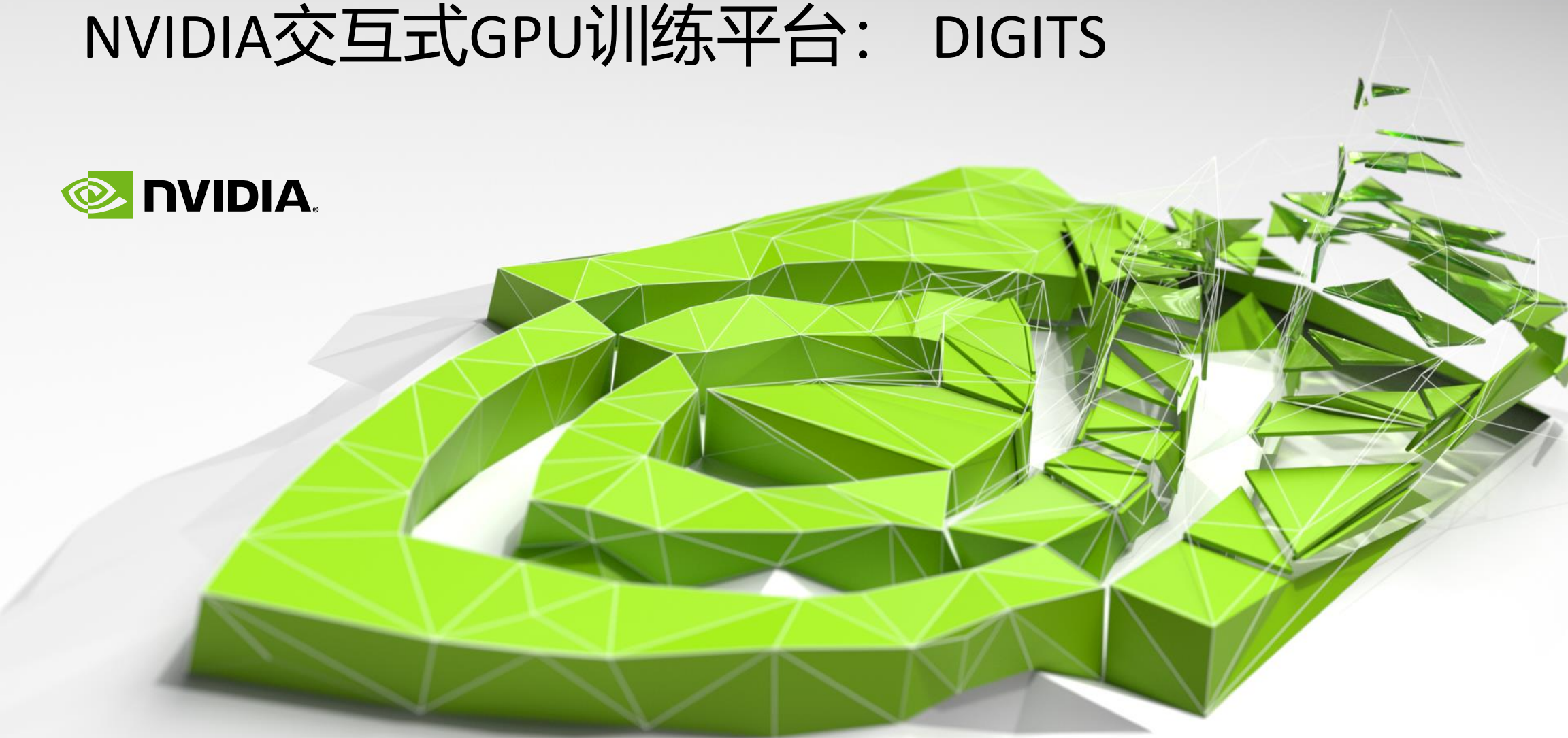


NVIDIA交互式GPU训练平台： DIGITS



AGENDA

开源免费的深度学习管理工具

1. DIGITS诞生背景及功能简介
2. DIGITS 的训练数据集导入
3. DIGITS 的基本模型训练操作：图片分类
4. DIGITS 用于目标检测
5. DIGITS 用于图像分割
6. DIGITS 迁移学习的使用
7. DIGITS Plugin的使用

传统的深度学习开发界面

- DL frameworks, Caffe, etc. aimed at computer scientist not data scientist
- Juggle multiple files & windows
- Handcrafted visualizations
- Manual log file parsing
- Manual experiment logging
- Model editing in Lua IDE files

```
111 -- stage 1 : filter bank -> squashing -> L2 pooling -> normalization
112 model:add(nn.SpatialConvolutionMM(nfeats, nstates[1], filter_size, filter_size))
113 model:add(nn.Tanh())
114 model:add(nn.SpatialLPPooling(nstates[1], 2, pool_size, pool_size, pool_size, pool_size))
115 model:add(nn.SpatialSubtractiveNormalization(nstates[1], normkernel))
116
117 -- stage 2 : filter bank -> squashing -> L2 pooling -> normalization
118 model:add(nn.SpatialConvolutionMM(nstates[1], nstates[2], filter_size, filter_size))
119 model:add(nn.Tanh())
120 model:add(nn.SpatialLPPooling(nstates[2], 2, pool_size, pool_size, pool_size, pool_size))
121 model:add(nn.SpatialSubtractiveNormalization(nstates[2], normkernel))
122
123 -- stage 3 : standard 2-layer neural network
124 model:add(nn.Reshape(nstates[2], filter_size, filter_size))
125 model:add(nn.Linear(nstates[2], filter_size, nstates[3]))
126 model:add(nn.Tanh())
127 model:add(nn.Linear(nstates[3], noutputs))
128
129 else
130
131     error('unknown -model')
132
133 end
134
135
136 print '==> here is the model:'
137 print(model)
```

```
--> here is the model:
nn.Sequential {
  [input -> (1) -> (2) -> (3) -> (4) -> (5) -> (6) -> (7) -> (8) -> (9) -> (10) -> (11) -> (12)]
  (1): nn.SpatialConvolutionMM(3 -> 64, 5x5)
  (2): nn.Tanh
  (3): nn.Sequential {
    [input -> (1) -> (2) -> (3) -> (4) -> output]
    (1): nn.Square
    (2): nn.SpatialAveragePooling(2,2,2,2)
    (3): nn.MulConstant
    (4): nn.Sqrt
  }
  (4): nn.SpatialSubtractiveNormalization
  (5): nn.SpatialConvolutionMM(64 -> 64, 5x5)
  (6): nn.Tanh
  (7): nn.Sequential {
    [input -> (1) -> (2) -> (3) -> (4) -> output]
    (1): nn.Square
    (2): nn.SpatialAveragePooling(2,2,2,2)
    (3): nn.MulConstant
    (4): nn.Sqrt
  }
  (8): nn.SpatialSubtractiveNormalization
  (9): nn.Reshape(1600)
  (10): nn.Linear(1600 -> 128)
  (11): nn.Tanh
  (12): nn.Linear(128 -> 10)
}
```

```
306 #28p a02 Take the trueNLL model w/ kappa_true=0.76 from expt#28 -> true + MSE under settings
307 d all data, 55 epochs, epochSize 500, regime with LR=0.2
308 th main_custom_noTest.lua -cache devAnnealAugFinal -data ./data-1024-fulltrain -nGPU 4 -name TrueNLL
nDonkeys 6 -nEpochs 55 -epochSize 500 -batchSize 32 -LR 0.2 -netType 1024/nin-preall-multi -retrain /
multiGPU/highres/learn_1024_anneal/devAnnealAug/EvenNLLtoTrueNLL_fixedLR_Crfinal=trueCR_LR=0.1_batchS
nGPU=4,netType=1024/nin-preall-multi-fixedLR/05-48_on_25-Jul-15/model_100.t7 -Crfinal trueCR -Crfinal
1 -manualSeed 4359 | tee expt28p_ap02_TrueNLLtoTrueMSEFinal.log
309
310
311 #29p a03 Take the 62411+0.75MSE model w/ kappa_true=0.78 from expt#24 -> true + MSE under set
312 b regular data, 55 epochs, epochSize 200, regime with LR=0.2
313 th main_custom.lua -cache devAnnealAug -data ./data-1024 -nGPU 4 -name CR62411MSE3by4_seed864521_exp
epochSize 200 -batchSize 32 -LR 0.2 -netType 1024/nin-preall-multi -retrain /home/mnt/retinopathy/ins
learn_1024_anneal/devAnnealAug/pretrained62411MSE3by4_trainFurther_Crfinal=6,2,4,1,1 -Crfinal=6,2,4,1,1
MSEInitial=0.75, batchSize=32, epochSize=300, nDonkeys=6, nGPU=4, netType=1024/nin-preall-multi-fixedLR/05
Crfinal 6,2,4,1,1 -Crfinal trueCR -MSEInitial 0 -MSEFinal 1 -manualSeed 864521 | tee expt29p_a03_CR6
314
315
316 #30p ap06 Take the 62411+0.75MSE model w/ kappa_true=0.78 from expt#24 -> true + MSE under
317 d all data, 55 epochs, epochSize 500, regime with LR=0.2
318 th main_custom_noTest.lua -cache devAnnealAugFinal -data ./data-1024-fulltrain -nGPU 4 -name CR62411
nDonkeys 6 -nEpochs 55 -epochSize 500 -batchSize 32 -LR 0.2 -netType 1024/nin-preall-multi -retrain /
multiGPU/highres/learn_1024_anneal/devAnnealAug/pretrained62411MSE3by4_trainFurther_Crfinal=6,2,4,1,1
MSEInitial=0.75, batchSize=32, epochSize=300, nDonkeys=6, nGPU=4, netType=1024/nin-preall-multi-fixedLR/05
model_55.t7 -Crfinal 6,2,4,1,1 -Crfinal trueCR -MSEInitial 0 -MSEFinal 1 -manualSeed 351987 | tee e
log
319
```

