show_results

October 3, 2016

1 Load model

1.1 Model

conv1_1 = new_conv_layer(image, [3, 3, 1 , 16], “conv1_1”)
conv1_2 = new_conv_layer(conv1_1, [3, 3, 16, 16], “conv1_2”)
pool1 = tf.nn.max_pool(conv1_2, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1], padding=‘SAME’, name=‘pool1’)
conv2_1 = new_conv_layer(pool1, [3, 3, 16, 16], “conv2_1”)
conv2_2 = new_conv_layer(conv2_1, [3, 3, 16, 16], “conv2_2”)
pool2 = tf.nn.max_pool(conv2_2, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1], padding=‘SAME’, name=‘pool2’)
conv3_1 = new_conv_layer(pool2, [3, 3, 16, 16], “conv3_1”)
fc1 = self.new_fc_layer(conv3_1, 25*25*16, 512, ‘fc1’)
fc2 = self.new_fc_layer(fc1 , 512 , 10 , ‘fc2’)

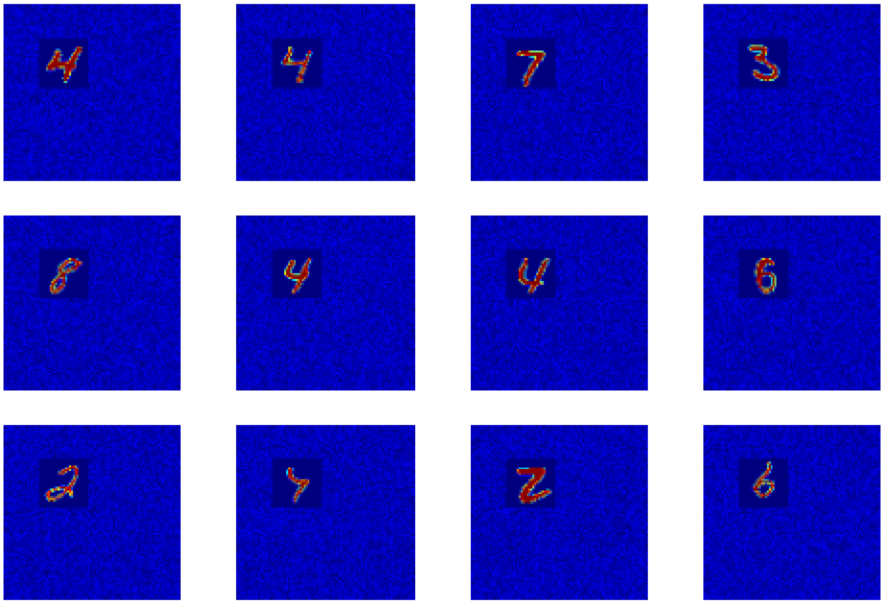
1.2 train params

L2 on weights (5e-5)

```
In [1]: import matplotlib.pyplot as plt
        import numpy as np
        import simple_model
        from simple_model import training_generator
        import utils


        simple_model=reload(simple_model)
        lr=.005
        back_size=100
        crop_pos=(20,20)
```

```
Exception AssertionError: AssertionError("Nesting violated for default stack of <ty
```

```
In [2]: """Plot training samples"""
        batch = utils.get_batch('train', 3*4, back_size, .1, crop_pos).next()
        fig, axs = plt.subplots(3,4)
        for ax,img in zip([b for a in axs for b in a],batch[0]):
            ax.imshow(img.reshape(back_size,back_size), vmin=0, vmax=1)
            ax.set_axis_off()
        plt.show()
```



```
In [3]: """Perform training"""
        gen = training_generator(lr=.005, back_size=back_size, crop_pos=(20,20) )
        for _ in range(15):
            _,accs = gen.next()
            print "max acc so far : "+str(max(accs)*100)
```

