show_results

November 8, 2016

1 Load model

1.1 Model

conv1_1 = new_conv_layer(image, [9, 9, 1 , 16], “conv1_1”) conv1_2 = new_conv_layer(conv1_1, [9, 9, 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 , [9, 9, 16, 16], “conv2_1”) conv2_2 = new_conv_layer(conv2_1, [9, 9, 16, 16], “conv2_2”) pool2 = tf.nn.max_pool(conv1_2, ksize=[1, 2, 2, 1], strides=[1, 2, 2, 1],padding=‘SAME’, name=‘pool2’)

conv3_1 = new_conv_layer(pool2 , [9, 9, 16, 16], “conv3_1”) conv3_2 = new_conv_layer(conv2_1, [9, 9, 16, 16], “conv3_2”)

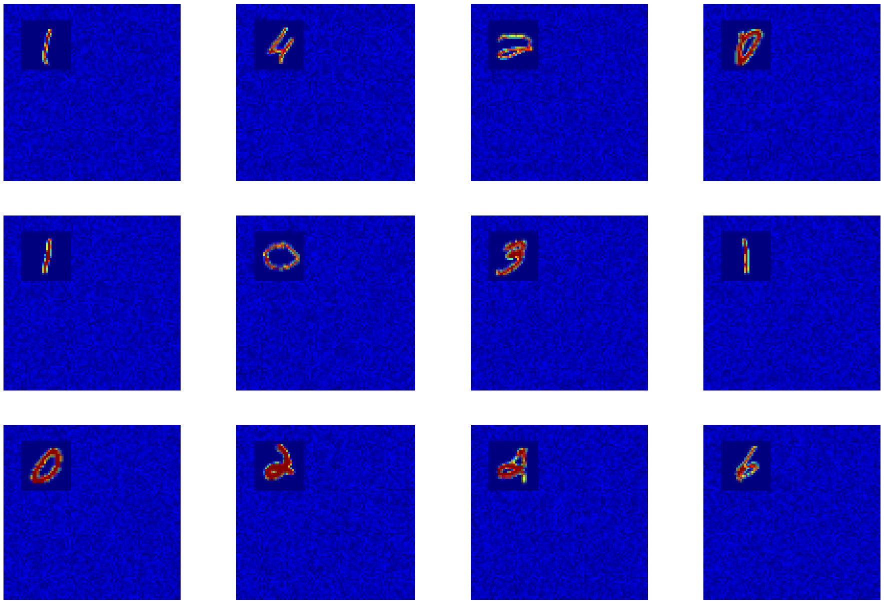
gap = tf.reduce_mean(conv3_2, [1,2]) ccn = tf.reshape(gap,[-1,10,nb_CCN]) ccn = tf.reduce_mean(ccn, 2)

1.2 train params

L2 on weights (5e-5) L1 on GAP Batch normalisation

```
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
        lr_decay  = .9
        back_size = 100
        noise     = .1
        crop_pos  = (10,10)
        n_CCN     = 5
```

```
Exception AssertionError: AssertionError("Nesting violated for default stack of <ty
In [2]: """Plot training samples"""
        batch = utils.get_batch('train', im_size=back_size, noise=noise, crop_pos=c

        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=lr, lr_decay=lr_decay, back_size=back_size, noi
        for _ in range(25):
            _,accs = gen.next()
            print "max acc so far : "+str(max(accs)*100)
```

```
***** EPOCH 0 ******
1-Loss on testset is 0.491104
1-Accuracy now is 95.04
2-Loss on testset is 0.379346
2-Accuracy now is 96.93
lr now is 0.00450
max acc so far : 95.04

***** EPOCH 1 ******
```

```
1-Loss on testset is 0.422786
1-Accuracy now is 95.08
2-Loss on testset is 0.309005
2-Accuracy now is 97.97
lr now is 0.00405
max acc so far : 95.08

***** EPOCH 2 ******
1-Loss on testset is 0.550990
1-Accuracy now is 94.20
2-Loss on testset is 0.352121
2-Accuracy now is 98.04
lr now is 0.00365
max acc so far : 95.08

***** EPOCH 3 ******
1-Loss on testset is 0.461325
1-Accuracy now is 95.39
2-Loss on testset is 0.279261
2-Accuracy now is 98.51
lr now is 0.00328
max acc so far : 95.39

***** EPOCH 4 ******
1-Loss on testset is 0.517542
1-Accuracy now is 92.34
2-Loss on testset is 0.258814
2-Accuracy now is 98.49
lr now is 0.00295
max acc so far : 95.39

***** EPOCH 5 ******
1-Loss on testset is 0.538104
1-Accuracy now is 93.51
2-Loss on testset is 0.274773
2-Accuracy now is 98.32
lr now is 0.00266
max acc so far : 95.39

***** EPOCH 6 ******
1-Loss on testset is 0.443324
1-Accuracy now is 94.85
2-Loss on testset is 0.234324
2-Accuracy now is 98.95
lr now is 0.00239
max acc so far : 95.39

***** EPOCH 7 ******
```

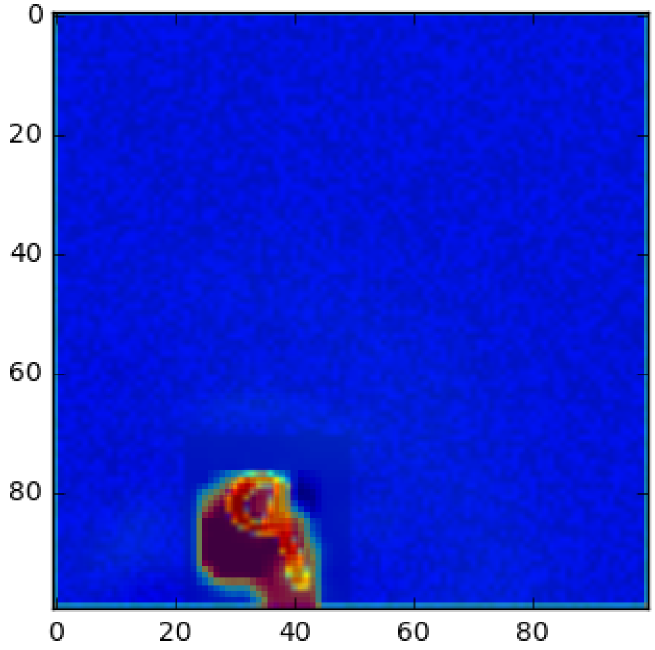
```
1-Loss on testset is 0.462213
1-Accuracy now is 94.77
2-Loss on testset is 0.212359
2-Accuracy now is 99.30
lr now is 0.00215
max acc so far : 95.39

***** EPOCH 8 ******
1-Loss on testset is 0.502543
1-Accuracy now is 92.82
2-Loss on testset is 0.215647
2-Accuracy now is 99.06
lr now is 0.00194
max acc so far : 95.39

***** EPOCH 9 ******
1-Loss on testset is 0.442977
1-Accuracy now is 94.78
2-Loss on testset is 0.225591
2-Accuracy now is 99.18
lr now is 0.00174
max acc so far : 95.39

***** EPOCH 10 ******
1-Loss on testset is 0.693889
1-Accuracy now is 88.58
2-Loss on testset is 0.210508
2-Accuracy now is 98.98
lr now is 0.00157
max acc so far : 95.39

***** EPOCH 11 ******
1-Loss on testset is 0.451239
1-Accuracy now is 93.68
2-Loss on testset is 0.202411
2-Accuracy now is 99.07
lr now is 0.00141
max acc so far : 95.39

***** EPOCH 12 ******
1-Loss on testset is 0.362518
1-Accuracy now is 96.41
2-Loss on testset is 0.193480
2-Accuracy now is 99.20
lr now is 0.00127
max acc so far : 96.41

***** EPOCH 13 ******
```

```
1-Loss on testset is 0.607089
1-Accuracy now is 89.93
2-Loss on testset is 0.207261
2-Accuracy now is 98.81
lr now is 0.00114
max acc so far : 96.41

***** EPOCH 14 ******
1-Loss on testset is 0.355806
1-Accuracy now is 96.23
2-Loss on testset is 0.179072
2-Accuracy now is 99.29
lr now is 0.00103
max acc so far : 96.41

***** EPOCH 15 ******
1-Loss on testset is 0.368860
1-Accuracy now is 96.19
2-Loss on testset is 0.189224
2-Accuracy now is 99.25
lr now is 0.00093
max acc so far : 96.41

***** EPOCH 16 ******
1-Loss on testset is 0.365751
1-Accuracy now is 95.20
2-Loss on testset is 0.178479
2-Accuracy now is 99.38
lr now is 0.00083
max acc so far : 96.41

***** EPOCH 17 ******
1-Loss on testset is 0.423728
1-Accuracy now is 93.73
2-Loss on testset is 0.177052
2-Accuracy now is 99.27
lr now is 0.00075
max acc so far : 96.41

***** EPOCH 18 ******
1-Loss on testset is 0.414389
1-Accuracy now is 92.96
2-Loss on testset is 0.172153
2-Accuracy now is 99.27
lr now is 0.00068
max acc so far : 96.41

***** EPOCH 19 ******
```

