

ILSVRC2012\_val\_00000067.JPG

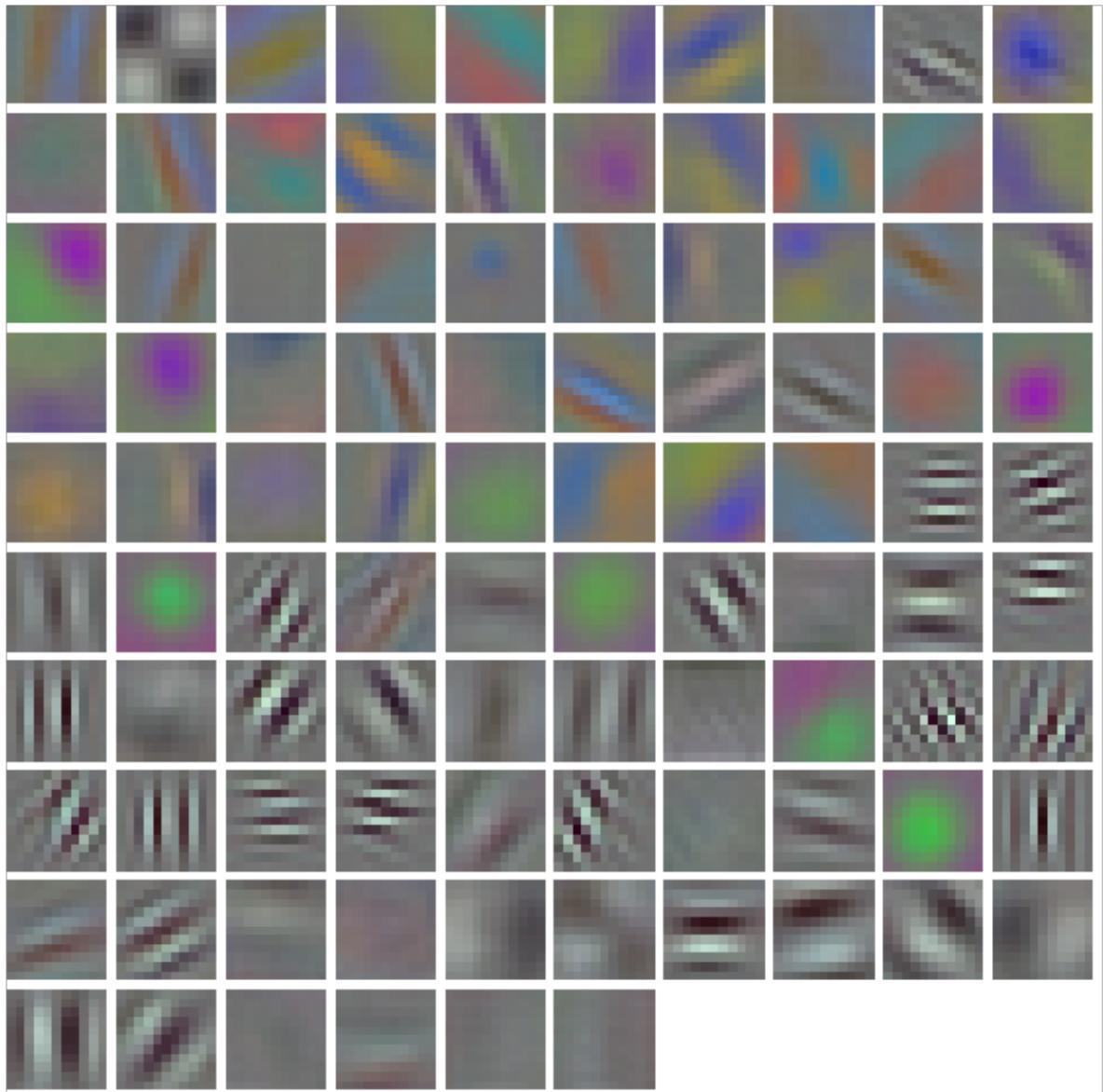
AlexNet:

```
Network data shape visualization
data    (10, 3, 227, 227)
conv1   (10, 96, 55, 55)
norm1   (10, 96, 55, 55)
pool1   (10, 96, 27, 27)
conv2   (10, 256, 27, 27)
norm2   (10, 256, 27, 27)
```