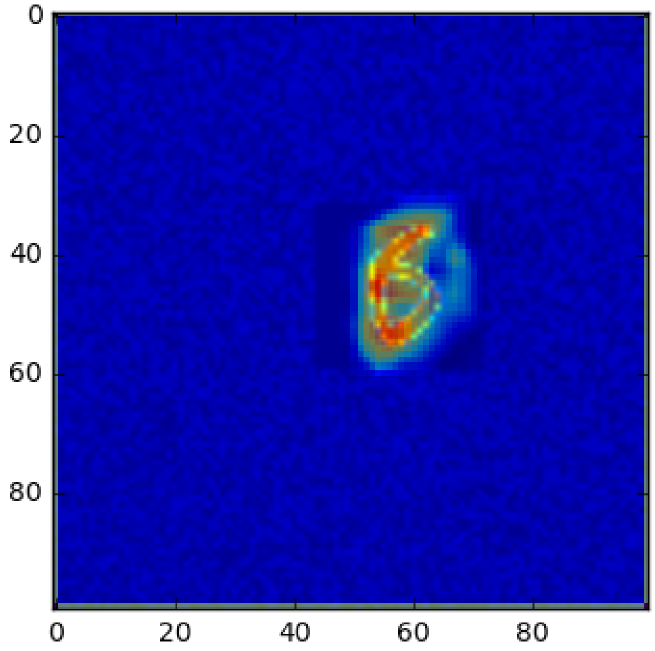
```
***** EPOCH 0 ******
Loss on testset is 2.177958
Accuracy now is 20.61
lr now is 0.00475
max acc so far : 20.61

***** EPOCH 1 ******
Loss on testset is 1.837432
Accuracy now is 30.08
lr now is 0.00451
max acc so far : 30.08
```

```
***** EPOCH 2 ******
Loss on testset is 1.359812
Accuracy now is 45.65
lr now is 0.00429
max acc so far : 45.65

***** EPOCH 3 ******
Loss on testset is 0.881984
Accuracy now is 68.78
lr now is 0.00407
max acc so far : 68.78

***** EPOCH 4 ******
Loss on testset is 0.708447
Accuracy now is 76.57
lr now is 0.00387
max acc so far : 76.57

***** EPOCH 5 ******
Loss on testset is 0.395126
Accuracy now is 88.89
lr now is 0.00368
max acc so far : 88.89

***** EPOCH 6 ******
Loss on testset is 0.322769
Accuracy now is 91.96
lr now is 0.00349
max acc so far : 91.96

***** EPOCH 7 ******
Loss on testset is 0.462606
Accuracy now is 88.71
lr now is 0.00332
max acc so far : 91.96

***** EPOCH 8 ******
Loss on testset is 0.408478
Accuracy now is 88.47
lr now is 0.00315
max acc so far : 91.96

***** EPOCH 9 ******
Loss on testset is 0.216448
Accuracy now is 94.66
lr now is 0.00299
max acc so far : 94.66
```

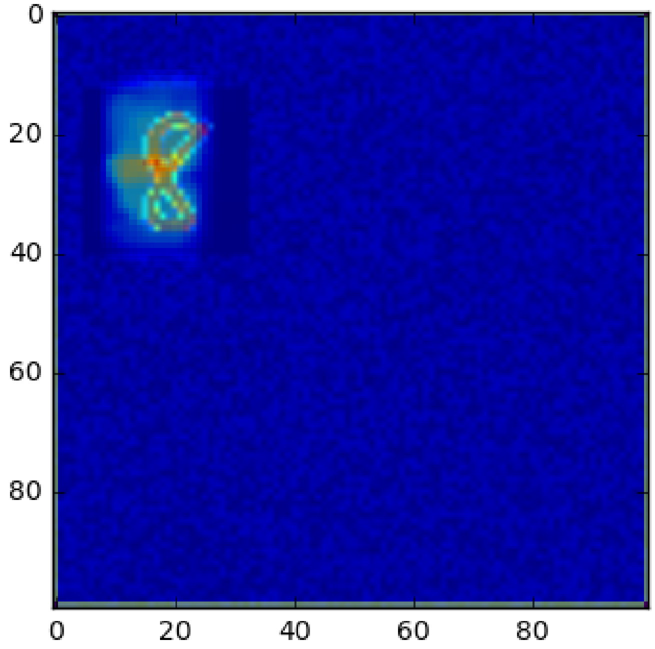
```
***** EPOCH 10 ******
Loss on testset is 0.366530
Accuracy now is 90.04
lr now is 0.00284
max acc so far : 94.66

***** EPOCH 11 ******
Loss on testset is 0.193801
Accuracy now is 95.27
lr now is 0.00270
max acc so far : 95.27

***** EPOCH 12 ******
Loss on testset is 0.158384
Accuracy now is 96.39
lr now is 0.00257
max acc so far : 96.39

***** EPOCH 13 ******
Loss on testset is 0.159983
Accuracy now is 96.25
lr now is 0.00244
max acc so far : 96.39

***** EPOCH 14 ******
Loss on testset is 0.206072
Accuracy now is 95.40
lr now is 0.00232
max acc so far : 96.39
```

```
In [4]: back_size  = 100

        for _ in range(5):
            print '----------------'*5
            imgs, lbls = utils.get_batch('test', 1, back_size, .1).next()

            simple_model.show_activation(imgs[0])
```