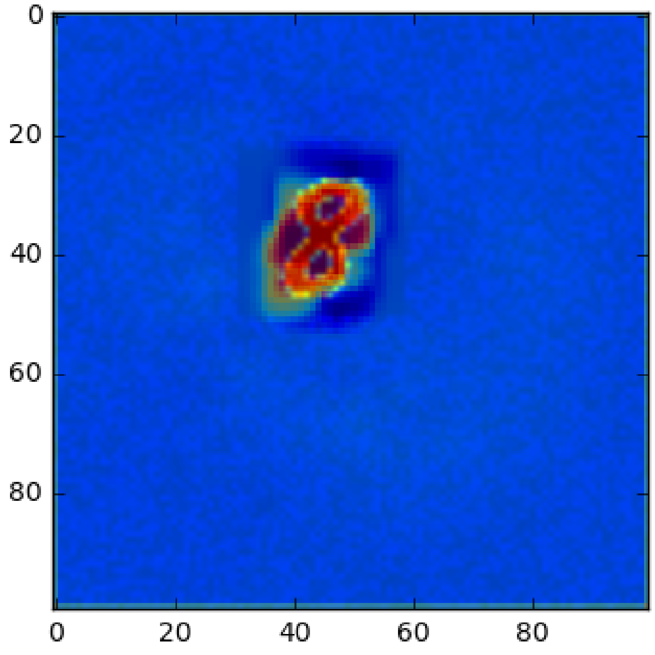
```
1-Loss on testset is 0.443332
1-Accuracy now is 92.01
2-Loss on testset is 0.165161
2-Accuracy now is 99.49
lr now is 0.00061
max acc so far : 96.41

***** EPOCH 20 ******
1-Loss on testset is 0.418533
1-Accuracy now is 92.76
2-Loss on testset is 0.178990
2-Accuracy now is 99.18
lr now is 0.00055
max acc so far : 96.41

***** EPOCH 21 ******
1-Loss on testset is 0.481347
1-Accuracy now is 88.54
2-Loss on testset is 0.166556
2-Accuracy now is 99.42
lr now is 0.00049
max acc so far : 96.41

***** EPOCH 22 ******
1-Loss on testset is 0.764983
1-Accuracy now is 78.44
2-Loss on testset is 0.173329
2-Accuracy now is 99.13
lr now is 0.00044
max acc so far : 96.41

***** EPOCH 23 ******
1-Loss on testset is 0.601726
1-Accuracy now is 84.33
2-Loss on testset is 0.165795
2-Accuracy now is 99.40
lr now is 0.00040
max acc so far : 96.41

***** EPOCH 24 ******
1-Loss on testset is 1.152715
1-Accuracy now is 69.80
2-Loss on testset is 0.169779
2-Accuracy now is 99.28
lr now is 0.00036
max acc so far : 96.41
```

```
In [4]: back_size  = 100

        for _ in range(10):
            print '----------------'*5
            imgs, lbls = utils.get_batch('test', 1, back_size, .1).next()

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