NVIDIA'S DIGITS

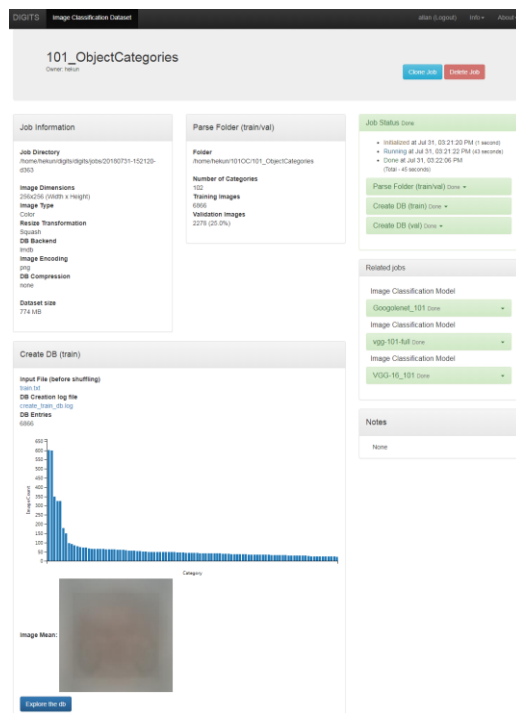
交互式深度学习GPU训练系统

- 简化通用的深度学习任务如：
 - 管理数据
 - 在多GPU系统上设计并训练神经网络
 - 使用高级可视化界面，监控实时性能
- 完全的交互式界面，使得数据科学家可以专注在设计及训练网络，节约编程及调试代码的时间
- 开源、免费

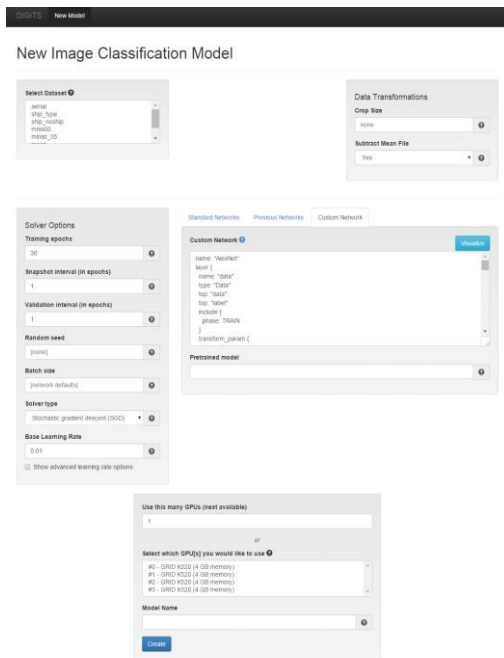
NVIDIA DIGITS

交互式深度学习GPU训练平台

数据加载



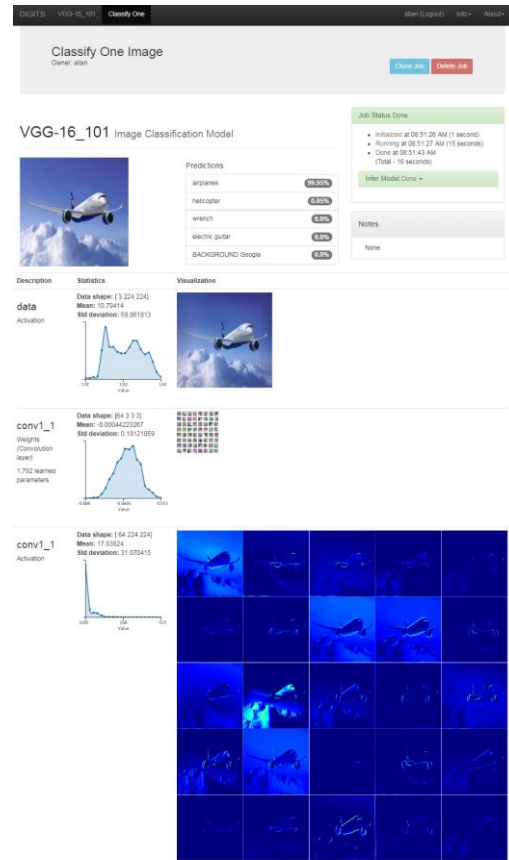
配置神经网络



监控训练过程



校验训练精度



DIGITS - HOME

Clicking DIGITS will bring you to this Home screen

The screenshot shows the DIGITS Home interface. At the top is a dark navigation bar with the 'DIGITS' logo on the left and 'ckillam (Logout)', 'Info', and 'About' on the right. Below the navigation bar, the word 'Home' is displayed. A status indicator on the right says '1/1 GPU available'. The main content area shows 'No Jobs Running' and three tabs: 'Datasets (0)', 'Models (0)', and 'Pretrained Models (0)'. Below these tabs is a section for 'Group Jobs' with a checked checkbox, 'Delete' and 'Group' buttons, and a table with columns 'name', 'framework', 'status', 'elapsed', and 'submitted'. The table currently shows 'No Models'. On the right side of the main content area, there is a 'New Model' button with a dropdown menu showing 'Images'. Green arrows point from text annotations to the 'DIGITS' logo, the 'Datasets (0)' and 'Models (0)' tabs, and the 'New Model' button.

Click here to see a list of existing datasets or models

Clicking here will present different options for model and dataset creation

DIGITS - DATASET

New Image Classification Dataset

Image Type ?

Color

Image size (Width x Height) ?

256 x 256

Resize Transformation ?

Squash

See example

Use Image Folder Use Text Files

Use Rectangular Snips

Training Images ?

folder or URL

Minimum samples per class ?

2

Maximum samples per class ?

% for validation ?

25

% for testing ?

0

☐ Separate validation images folder

☐ Separate test images folder

DB backend

LMDB

Image Encoding ?

