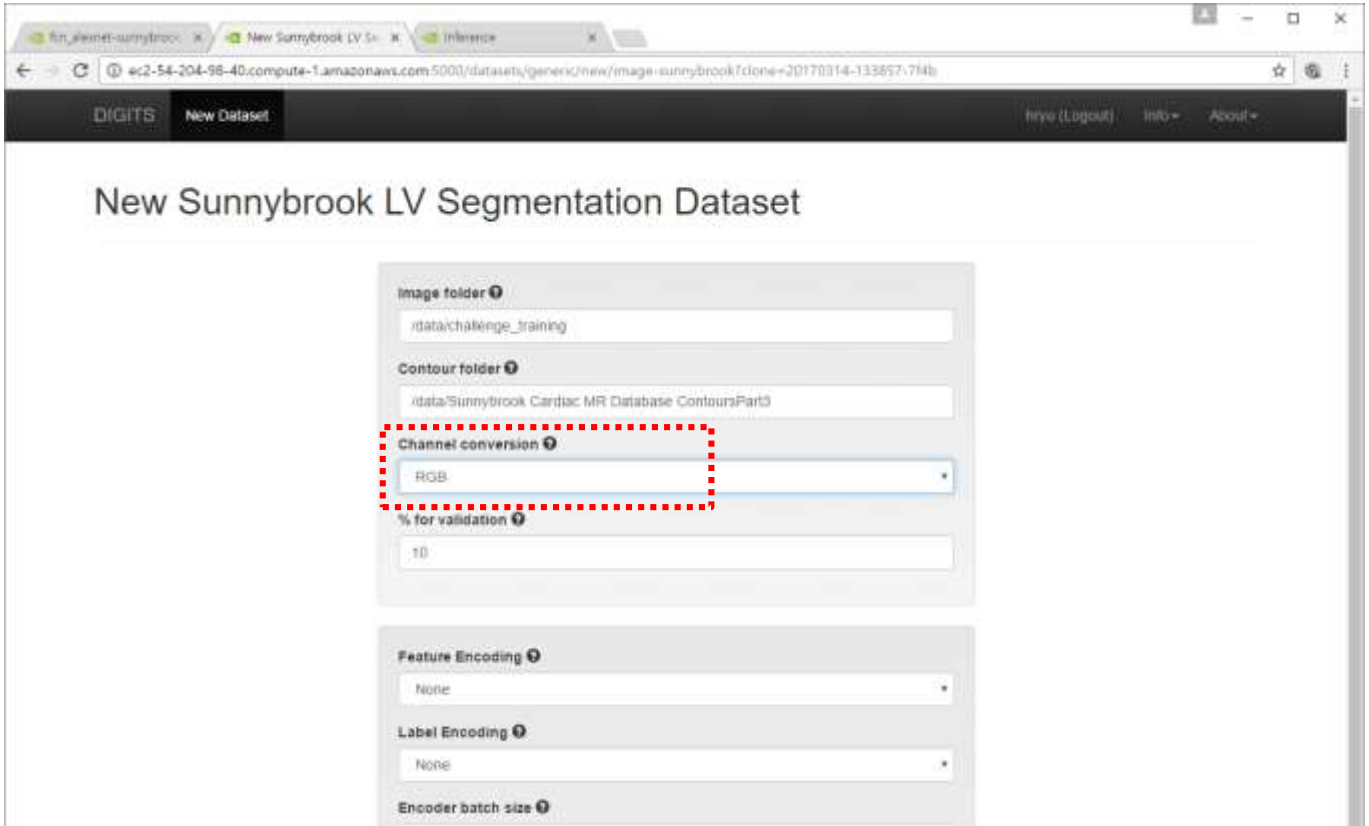
Summary

Output visualizations

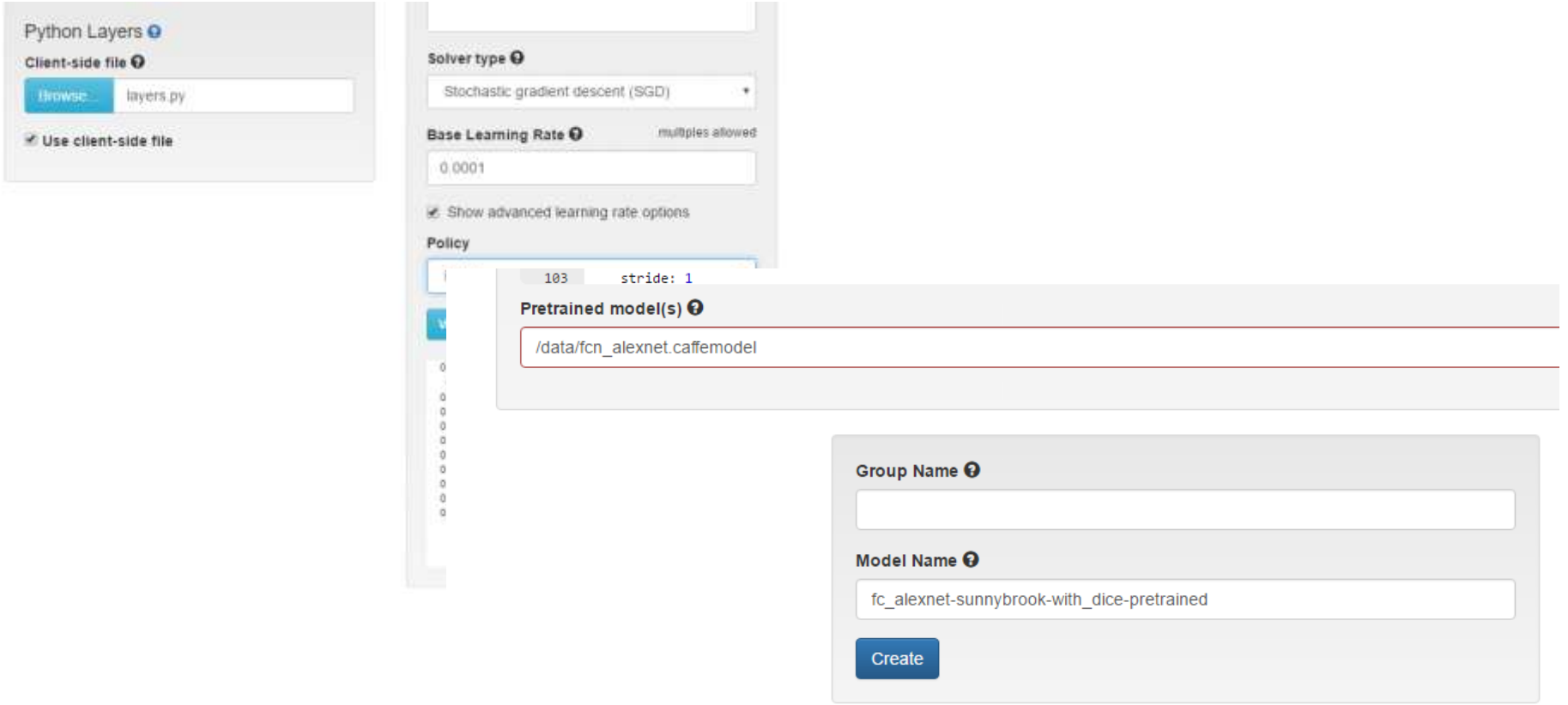