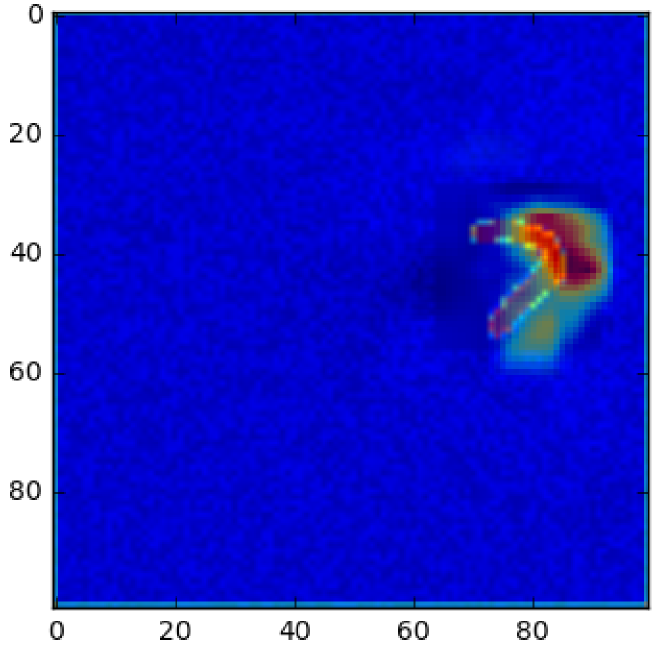
```
--------------------------------------------------------------------------------
prediction is : 9 with 7.396
```



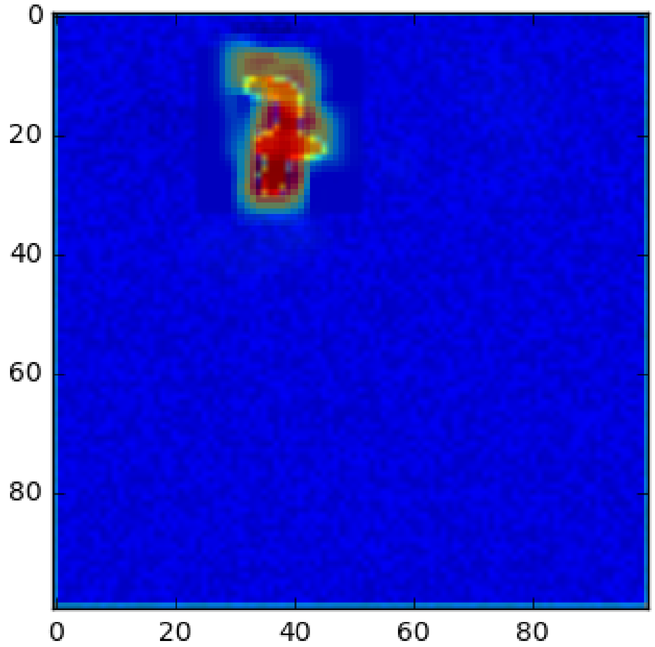
```
--------------------------------------------------------------------------------
prediction is : 8 with 9.911
```



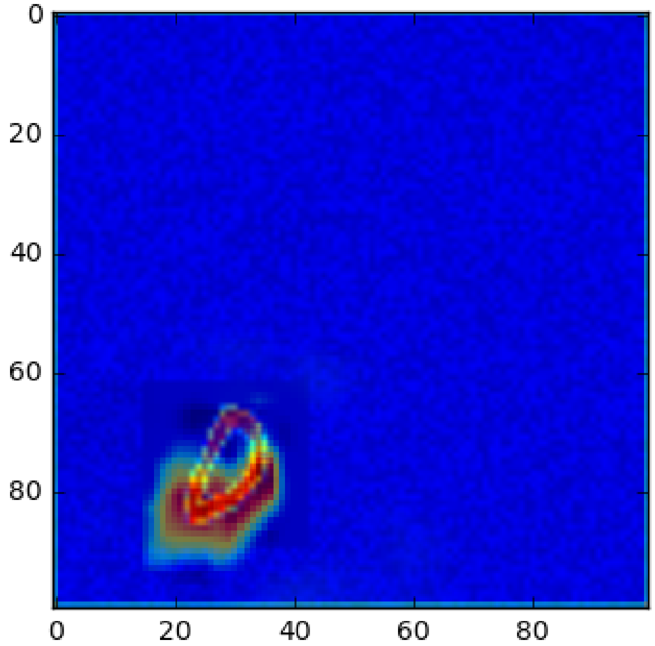
```
--------------------------------------------------------------------------------
prediction is : 7 with 14.811
```



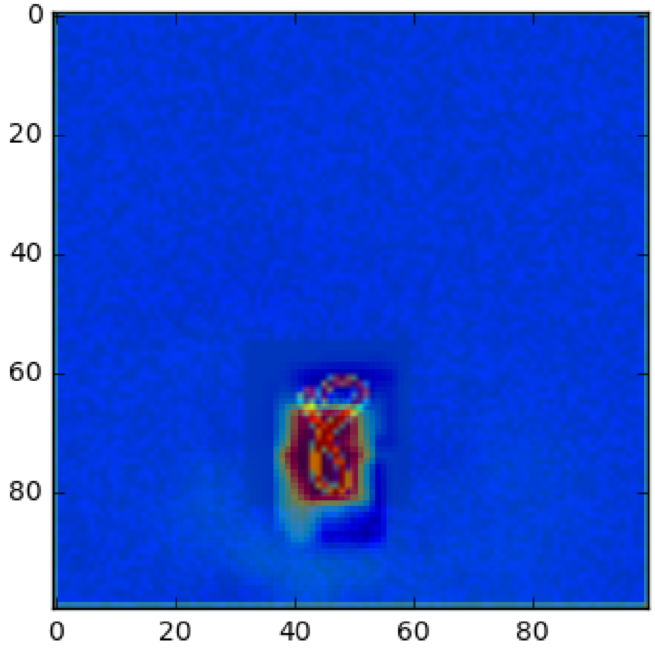
```
--------------------------------------------------------------------------------
prediction is : 7 with 9.760
```



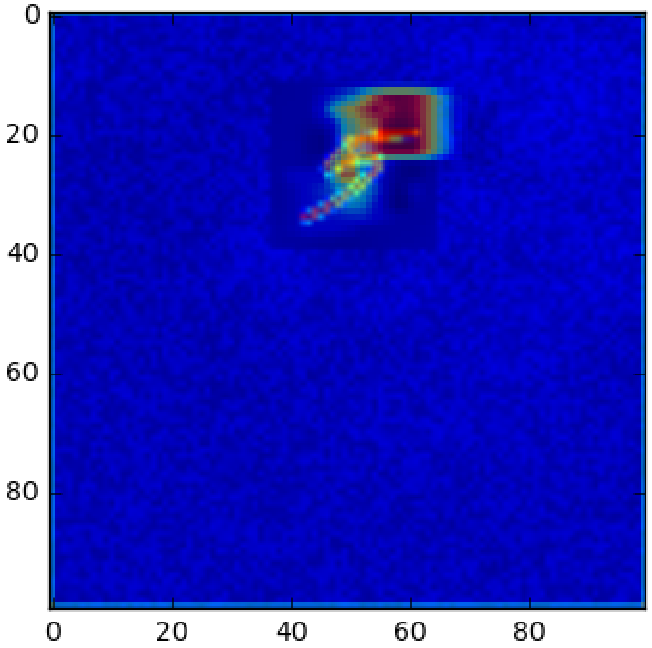
```
--------------------------------------------------------------------------------
prediction is : 0 with 9.685
```



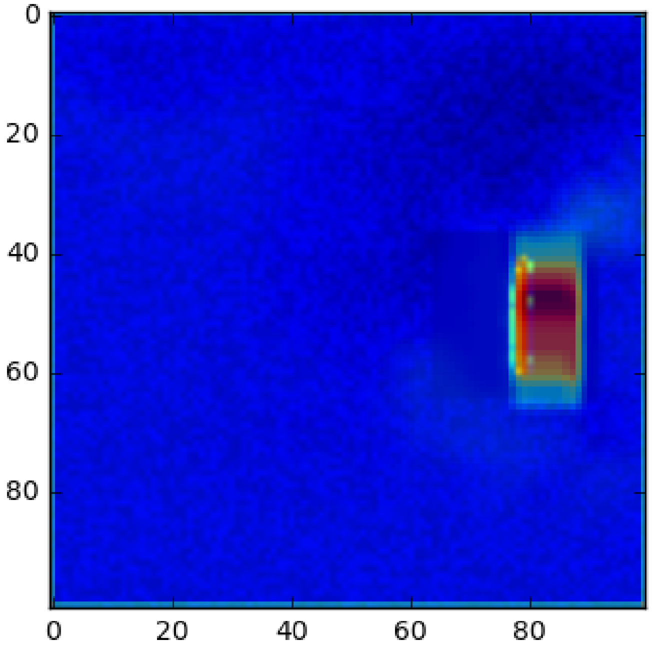
```
--------------------------------------------------------------------------------
prediction is : 8 with 7.898
```



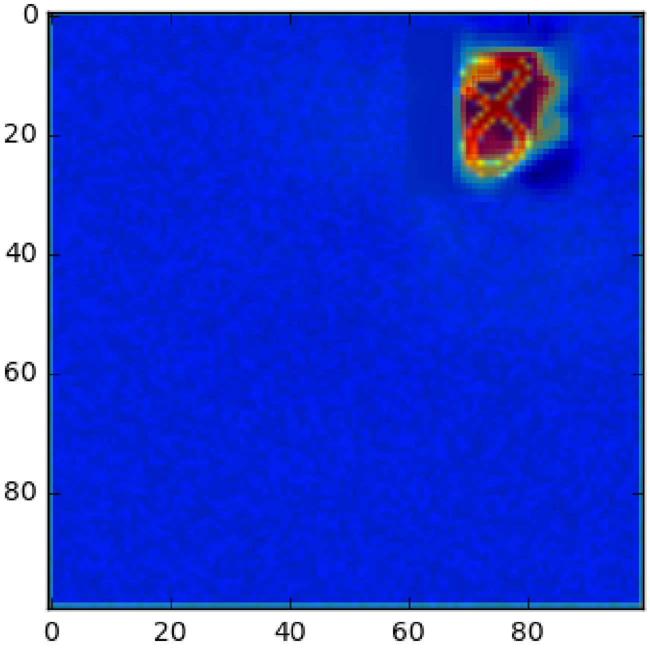
```
--------------------------------------------------------------------------------
prediction is : 5 with 8.021
```



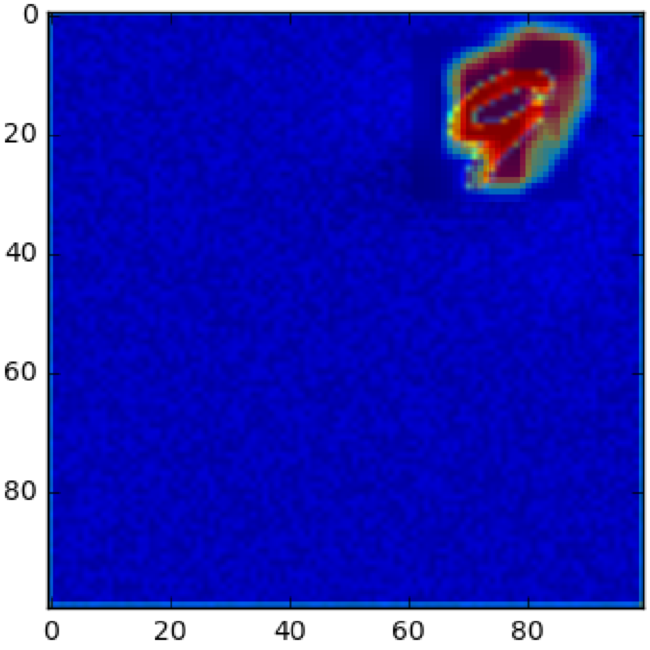
```
--------------------------------------------------------------------------------
prediction is : 1 with 6.352
```