PNG (lossless)

Group Name

Dataset Name

Create

Different options will be presented based upon the task

DIGITS - MODEL

Define custom layers
with Python

New Image Classification Model

Select Dataset ⓘ

Python Layers ⓘ

Server-side file ⓘ

☐ Use client-side file

Solver Options

Training epochs ⓘ

30

Snapshot interval (in epochs) ⓘ

1

Validation interval (in epochs) ⓘ

1

Random seed ⓘ

[none]

Batch size ⓘ multiples allowed

[network defaults]

Batch Accumulation ⓘ

Solver type ⓘ

Stochastic gradient descent (SGD)

Base Learning Rate ⓘ multiples allowed

0.01

☐ Show advanced learning rate options

Data Transformations

Subtract Mean ⓘ

Image

Crop Size ⓘ

none

Can anneal the
learning rate

Standard Networks Previous Networks Pretrained Networks Custom Network		
Caffe Torch		
Network	Details	Intended image size
<input checked="" type="radio"/> LeNet	Original paper [1998]	28x28 (gray)

Differences may exist between model tasks

DIGITS- 模型编程入口及可视化

DIGITS New Model

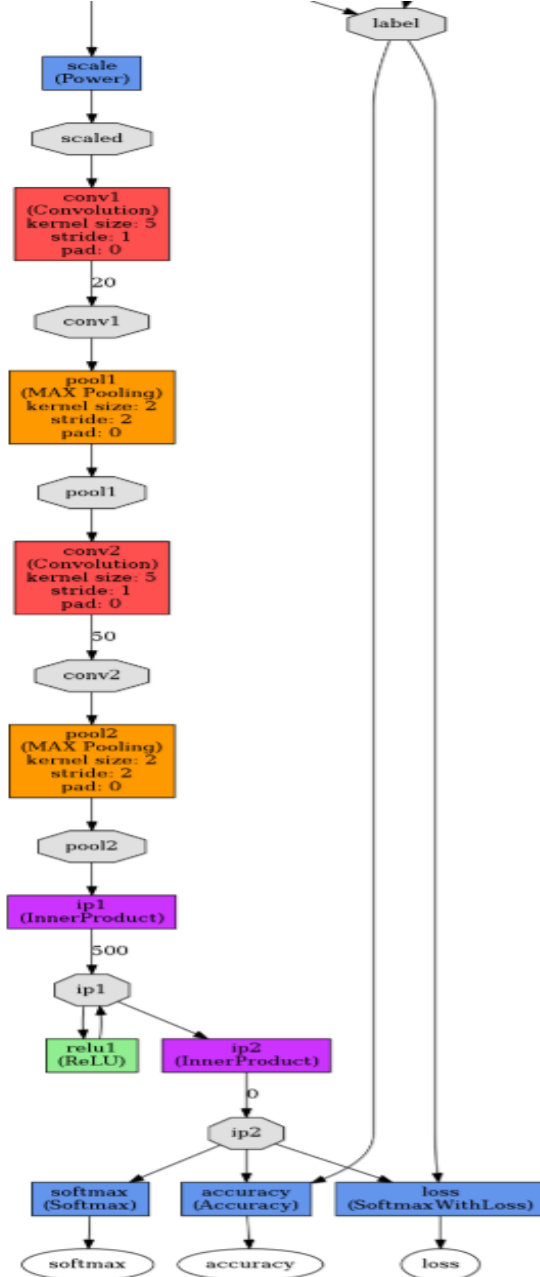
Standard Networks Previous Networks Pretrained Networks Custom Network

Caffe Tensorflow

Custom Network ? Visualize

```
27 top: "scaled"
28 type: "Power"
29 power_param {
30   # 1/(standard deviation on MNIST dataset)
31   scale: 0.0125
32 }
33 }
34 layer {
35   name: "conv1"
36   type: "Convolution"
37   bottom: "scaled"
38   top: "conv1"
39   param {
40     lr_mult: 1
41   }
42   param {
43     lr_mult: 2
44   }
45   convolution_param {
46     num_output: 20
47     kernel_size: 5
48     stride: 1
49     weight_filler {
50       type: "xavier"
51     }
52     bias_filler {
53       type: "constant"
54     }
55   }
56 }
57 layer {
58   name: "pool1"
59   type: "Pooling"
60   bottom: "conv1"
61   top: "pool1"
62   pooling_param {
63     pool: MAX
64     kernel_size: 2
65     stride: 2
66   }
67 }
68 layer {
69   name: "conv2"
```

Pretrained model(s) ?



MODIFY THE NETWORK

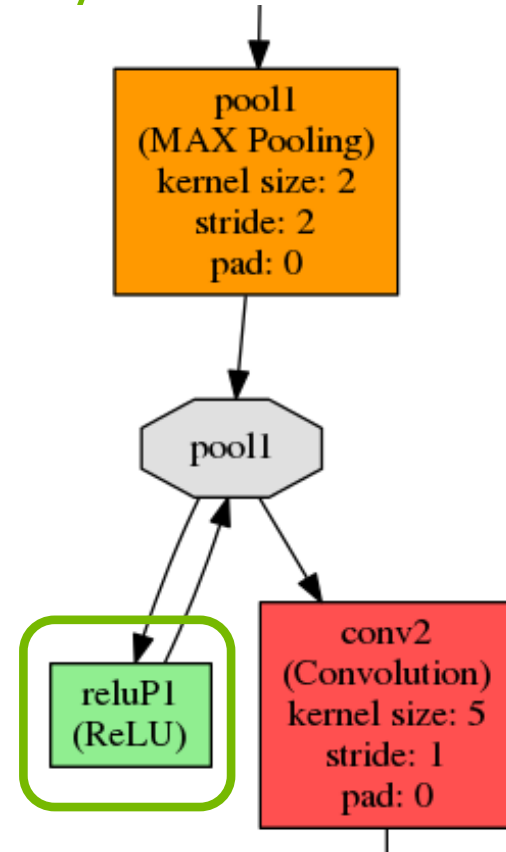
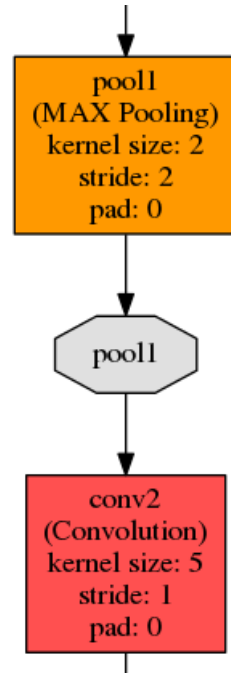
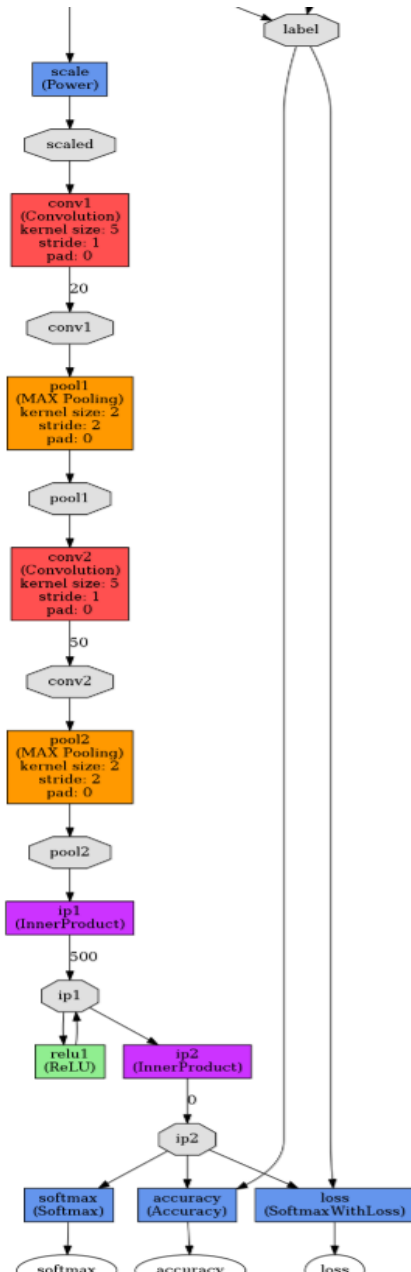
Adding filters and ReLU layer

```
layer {  
  name: "pool1"  
  type: "Pooling"  
  ...  
}  
  
layer {  
  name: "reluP1"  
  type: "ReLU"  
  bottom: "pool1"  
  top: "pool1"  
}  
  
layer {  
  name: "reluP1"
```

```
layer {  
  name: "conv1"  
  type: "Convolution"  
  ...  
  convolution_param {  
    num_output: 75  
    ...  
  }  
  layer {  
    name: "conv2"  
    type: "Convolution"  
    ...  
    convolution_param {  
      num_output: 100  
      ...  
    }  
  }  
}
```