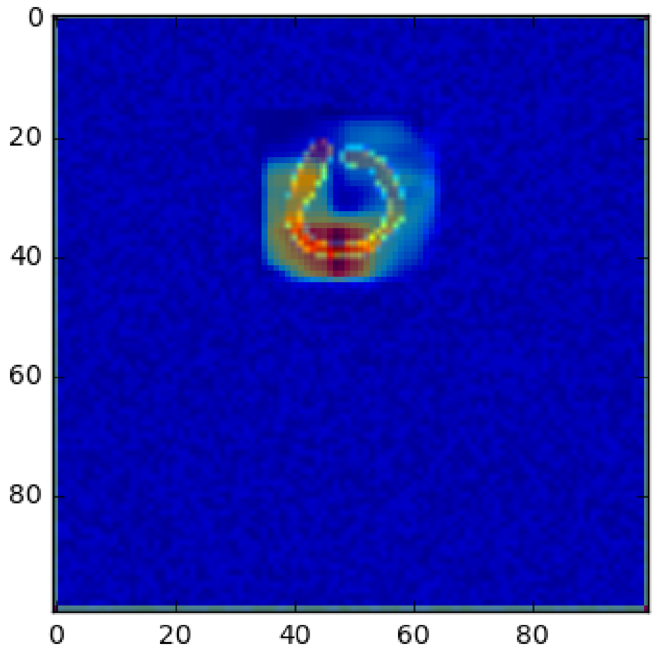
```
--------------------------------------------------------------------------------
prediction is : 6 with 16.228
```



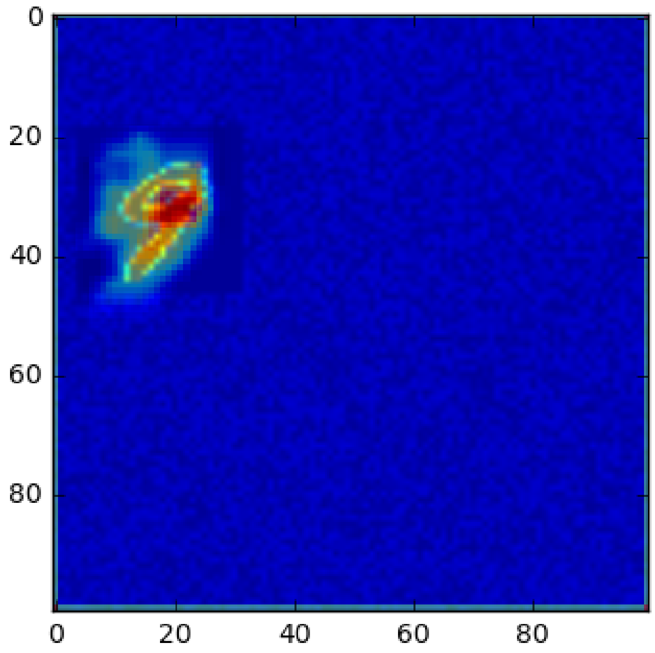
```
--------------------------------------------------------------------------------
prediction is : 8 with 17.389
```



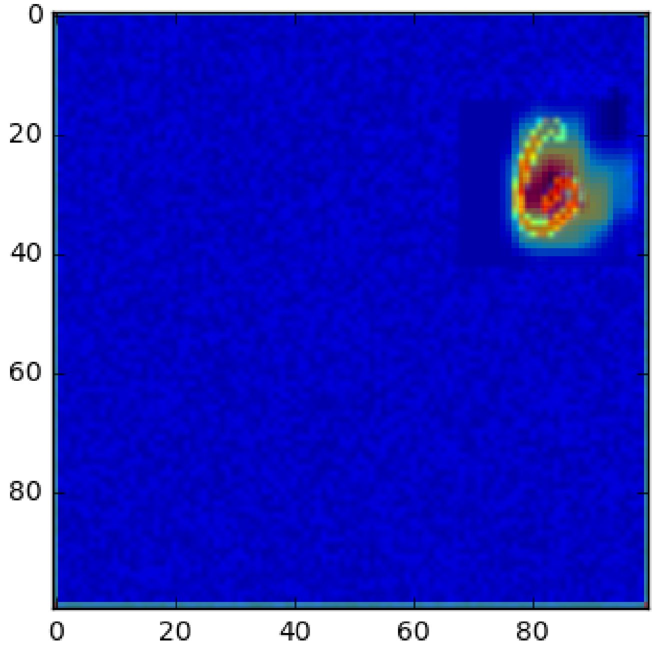
```
--------------------------------------------------------------------------------
prediction is : 0 with 22.961
```



```
--------------------------------------------------------------------------------
prediction is : 9 with 10.489
```



```
--------------------------------------------------------------------------------
prediction is : 6 with 17.753
```



```
In [ ]:
```

```
In [ ]:
```