```
--------------------------------------------------------------------------------
prediction is : 8 with 7.953
```



```
--------------------------------------------------------------------------------
prediction is : 9 with 6.643
```



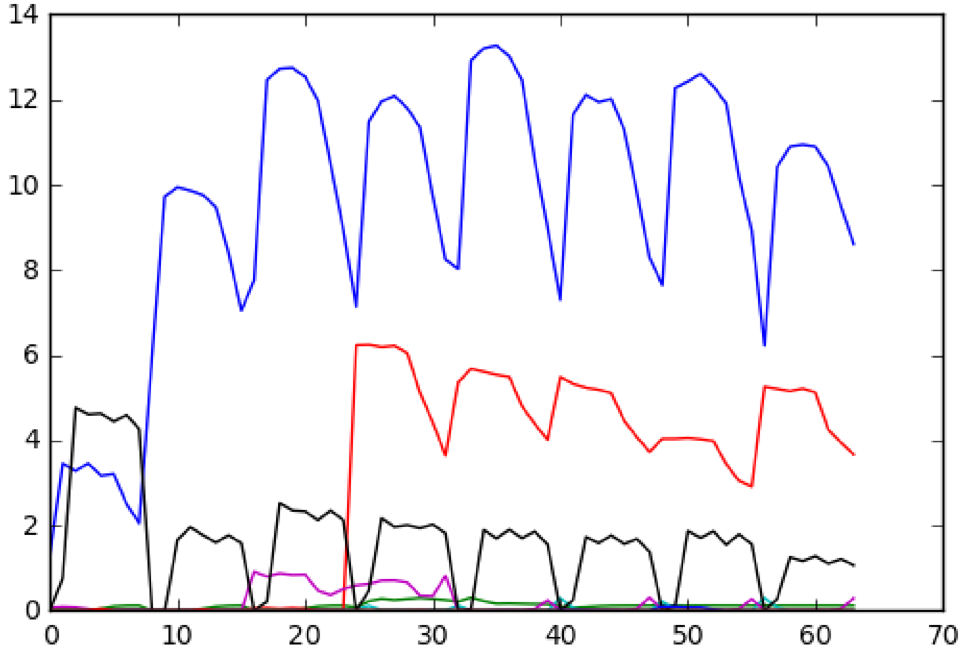
```
In [5]: accuracies = []
        for _ in range(100):
            digit,lbl  = utils.get_batch('test', 1, .1).next()
            digit      = digit[0].reshape(28,28)
            img        = np.random.random((100,100))*.1
            height     = img.shape[0]
            width      = img.shape[1]
            box_size   = 28
            step_size  = 10
            n_x_boxes  = (width -box_size)/step_size +1
            n_y_boxes  = (height-box_size)/step_size +1
            imgs       = np.tile(img, (n_x_boxes*n_y_boxes,1,1))

            for xx in range(0, n_x_boxes ):
                for yy in range(0, n_y_boxes ):
                    idx = xx*n_x_boxes+yy
                    x   = xx*step_size
                    y   = yy*step_size
                    imgs[idx, x:x+box_size, y:y+box_size ] = digit


            imgs  = imgs.reshape((-1,100,100,1))
            preds = simple_model.sess.run(simple_model.tf_out, feed_dict={simple_mo
            pred  = np.argmax(preds[-1])

            accuracy = sum(preds.argmax(axis=1) == lbl) / float(len(preds))
            accuracies.append(accuracy)

        print sum(accuracies)/len(accuracies)
```

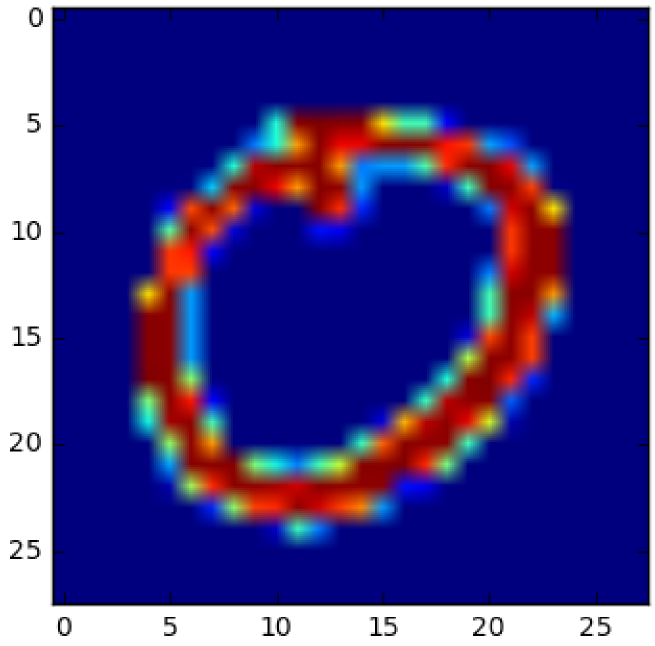
```
0.96546875
```

```
In [6]: digit,lbl  = utils.get_batch('test', 1, .1).next()
        digit      = digit[0].reshape(28,28)
        img        = np.random.random((100,100))*.1
        height     = img.shape[0]
        width      = img.shape[1]
        box_size   = 28
        step_size  = 10
        n_x_boxes  = (width -box_size)/step_size +1
        n_y_boxes  = (height-box_size)/step_size +1
        imgs       = np.tile(img, (n_x_boxes*n_y_boxes,1,1))

        for xx in range(0, n_x_boxes ):
```

```
    for yy in range(0, n_y_boxes ):
        idx = xx*n_x_boxes+yy
        x   = xx*step_size
        y   = yy*step_size
        imgs[idx, x:x+box_size, y:y+box_size ] = digit
lr_decay

imgs  = imgs.reshape((-1,100,100,1))
preds = simple_model.sess.run(simple_model.tf_out, feed_dict={simple_model.
pred  = np.argmax(preds[-1])

accuracy = sum(preds.argmax(axis=1) == lbl) / float(len(preds))

print preds.argmax(axis=1)
print accuracy*100
plt.plot(preds)
plt.show()