MODIFY THE NETWORK

Adding ReLU Layer



DIGITS – 训练过程监控

使用的数据集信息

模型训练配置文件

vgg-101-full

Owner: hekun

Clone Job Abort Job Delete Job

Job Directory

/home/hekun/digits/digits/jobs/20180910-183417-94cb

Disk Size

0 B

Network (train/val)

train_val.prototxt

Network (deploy)

deploy.prototxt

Network (original)

original.prototxt

Solver

solver.prototxt

Raw caffe output

caffe_output.log

Pretrained Model

/home/hekun/digits/digits/jobs/20180726-123358-be04/model_caffemodel

Dataset

101_ObjectCategories

Done Jul 31, 03:22:06 PM

Image Size

256x256

Image Type

COLOR

DB backend

lmdb

Create DB (train)

6866 images

Create DB (val)

2278 images

Job Status Initialized

• Initialized at 06:34:17 PM

Train Caffe Model Initialized ▾

Hardware

Tesla P100-PCIE-16GB (#0)

Memory

8.28 GB / 15.9 GB (52.1%)

GPU Utilization

100%

Temperature

56 °C

Process #3407

CPU Utilization

129.9%

Memory

2.87 GB (4.9%)

Related jobs

Image Classification Dataset

101_ObjectCategories Done ▾

Image Classification Model

GoogLeNet_101 Done ▾

Image Classification Model

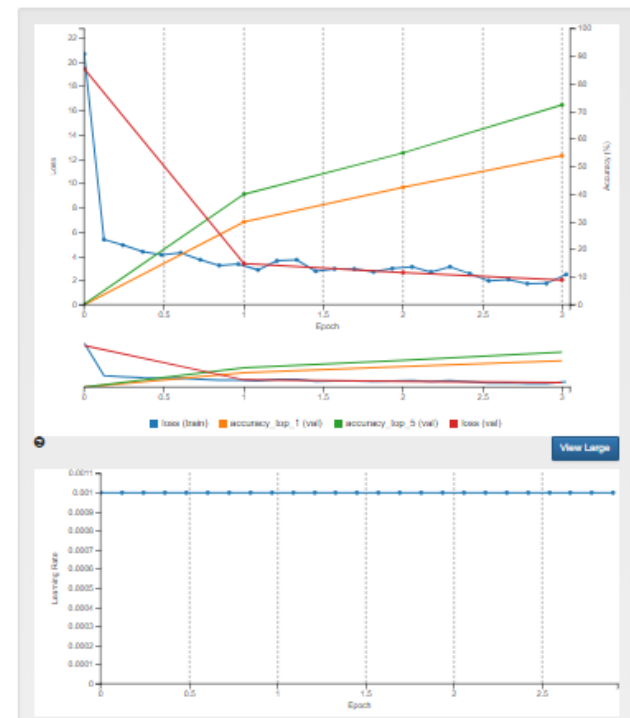
vgg-101-full Done ▾

Image Classification Model

VGG-16_101 Done ▾

Notes

None ☑



GPU工作状态

损失函数及精度的实时显示

学习率

DIGITS 的功能

图像分类



98% Dog

2% Cat

Classify images into classes or categories

Object of interest could be anywhere in the image

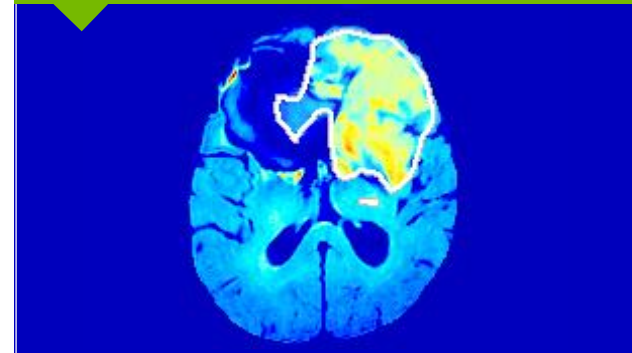
目标检测



Find instances of objects in an image

Objects are identified with bounding boxes

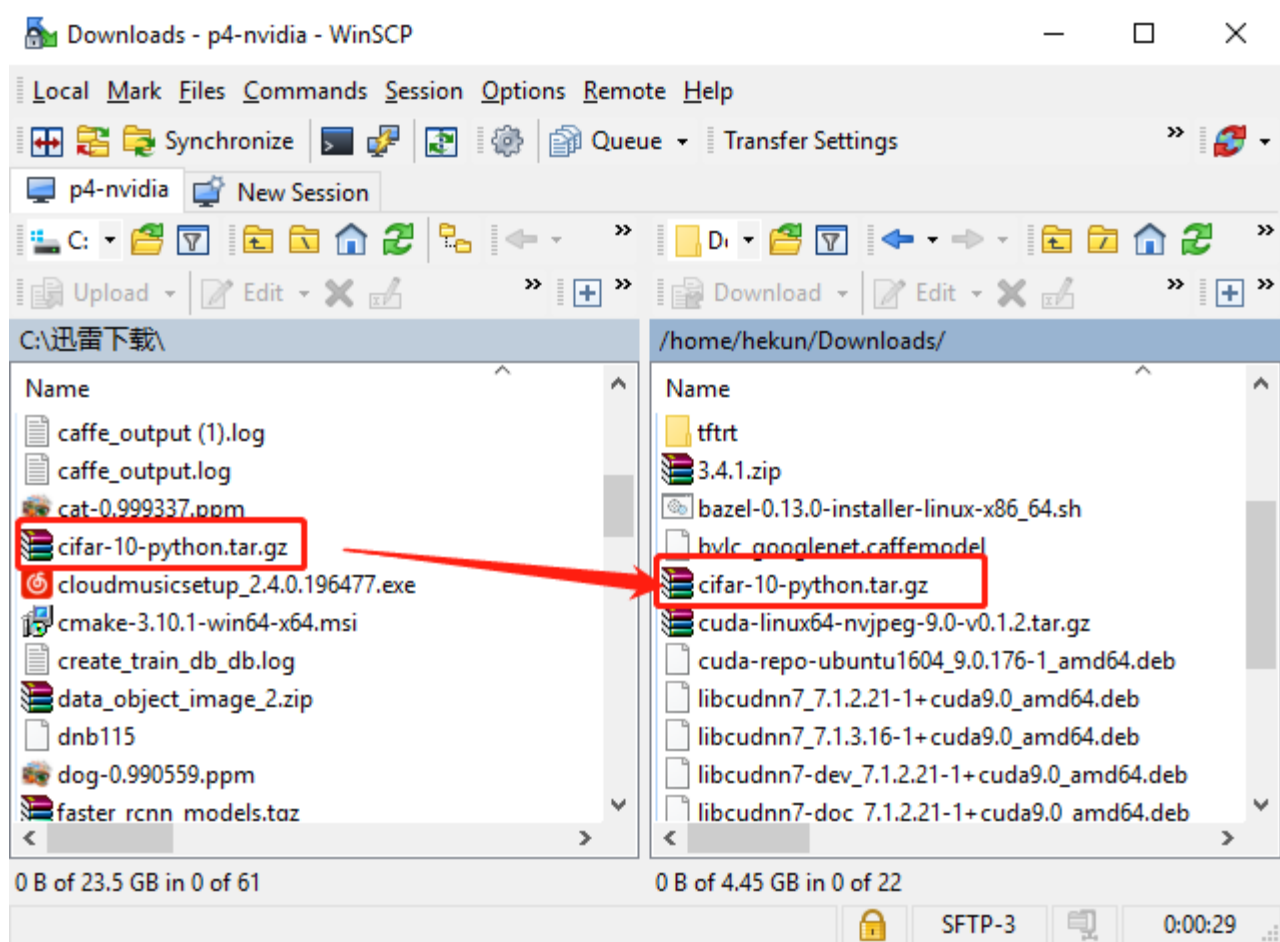
图像分割



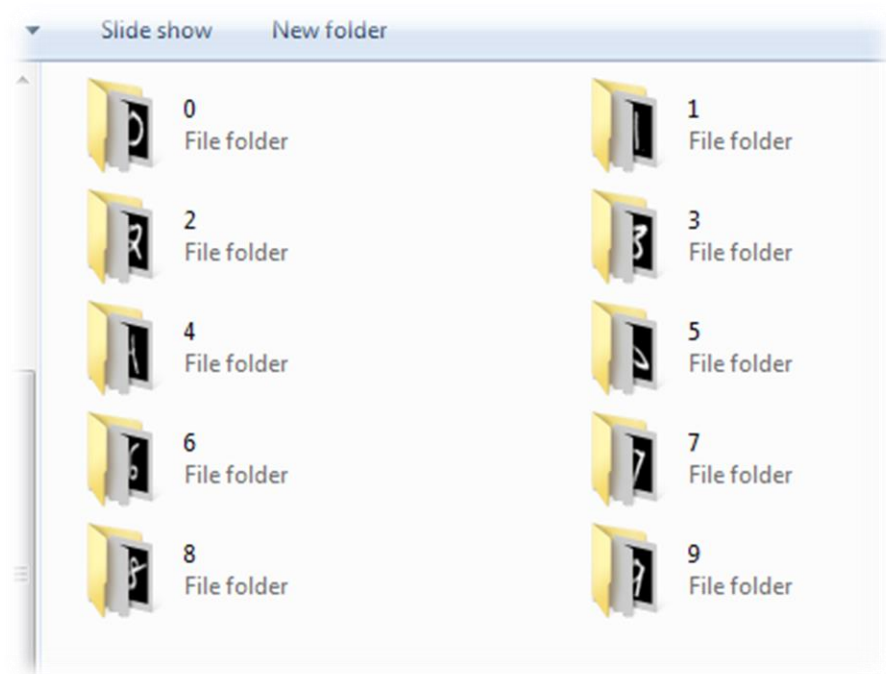
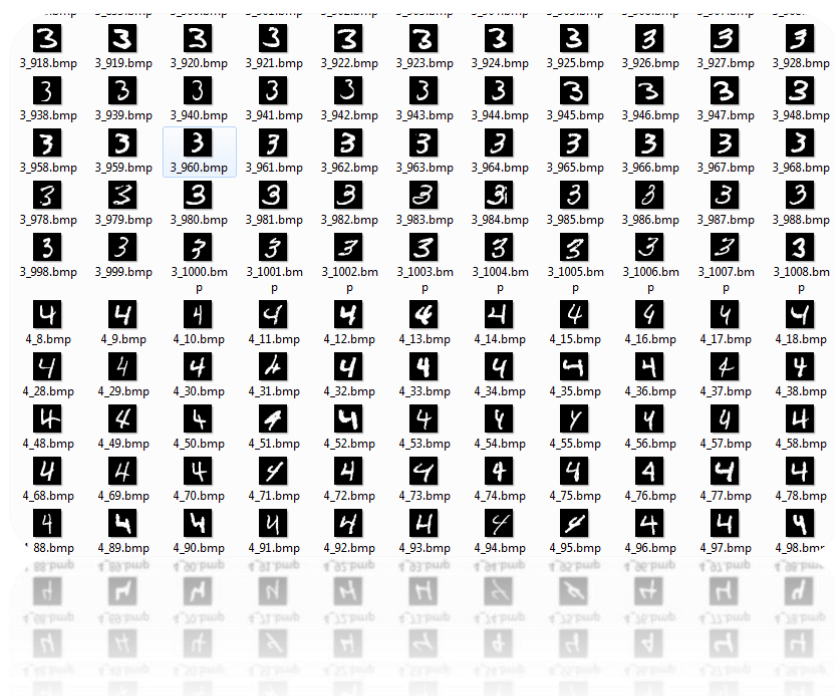
Partition image into multiple regions

Regions are classified at the pixel level

DIGITS 的训练数据集导入



DIGITS : 图片分类的训练数据集格式



DIGITS : 图片分类

创建用于训练的数据集：

1. 定义图像类型、尺寸、变换方式等
2. 定义图片数据集在服务器中的地址或者网络中的地址
3. 定义训练集和验证集的比例大小以及详细信息
4. 定义图片的编码格式以及训练集的格式
5. 给数据集命名

New Image Classification Dataset

The screenshot shows the 'New Image Classification Dataset' form in the DIGITS application. It is divided into several sections:

- Image Type:** A dropdown menu set to 'Color'. Below it, 'Image size (Width x Height)' is set to '256 x 256'. 'Resize Transformation' is set to 'Squash'. A 'See example' button is present.
- Source Selection:** Three tabs at the top: 'Use Image Folder' (selected), 'Use Text Files', and 'Use S3'.
- Training Images:** A text input field for 'folder or URL'. Below it, 'Minimum samples per class' is set to '2' and 'Maximum samples per class' is empty. '% for validation' is set to '25' and '% for testing' is set to '0'. There are two checkboxes: 'Separate validation images folder' and 'Separate test images folder', both of which are unchecked.
- DB backend:** A dropdown menu set to 'LMDB'.
- Image Encoding:** A dropdown menu set to 'PNG (lossless)'.
- Group Name:** An empty text input field.
- Dataset Name:** An empty text input field.
- Create:** A blue button at the bottom right to submit the form.

DIGITS : 图片分类

定义训练参数：

1. 选择已经创建好的数据集
2. 定义Epoch,batchsize,base_lr等训练参数
3. 选择使用的训练框架
4. 选择或者自定义网络结构
5. 给训练的模型命名

The screenshot displays the DIGITS web interface for configuring a training job. It is divided into several sections:

- Select Dataset:** A dropdown menu showing available datasets: 101_reduce_224x224, 101_ObjectCategories, test100, test1, and mnist-reverse-hakun.
- Python Layers:** A section for specifying the server-side file, with a checkbox for "Use client-side file".
- Solver Options:** A central panel for training parameters:
 - Training epochs:** 30
 - Snapshot interval (in epochs):** 1
 - Validation interval (in epochs):** 1
 - Random seed:** [none]
 - Batch size:** [network defaults] (multiples allowed)
 - Batch Accumulation:** [empty field]
 - Solver type:** SGD (Stochastic Gradient Descent)
 - Base Learning Rate:** 0.01 (multiples allowed)
 - ☐ Show advanced learning rate options
- Data Transformations:** A section for image processing:
 - Subtract Mean:** Image
 - Crop Size:** none