PRACTICE 3
USE PRE-TRAINED PARAMETERS

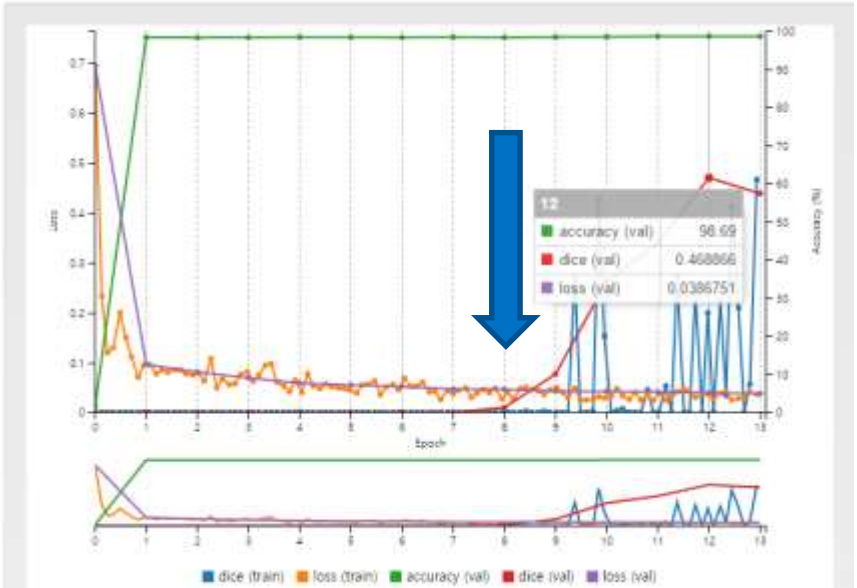
FOR ALEXNET (COLOR)