plt.imshow(digit)
plt.show()
```

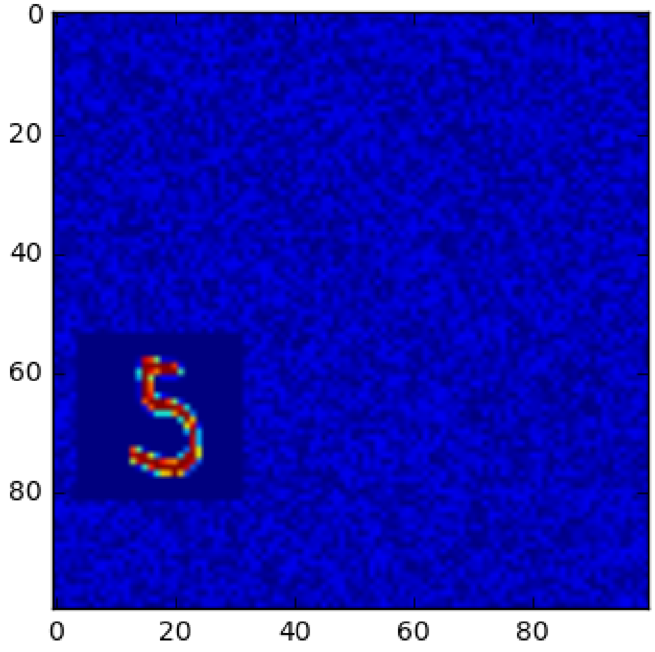
```
[0 0 6 6 6 6 6 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
90.625
```

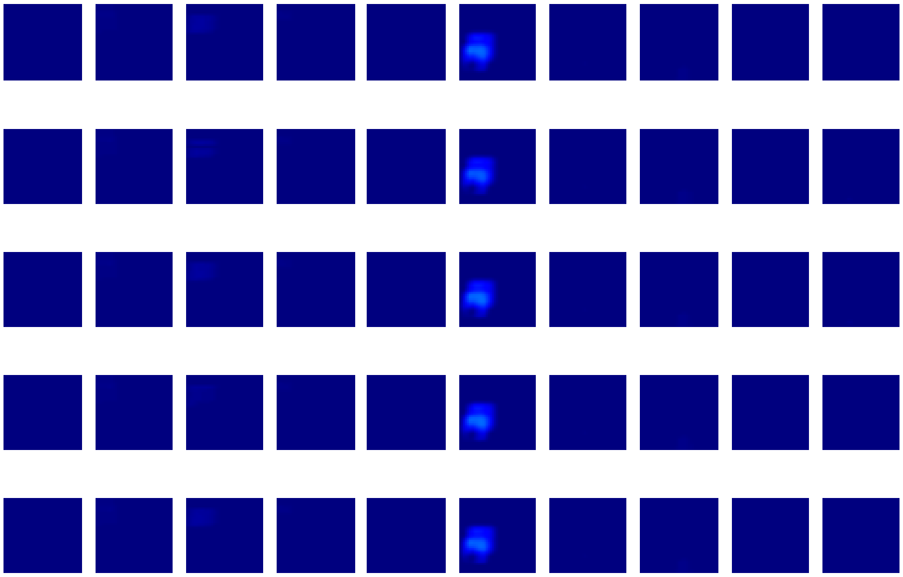




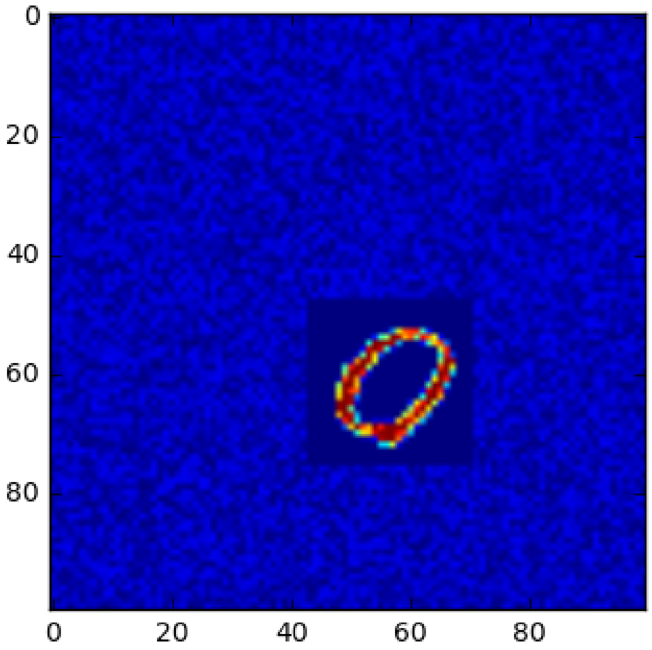
```
In [7]: for _ in range(10):
            simple_model.plot_classes_maps(100, n_CCN)