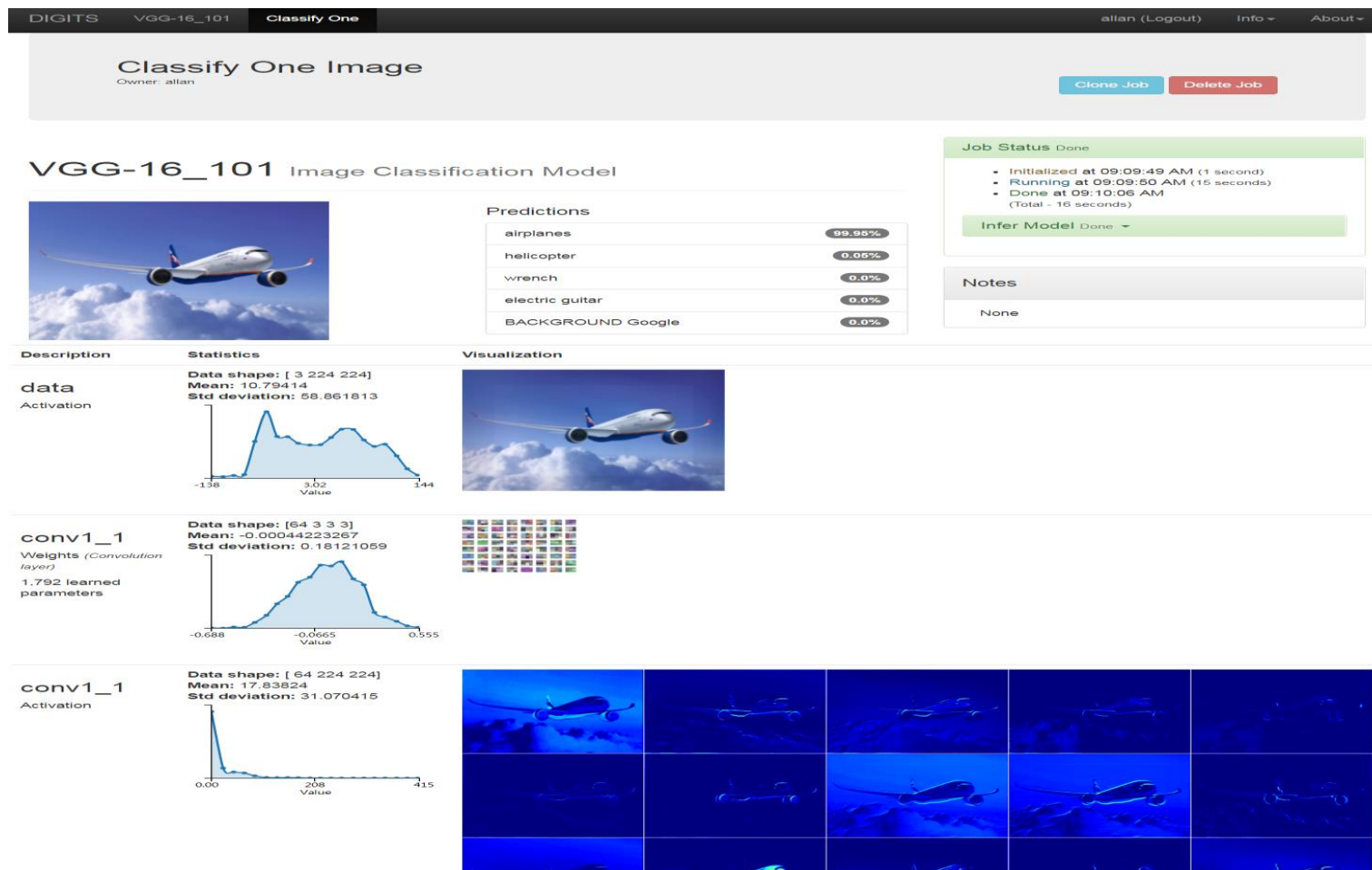
Below these sections, there is a tabbed interface for selecting a network structure. The "Standard Networks" tab is active, showing a table of pre-trained models:

Network	Details	Intended image size
LeNet	Original paper [1998]	28x28 (gray)
AlexNet	Original paper [2012]	256x256
GoogLeNet	Original paper [2014]	256x256

At the bottom, there is a "Group Name" and "Model Name" section with input fields and a "Create" button.

NVIDIA DIGITS

图片分类校验结果输出



DIGITS : 目标检测的训练数据集格式

每张图片对应有一个和它相同名字的label文件, label文件中包含了对应图片中目标的类别和代表位置的矩形框信息

名称	大小	压缩后大小	名称	大小	压缩后大小	类型	修改时间	CRC32
..			..			Local Disk		
000000.png	974,013	9	000000.txt	87	87	Text Document	8/6/2012 8:49 PM	8463E82E
000001.png	806,039	8	000001.txt	565	565	Text Document	8/6/2012 8:48 PM	0A7C5C2F
000002.png	981,429	9	000002.txt	164	164	Text Document	8/6/2012 8:49 PM	E087B097
000003.png	721,309	7	000003.txt	234	234	Text Document	8/6/2012 8:49 PM	4A81A2FC
000004.png	853,446	8	000004.txt	556	556	Text Document	8/6/2012 8:48 PM	3DA28D38
000005.png	967,522	9	000005.txt	400	400	Text Document	8/6/2012 8:49 PM	88BFCDE5
000006.png	811,426	8	000006.txt	486	486	Text Document	8/6/2012 8:49 PM	1E2E53AB
000007.png	884,873	8	000007.txt	487	487	Text Document	4/8/2015 12:03 AM	E7828B01
000008.png	853,828	8	000008.txt	798	798	Text Document	8/6/2012 8:49 PM	76210665
000009.png	893,482	8	000009.txt	399	399	Text Document	8/6/2012 8:48 PM	7BC798E4
000010.png	716,559	7	000010.txt	1,050	1,050	Text Document	8/6/2012 8:49 PM	3EA9E58C
000011.png	806,035	8	000011.txt	740	740	Text Document	8/6/2012 8:49 PM	E17802B2
000012.png	671,005	6	000012.txt	398	398	Text Document	8/6/2012 8:48 PM	EC95EE2C
000013.png	930,121	9	000013.txt	159	159	Text Document	8/6/2012 8:49 PM	4F326C76
000014.png	886,038	8	000014.txt	236	236	Text Document	8/6/2012 8:48 PM	0365BED8
000015.png	849,070	8	000015.txt	824	824	Text Document	8/6/2012 8:49 PM	B0F7048F
000016.png	782,362	7	000016.txt	1,753	1,753	Text Document	8/6/2012 8:48 PM	1308835E
000017.png	721,620	7	000017.txt	160	160	Text Document	8/6/2012 8:48 PM	FC3601E3
000018.png	826,091	8	000018.txt	1,587	1,587	Text Document	8/6/2012 8:48 PM	B28C21E7
000019.png	732,960	7	000019.txt	480	480	Text Document	8/6/2012 8:49 PM	E0F50280
000020.png	883,903	8	000020.txt	82	82	Text Document	8/6/2012 8:49 PM	FA68ABF4
000021.png	839,094	8	000021.txt	815	815	Text Document	8/6/2012 8:49 PM	F70349C7
000022.png	857,360	8	000022.txt	245	245	Text Document	8/6/2012 8:49 PM	DE235CB3
000023.png	875,918	8	000023.txt	487	487	Text Document	8/6/2012 8:48 PM	4B96B1D0
000024.png	855,197	8	000024.txt	562	562	Text Document	8/6/2012 8:49 PM	8933C5D3
000025.png	821,887	8	000025.txt	726	726	Text Document	8/6/2012 8:49 PM	E006D36E
000026.png	813,636	8	000026.txt	165	165	Text Document	8/6/2012 8:49 PM	0CBE7D7E
000027.png	767,925	7	000027.txt	241	241	Text Document	8/6/2012 8:48 PM	7A56D0AA
000028.png	879,097	8	000028.txt	89	89	Text Document	8/6/2012 8:49 PM	7E1E3CC5

DIGITS : 目标检测

创建用于训练的数据集：

1. 定义图像类型、尺寸、变换方式等
2. 定义图片数据集在服务器中的地址
3. 定义样本标签在服务器中的地址
4. 定义训练集和验证集的比例大小以及详细信息
5. 定义图片的编码格式以及训练集的格式
6. 给数据集命名

Object Detection Dataset Options

Images can be stored in any of the supported file formats (.png, .jpg, .jpeg, .bmp, .tiff).

Training image folder

Label files are expected to have the .txt extension. For example if an image file is named foo.png the corresponding label file should be foo.txt.

Training label folder

Validation image folder

Validation label folder

Pad image (Width x Height)

x

Resize image (Width x Height)

x

Channel conversion

RGB

Minimum box size (in pixels) for validation set

Custom classes

Feature Encoding

PNG (lossless)

Label Encoding

None

Encoder batch size

Number of encoder threads

DB backend

LMDB

Enforce same shape

Yes

Group Name

Dataset Name

Create

DIGITS : 目标检测

定义训练参数：

1. 选择已经创建好的数据集
2. 定义Epoch,batchsize,base_lr等训练参数
3. 选择使用的训练框架
4. 选择或者自定义网络结构
5. 选择是否使用Pretrained Model
6. 给训练的模型命名

New Object Detection Model

Select Dataset

KITT

Python Layers

Server-side file

Use client-side file

Solver Options

Training epochs

30

Snapshot interval (in epochs)

1

Validation interval (in epochs)

1

Random seed

[none]

Batch size

[network defaults]

Batch Accumulation

Solver type

SGD (Stochastic Gradient Descent)

Base Learning Rate

0.01

Show advanced learning rate options

Data Transformations

Subtract Mean

Image

Crop Size

none

Standard Networks Previous Networks Pretrained Networks Custom Network

Pretrained Model

loadFiles	Customize
test-1	Customize
celeb-a-gan	Customize
celeb-a-gan	Customize
AlexNet	Customize
GoogLeNet	Customize
VGG-16	Customize
DetectNet (KITTI)	Customize

Group Name

Model Name

Create

NVIDIA DIGITS

目标检测校验结果输出

DIGITS detector-epochs-100 Test One

Infer One Image
Owner: hekun

detector-epochs-100 Generic Image Model

Found 5 bounding boxes.