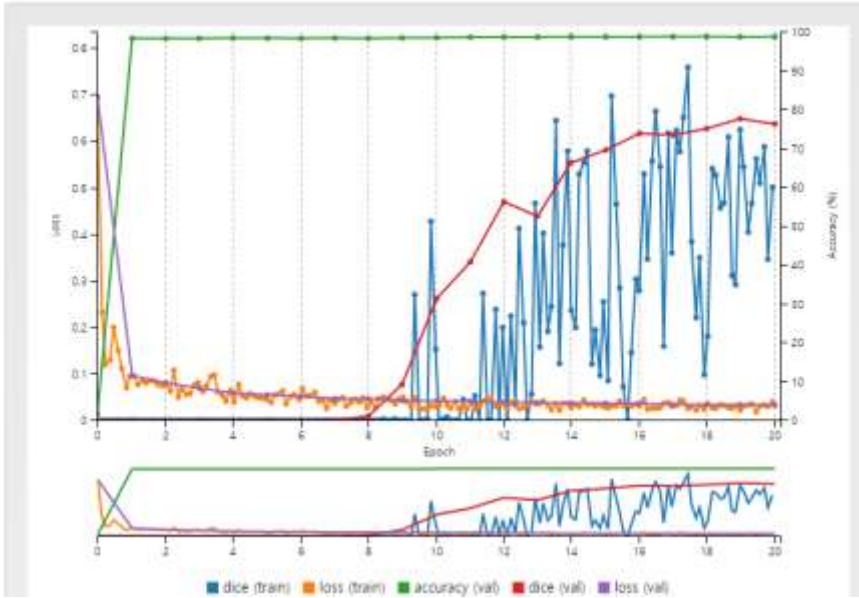


CONFIGURE DL MODEL WITH PRE-TRAINED



DICE(VAL)