```

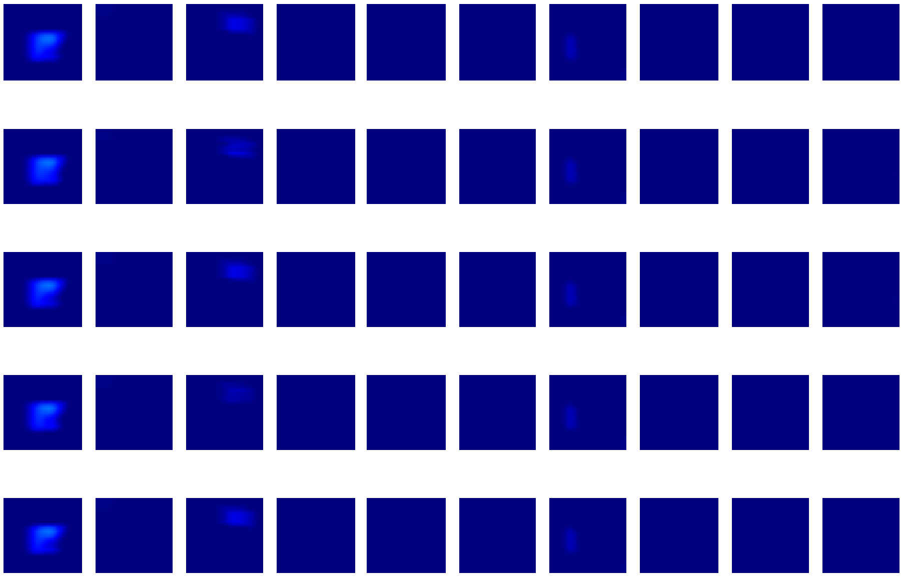
```
5
```



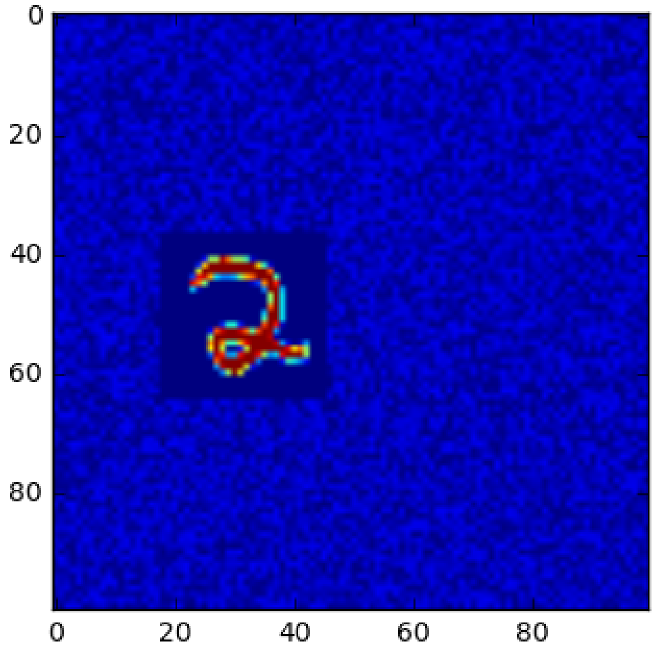


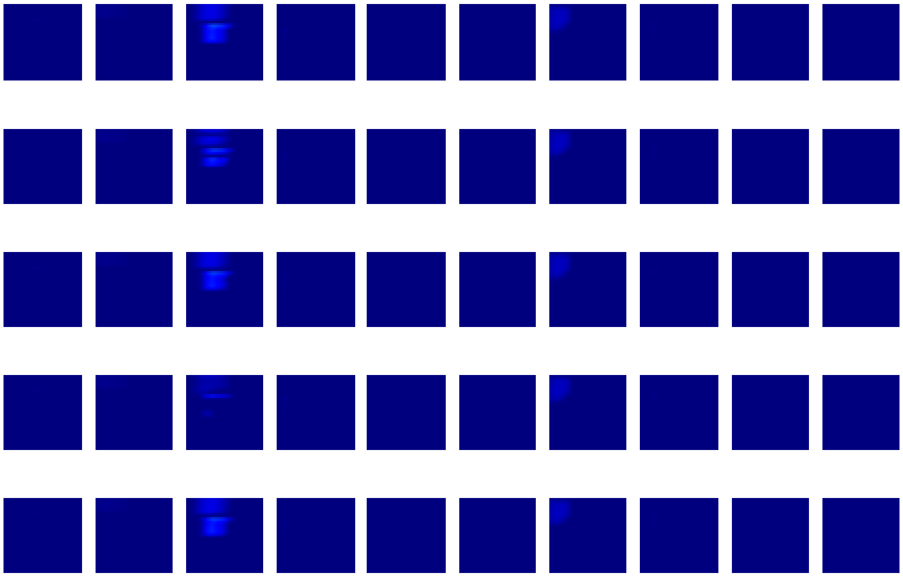
0



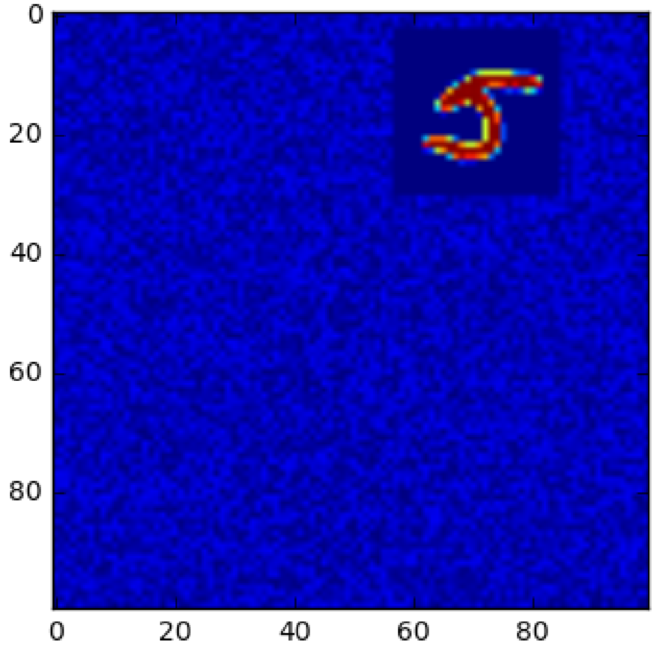


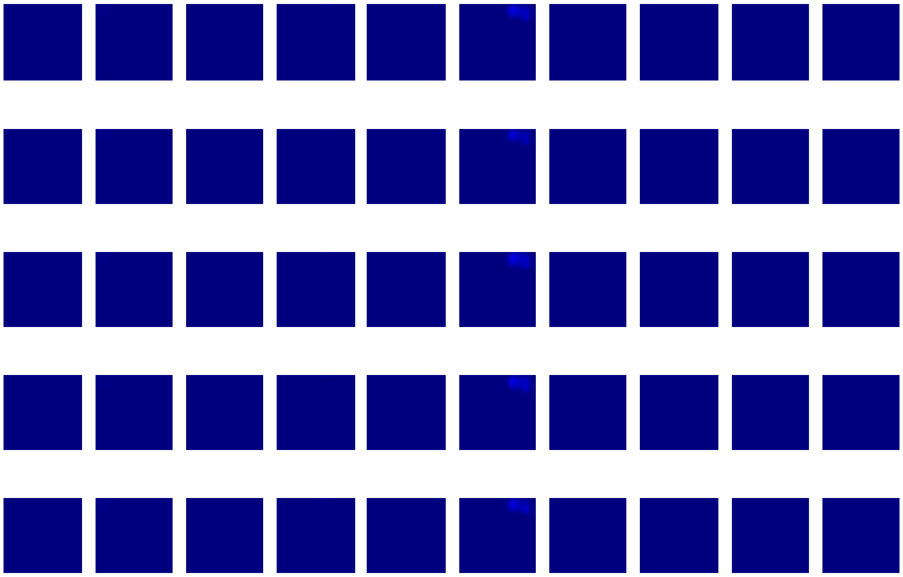
2



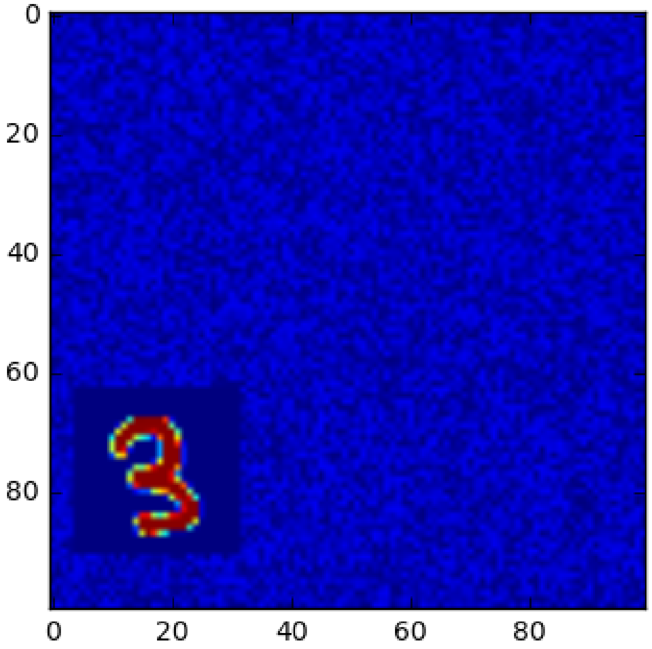