Source image



Inference visualization



■ bbox-list

DIGITS Generic Image Model

Trained Models

Select Model
Epoch #100 [Download Model](#) [Make Pretrained Model](#)

Select Visualization Method

Bounding boxes
Raw Data
Bounding boxes
GAN
Image output
Image Segmentation
Object Detection

Visualization Options

Draw a bounding box around a detected object. The expected network output is a dictionary, keyed by class, where the values are lists of bounding boxes for that image. e.g. {'class1': [[l, t, r, b, confidence], ...], ...}

☐ Show visualizations and statistics ?

[Test](#)

Test a single image

Image Path ?
/home/hekun/digits/examples/object-detection/kitti-

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
All
Leave blank to use all

[Test Many](#) ?

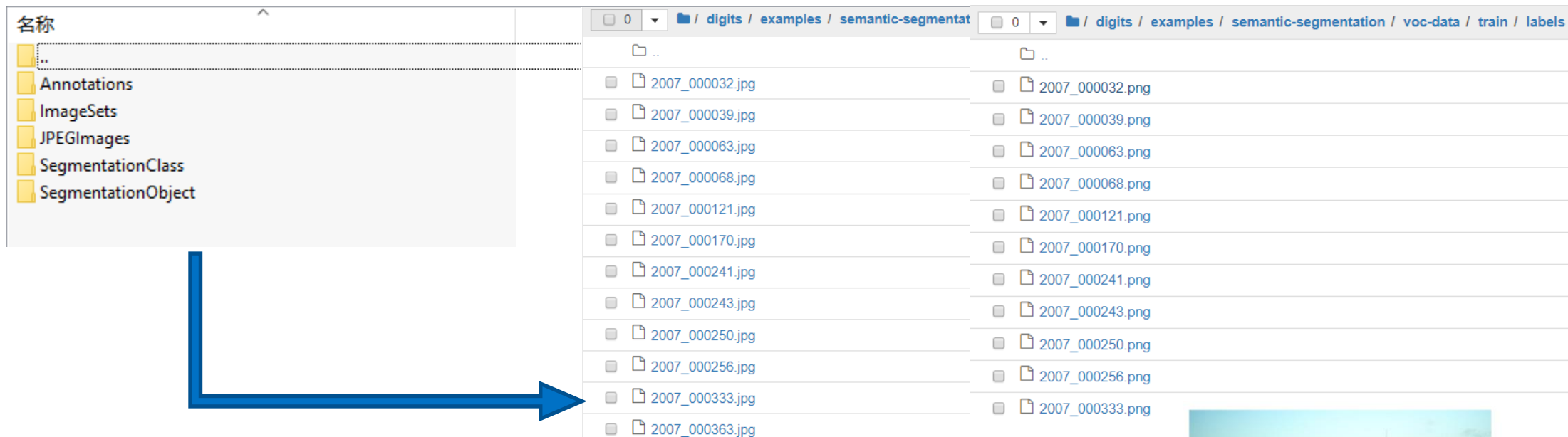
Test a Database

LMDb path

Specify path to LMDb database on server

[Test One](#)

DIGITS : 图像分割的训练数据集格式



```
$ ./prepare_pascal_voc_data.sh /data/VOCtrainval_11-May-2012.tar ./voc-data
```

图像分割数据集中包含了图片样本，以及他们的label文件。这里的label文件与之前的目标检测不同的地方在于，这个label文件本身也是一个图片的形式。它标注了对应的图片样本中，目标的边界。



DIGITS : 图像分割

New Segmentation Dataset

创建用于训练的数据集：

1. 定义图片数据集在服务器中的地址
2. 定义样本标签在服务器中的地址
3. 定义训练集和验证集的比例大小以及详细信息
4. 定义图片的编码格式以及训练集的格式
5. 定义样本标签的编码方式
6. 给数据集命名

Feature image folder ⓘ
folder

Label image folder ⓘ
folder

% for validation ⓘ
10

☐ Separate validation images

Class labels (optional) ⓘ
file

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 ⓘ
None ▼

Encoder batch size ⓘ
32

Number of encoder threads ⓘ
4

DB backend
LMDB ▼

Enforce same shape ⓘ
Yes ▼

Group Name

DIGITS : 图像分割

定义训练参数:

1. 选择已经创建好的数据集
2. 定义Epoch,batchsize,base_lr等训练参数
3. 选择使用的训练框架
4. 选择或者自定义网络结构
5. 选择是否使用Pretrained Model
6. 给训练的模型命名

New Image Model

Select Dataset

VOC-Segmentation
VOC-Segmentation
KIT2
KIT2
KIT11

VOC-Segmentation
Done 10:46:20 PM

- 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, 386, 500)
 - Label shape (1, 386, 500)

Solver Options

Training epochs 30

Snapshot interval (in epochs) 1.0

Validation interval (in epochs) 1.0

Random seed [none]

Batch size 2 multiple allowed

Batch Accumulation

Solver type SGD (Stochastic Gradient Descent)

Base Learning Rate 0.0001 multiple allowed

☐ Show advanced learning rate options

Data Transformations

Subtract Mean [None]

Crop Size [none]

Python Layers

Server-side file

☐ Use client-side file

Standard Networks Previous Networks Pretrained Networks Custom Network

Caffe tensorflow

Custom Network Visualize

```
1 {
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 }
```

Pretrained model(s)

./models/digits/complex/semantic-segmentation/cnn_alexnet_caffe_model

Group Name

Model Name segmentation1

Create

NVIDIA DIGITS

图像分割校验结果输出

Infer One Image

Owner: hekun

Clone Job

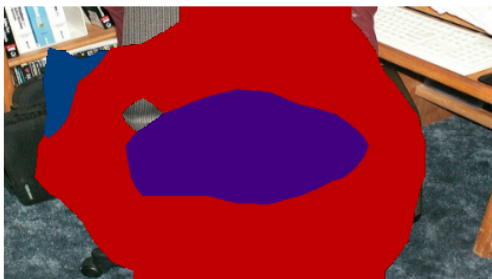
Delete Job

segmentation3 Generic Image Model

Source image



Inference visualization



chair dog tvmonitor

Job Status Done

- Initialized at 06:41:22 AM (1 second)
- Running at 06:41:23 AM (8 seconds)
- Done at 06:41:31 AM (Total - 9 seconds)

Infer Model Done

Notes

None

Trained Models

Select Model

Epoch #28

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)

Select Inference form

Segmentation

☐ Show visualizations and statistics

Test

Test a single image

Image Path

/home/hekun/digits/examples/semantic-segmentati

Upload image

Browse...

☐ 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

NVIDIA DIGITS 模型输出

Trained Models

Select Model

Epoch #5

Download Model

Make Pretrained Model

Publish to inference server

Test a single image

Image Path ?

Upload image

Browse...

☐ Show visualizations and statistics ?

Classify 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

All

Leave blank to use all

Classify Many ?

Number of images to show per category

9

Top N Predictions per Category ?

20180708-064929-6e4b_epoch_10.0.tar.gz 7/8/2018 6:50 AM WinRAR 压缩文件 1,561 KB

名称	大小	压缩后大小	类型	修改时间	CRC32
Local Disk					
deploy.prototxt	1,823	?	PROTOTXT File	7/8/2018 6:49 ...	
info.json	769	?	JSON File	1/1/1970 8:00 ...	
labels.txt	20	?	Text Document	7/8/2018 6:47 ...	
mean.binaryproto	3,147	?	BINARYPROTO File	7/8/2018 6:48 ...	
original.prototxt	2,346	?	PROTOTXT File	7/8/2018 6:49 ...	
snapshot_iter_7040.caffemodel	1,725,144	?	CAFFEMODEL File	7/8/2018 6:50 ...	
solver.prototxt	300	?	PROTOTXT File	7/8/2018 6:49 ...	
train_val.prototxt	2,600	?	PROTOTXT File	7/8/2018 6:49 ...	