RESULT WITH PRE-TRAIN

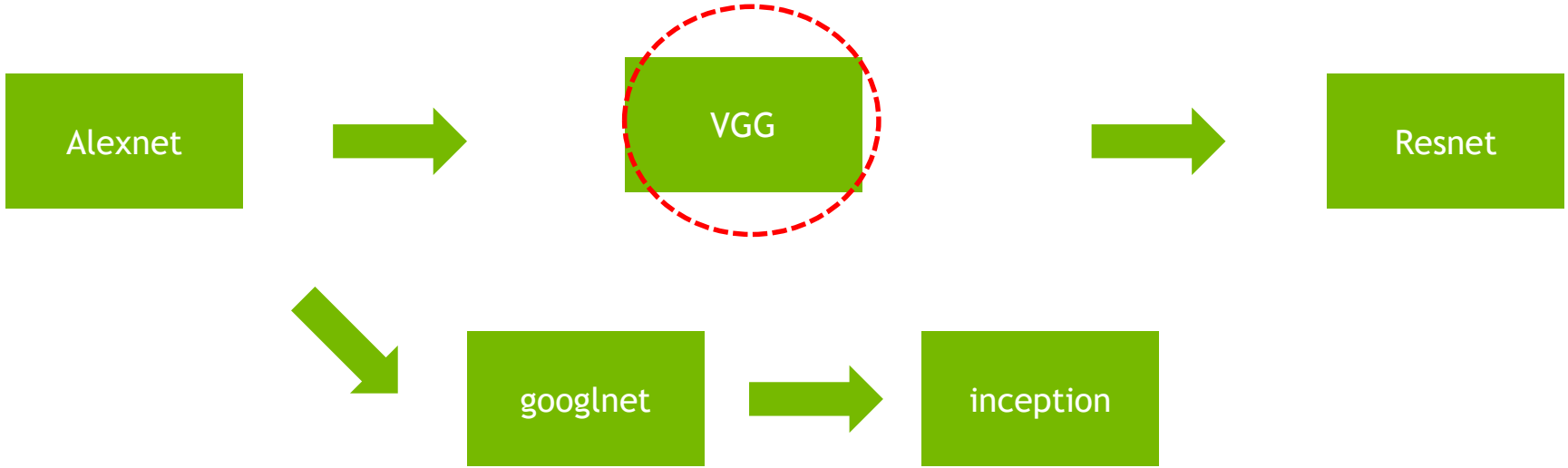
Summary

Output visualizations

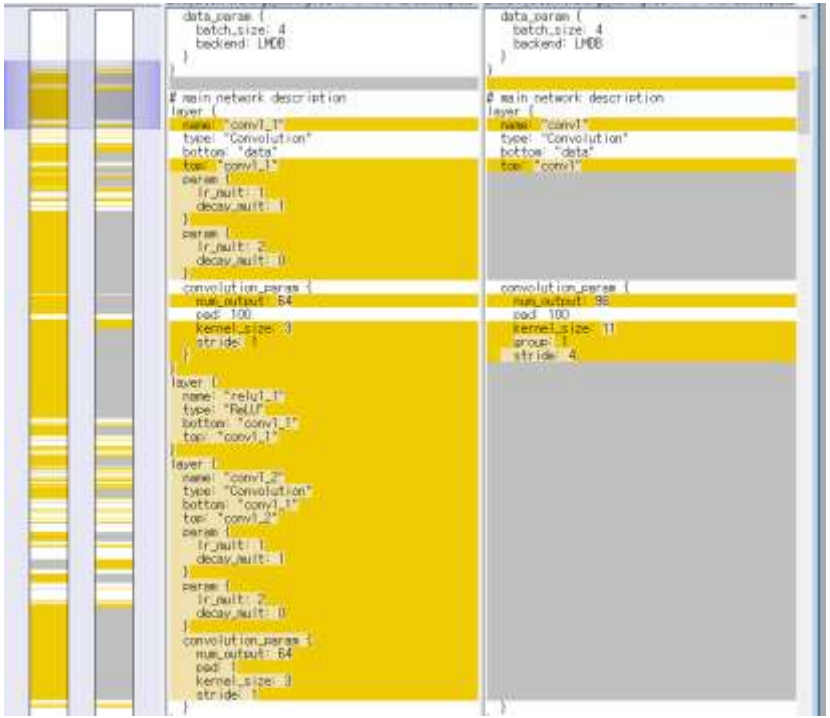
☐ left ventricle

PRACTICE 4
MORE FINE DL MODEL

FCN-8S



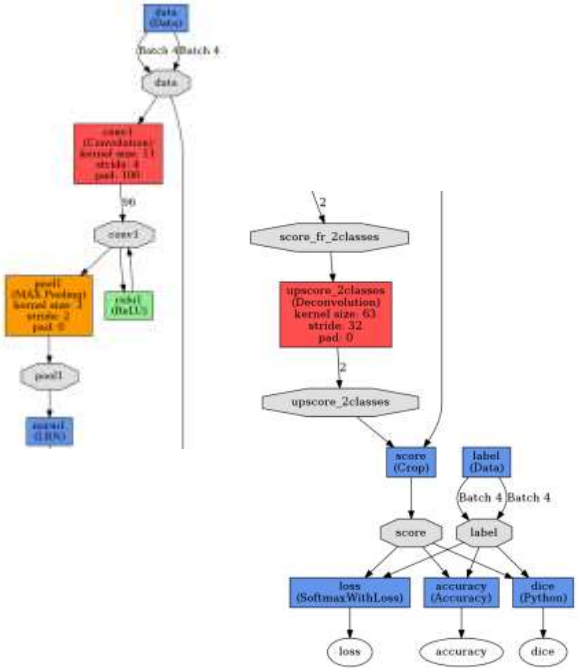
DIFF



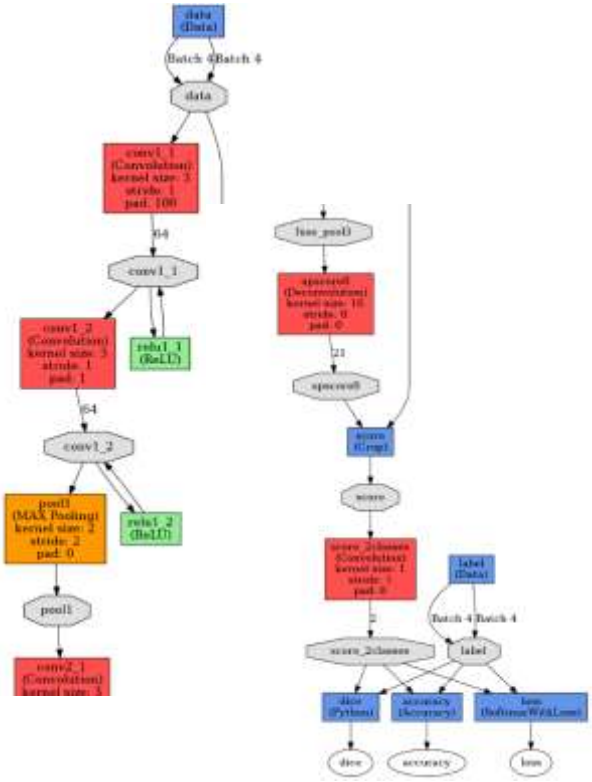
VGG style

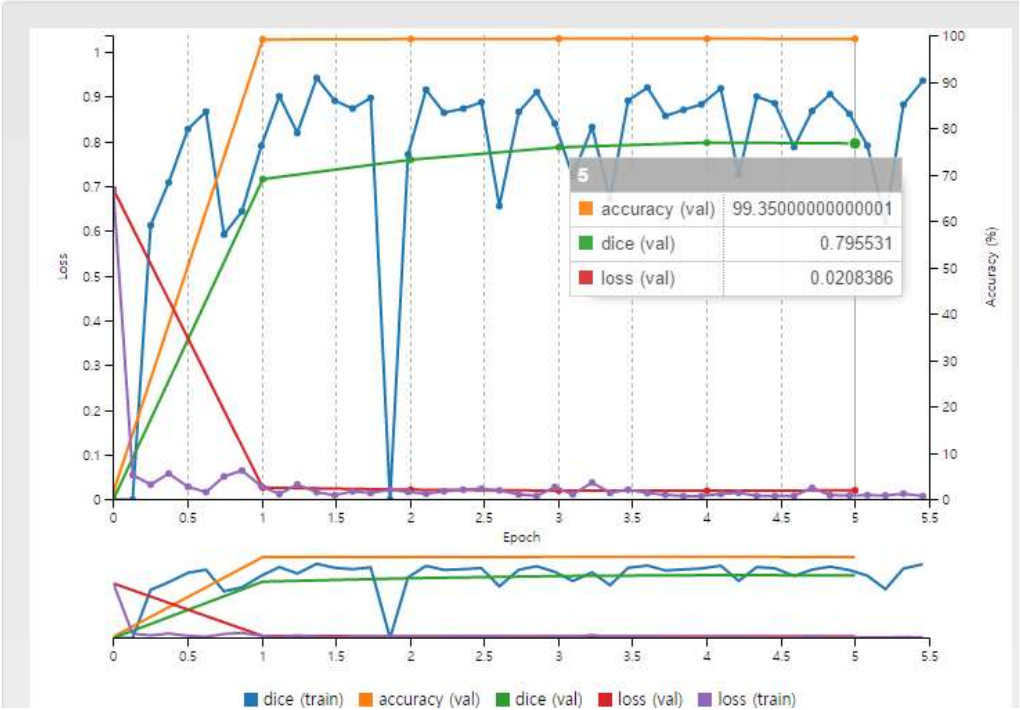
Alexnet

Alxenet



VGG style





RESULT

Summary

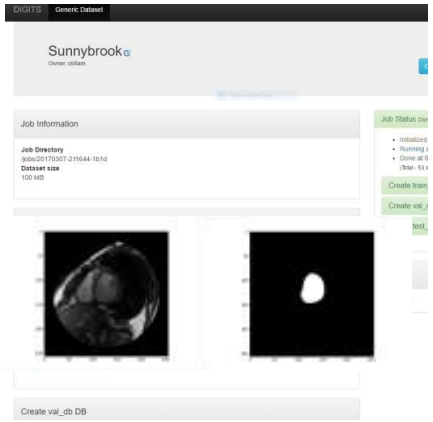
Output visualizations

☐ left ventricle

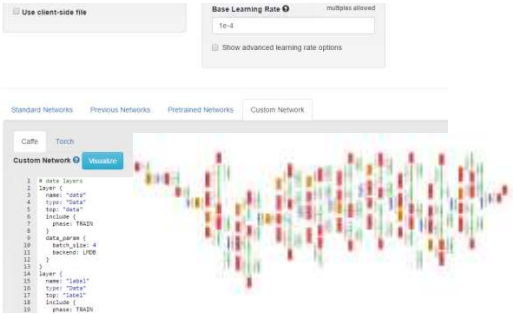
MEDICAL IMAGE SEGMENTATION WITH DIGITS

summary

Prepare Dataset

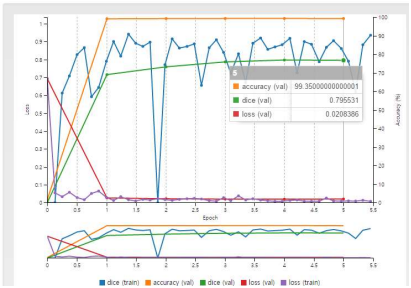


Configure DL Model



With python layer(DICE metric)

DL Training



Segmentation

WHAT'S NEXT

TAKE SURVEY

ACCESS ONLINE LABS

Check your email to access more DLI training online.

ATTEND WORKSHOP

Visit www.nvidia.com/dli for workshops in your area.

JOIN DEVELOPER PROGRAM

Visit https://developer.nvidia.com/join for more.

DEEP
LEARNING
INSTITUTE

www.nvidia.com/dli