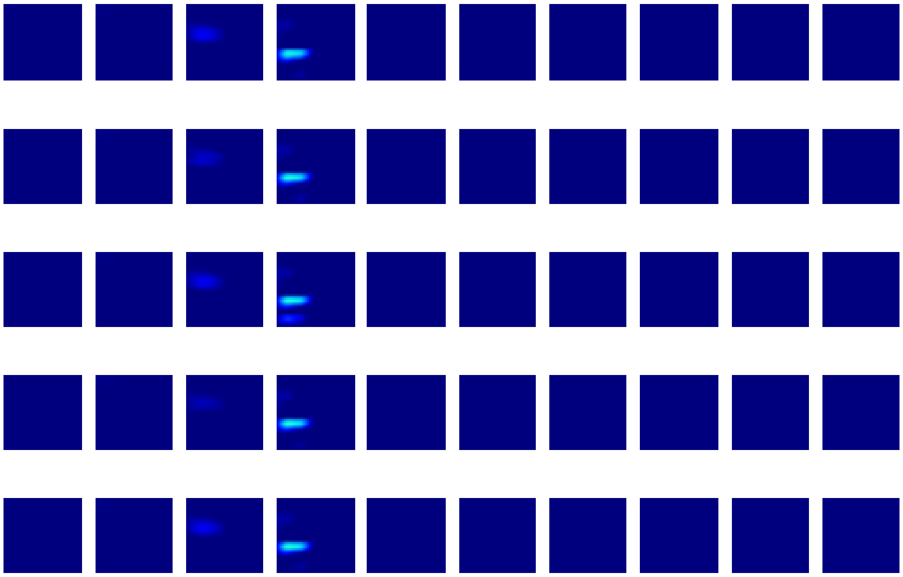
5



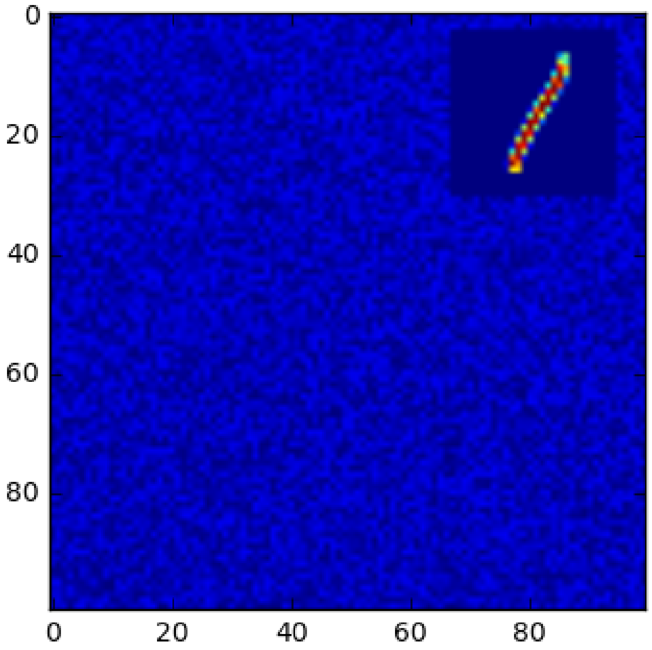


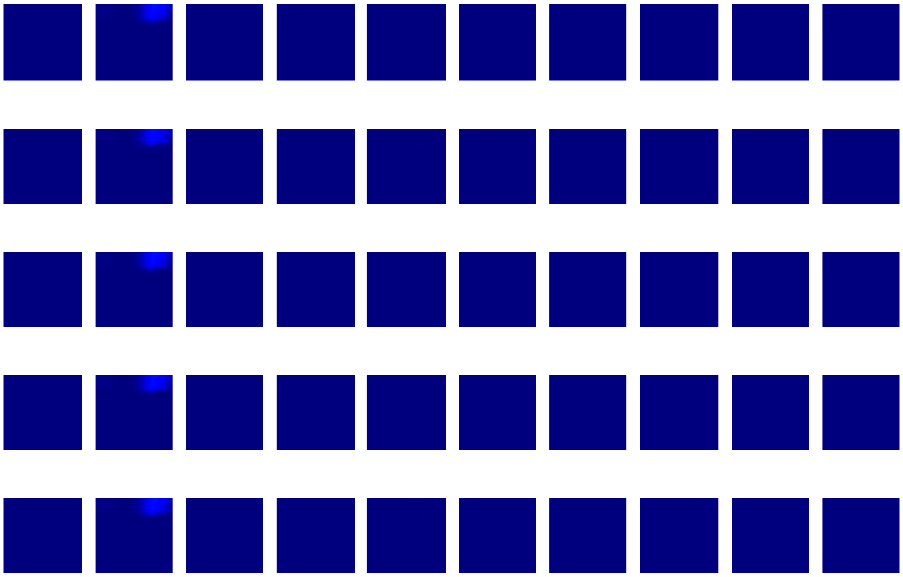
3



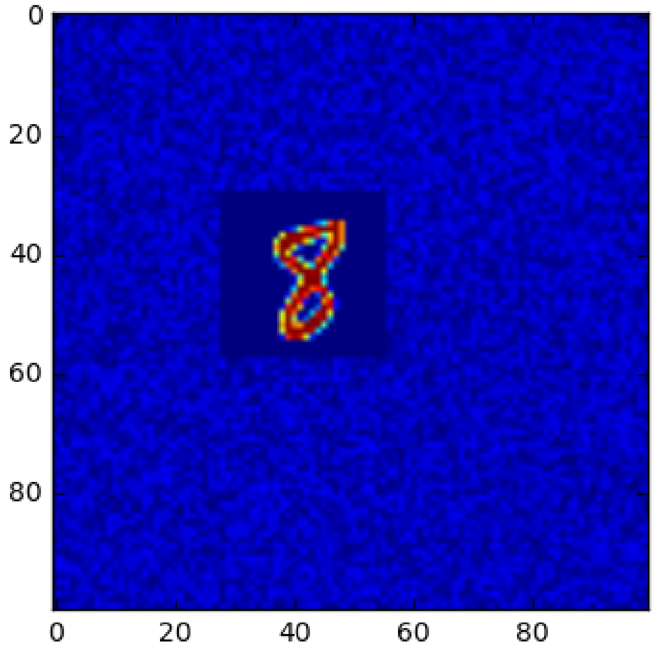


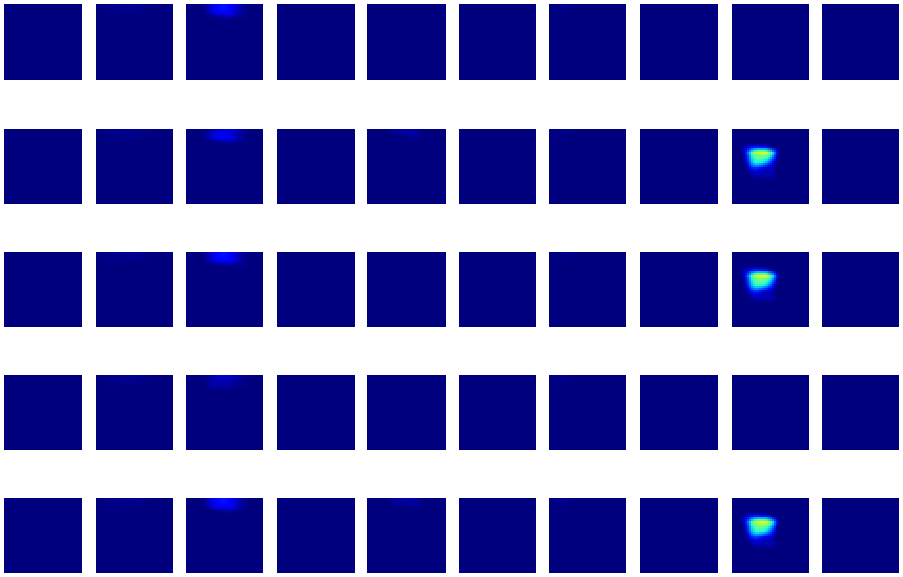
```
1
```



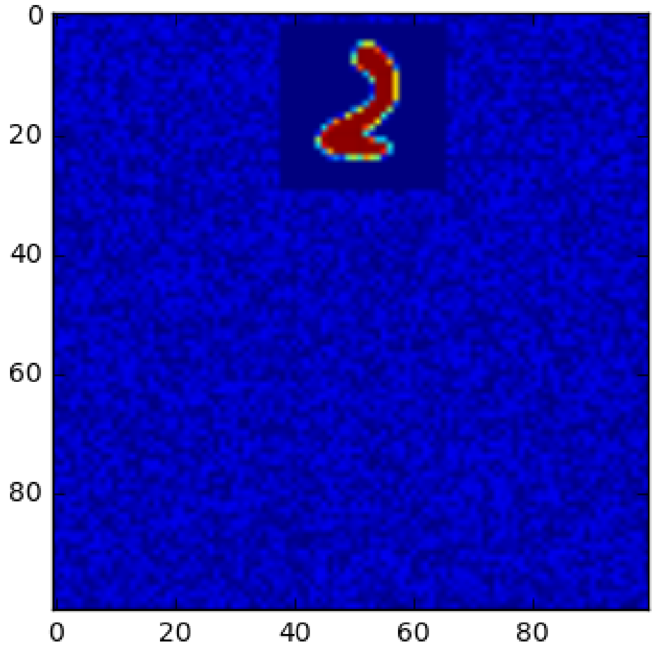


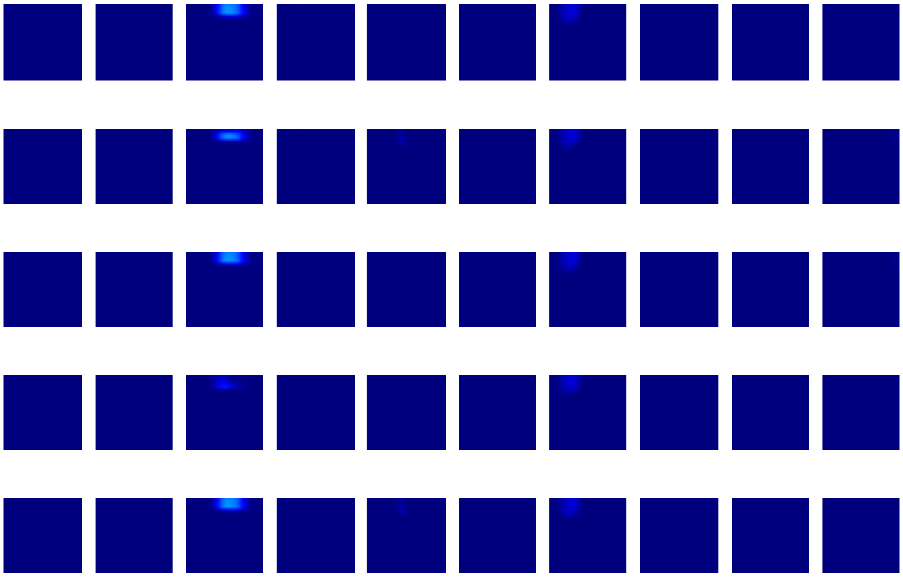