迁移学习在DIGITS里的使用

加载model store里的模型

DIGITS Login Info About

Home 1/1 GPU available

No Jobs Running

Datasets (7) Models (6) Pretrained Models (6)

Group Jobs: ☒ Load Model Images

Retrieve from Model Store
Upload Pretrained Model

name	framework	username	has_labels	status	elapsed	submitted
▼ Ungrouped						
celeb-a-gan	tensorflow	hekun	✗	Done	0s	Sep 1, 18
celeb-a-gan	tensorflow	hekun	✗	Done	0s	Sep 1, 18
AlexNet	caffe	allan	✗	Done	0s	Aug 11, 18
GoogLeNet	caffe	allan	✗	Done	0s	Aug 11, 18
VGG-16	caffe		✗	Done	0s	Jul 26, 18
DetectNet (KITTI)	caffe		✗	Done	0s	Jul 26, 18

DIGITS Login Info About

Model Store

Update Model List

Filter

Name	Contributor	Affiliate	Note	Data sets	License
NVIDIA Public Model Store					
LeNet	NVIDIA			MNIST	Multiple
AlexNet	NVIDIA		Top1: 58.5%, Top5: 81.3	ILSVRC2012	BSD-3-Clause
GoogLeNet	NVIDIA		Top1: 72.1%, Top5: 91.0	ILSVRC2012	BSD-3-Clause
VGG-16	NVIDIA		Top1: 75.0%, Top5: 93.4	ILSVRC2012	Multiple
DetectNet (KITTI)	NVIDIA			ILSVRC2012, KITTI	BSD-3-Clause
celeb-a-gan	NVIDIA				
celeb-a-gan-encoder	NVIDIA				
U-NET	NVIDIA			ISBI	

- Image Classification: VGG-16,
- Object Detection: DetectNet

Standard Networks Previous Networks Pretrained Networks Custom Network

Caffe Tensorflow

Pretrained Model

AlexNet caffe

GoogLeNet caffe

VGG-16 caffe

DetectNet (KITTI) caffe

Customize

Group Name

Model Name

Create

迁移学习在DIGITS里的使用

加载自己的模型

DIGITS Login Info About

Home 1/1 GPU available

No Jobs Running

Datasets (7) Models (6) Pretrained Models (6)

Group Jobs: ☒

Load Model Images









Delete Group

Filter

Retrieve from Model Store
Upload Pretrained Model

name	framework	username	has_labels	status	elapsed	submitted
▼ Ungrouped						
celeb-a-gan	tensorflow	hekun	✗	Done	0s	Sep 1, 18
celeb-a-gan	tensorflow	hekun	✗	Done	0s	Sep 1, 18
AlexNet	caffe	allan	✗	Done	0s	Aug 11, 18
GoogLeNet	caffe	allan	✗	Done	0s	Aug 11, 18
VGG-16	caffe		✗	Done	0s	Jul 26, 18
DetectNet (KITTI)	caffe		✗	Done	0s	Jul 26, 18

20180626-151718-00e7_epoch_5.0.tar.gz

Name	Date modified	Type	Size
 deploy.prototxt	6/26/2018 3:17 PM	PROTOTXT File	2 KB
 info.json		JSON File	1 KB
 labels.txt	6/26/2018 3:14 PM	Text Document	1 KB
 mean.binaryproto	6/26/2018 3:16 PM	BINARYPROTO File	4 KB
 original.prototxt	6/26/2018 3:17 PM	PROTOTXT File	3 KB
 snapshot_iter_7035.caffemodel	6/26/2018 3:17 PM	CAFFEMODEL File	1,685 KB
 solver.prototxt	6/26/2018 3:17 PM	PROTOTXT File	1 KB
 train_val.prototxt	6/26/2018 3:17 PM	PROTOTXT File	3 KB

DIGITS allan (Logout) Info About

Home 1/1 GPU available

No Jobs Running

Datasets (7) Models (7) Pretrained Models (7)

Load Model Images

Delete Group

Filter

Retrieve from Model Store
Upload Pretrained Model

name	framework	username	has_labels	status	elapsed	submitted
▼ Ungrouped						
test-1	caffe		✓	Done	0s	7:24 AM
celeb-a-gan	tensorflow	hekun	✗	Done	0s	Sep 1, 18
celeb-a-gan	tensorflow	hekun	✗	Done	0s	Sep 1, 18
AlexNet	caffe	allan	✗	Done	0s	Aug 11, 18
GoogLeNet				Done	0s	Aug 11, 18
VGG-16				Done	0s	Jul 26, 18
DetectNet (KITTI)				Done	0s	Jul 26, 18

Upload Pretrained Model

Manual Entry Upload Tar or Zip Archive

Jobname: load3files Framework: Caffe

Image Type: Grayscale Resize Mode: Squash

Width: 28 Height: 30

Weights (**.caffemodel): snapshot_iter_7035.caffem

Model Definition (original.prototxt): original.prototxt

Labels file: (labels.txt): labels.txt

Upload Model

使用Plugin导入数据及自定义显示输出

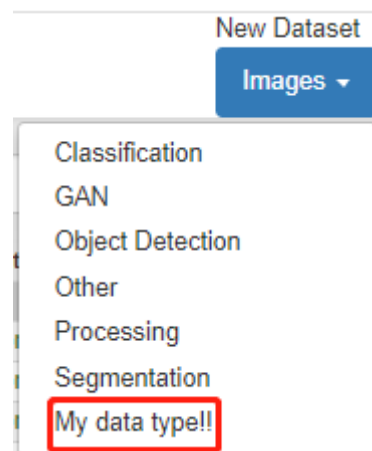
GAN?

医疗影像分析?

文字语义识别?

更多类型的模型训练?

Plugin



Below is an example file tree for a data plugin:

```
sunnybook/
├── digitsDataPluginSunnybrook/
│   ├── templates
│   │   ├── dataset_template.html
│   │   └── inference_template.html
│   ├── __init__.py
│   ├── data.py
│   └── forms.py
├── MANIFEST.in
└── setup.py
```

```
@subclass
class DatasetForm(Form):
    """
    A form used to create a Sunnybrook dataset
    """

    def validate_folder_path(form, field):
        if not field.data:
            pass
        else:
            # make sure the filesystem path exists
            if not os.path.exists(field.data) or not os.path.isdir(field.data):
                raise validators.ValidationError(
                    'Folder does not exist or is not reachable')
            else:
                return True

    image_folder = utils.forms.StringField(
        u'Image folder',
        validators=[
            validators.DataRequired(),
            validate_folder_path,
        ],
        tooltip="Specify the path to the image folder"
    )

    contour_folder = utils.forms.StringField(
        u'Contour folder',
        validators=[
            validators.DataRequired(),
            validate_folder_path,
        ],
        tooltip="Specify the path to the contour folder"
    )

    channel_conversion = utils.forms.SelectField(
        'Channel conversion',
        choices=[
            ('RGB', 'RGB'),
            ('L', 'Grayscale'),
        ],
        default='L',
        tooltip="Perform selected channel conversion."
    )

    folder_pct_val = utils.forms.IntegerField(
        u'% for validation',
        default=10,
        validators=[
```

如何得到DIGITS

- **Simple way:**

- OS – Ubuntu16.04
- Download link:
<https://developer.nvidia.com/digits>

Others (from source code):

- Download NVIDIA-Caffe:
<https://github.com/NVIDIA/caffe>
- Download Digits:
<https://github.com/NVIDIA/DIGITS>
- More Examples:
<https://github.com/NVIDIA/DIGITS/tree/master/examples>

Recommended HW/SW environment:

- GPU Compute Capability > 3.5 (Kepler and later), cuDNN v7
- OS – Ubuntu16.04

深度学习培训 @ GTC CHINA 2018

6大主题全中文首发 | 6 折限时特惠 | 全配置GPU云实验环境 | 全球标准培训证书

自动驾驶感知

深度学习模型训练及部署至自动驾驶汽车核心DrivePX

游戏开发

训练深度强化学习代理玩星际争霸2

计算机视觉升级版

快速入门深度学习完整实践流程

多数据类型

1天解锁计算机视觉和自然语言处理两大技能

自然语言处理

用深度学习方法完成文本分类和机器翻译

创建数字内容

为媒体娱乐业注入“深度学习”新活力



查看课程和报名



培训咨询