8



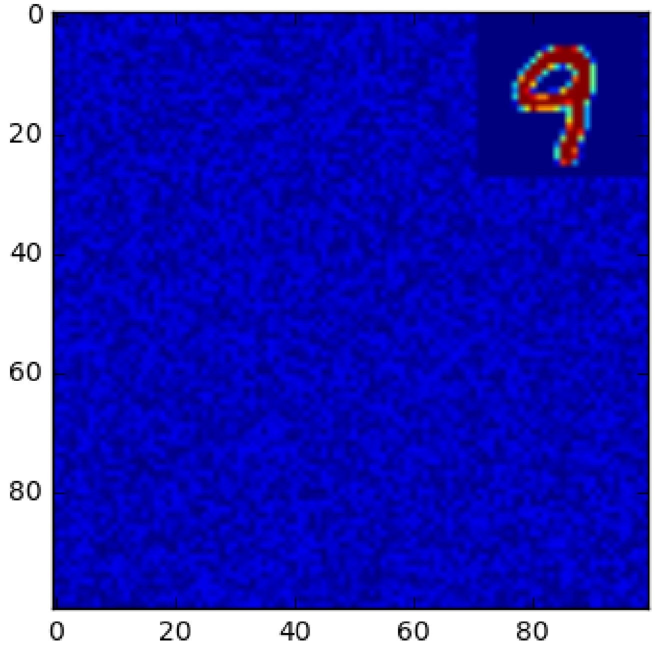


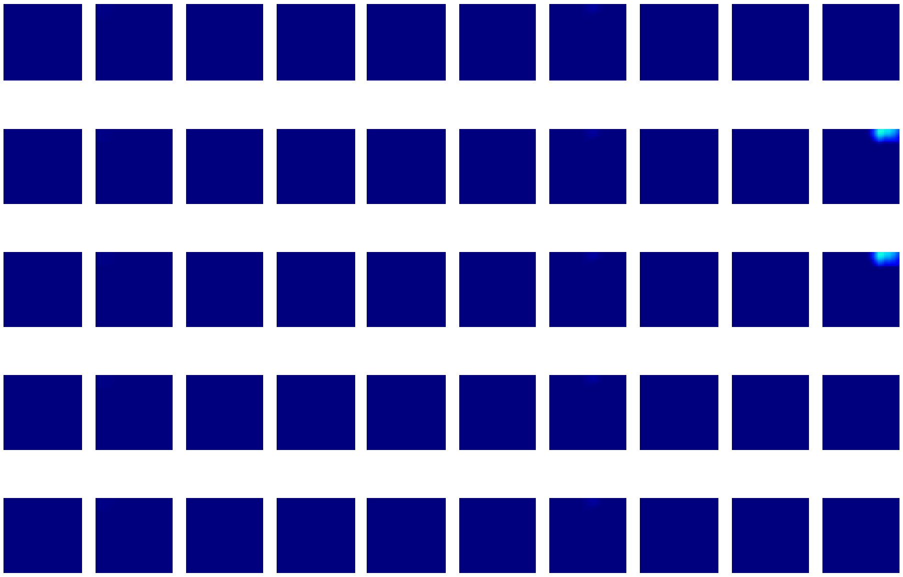
```
2
```



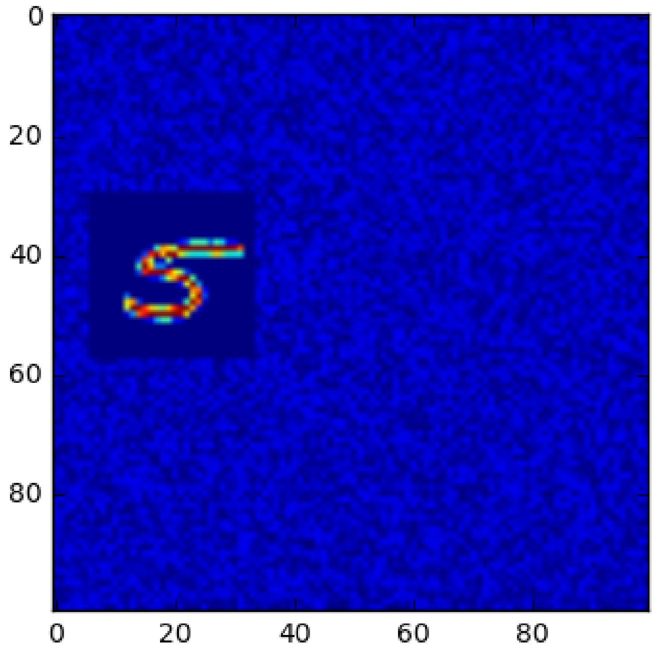


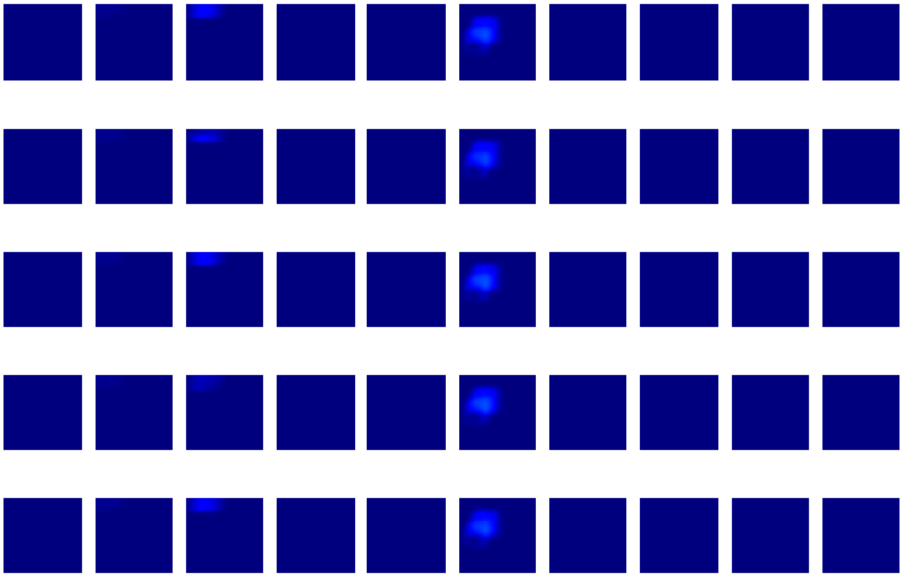
9





5





```
In [8]: simple_model=reload(simple_model)
```

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