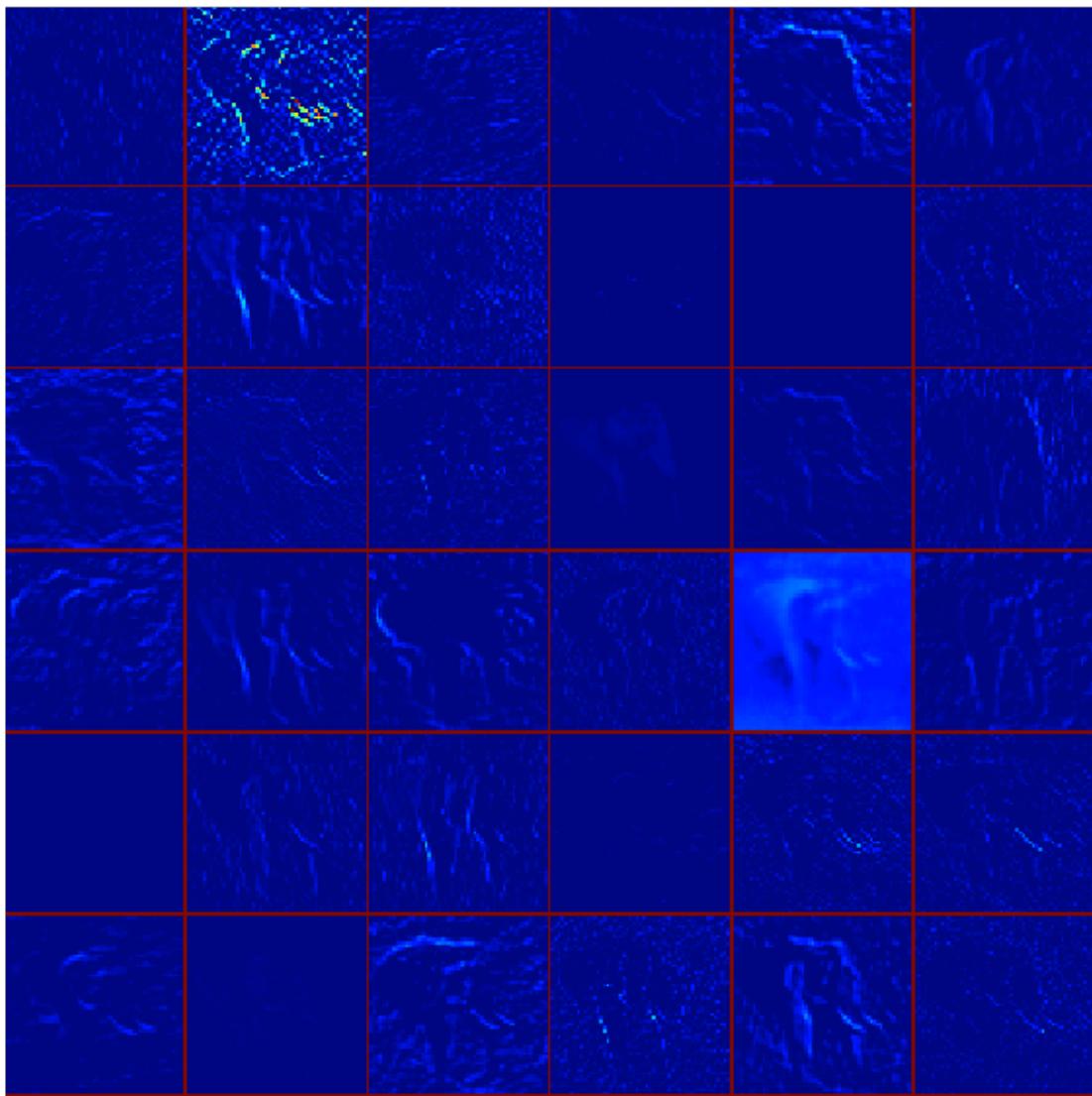
```
pool2  (10, 256, 13, 13)
conv3  (10, 384, 13, 13)
conv4  (10, 384, 13, 13)
conv5  (10, 256, 13, 13)
pool5  (10, 256, 6, 6)
fc6    (10, 4096)
fc7    (10, 4096)
fc8    (10, 1000)
prob   (10, 1000)

Network parameters shape visualization
conv1  (96, 3, 11, 11) (96,)
conv2  (256, 48, 5, 5) (256,)
conv3  (384, 256, 3, 3) (384,)
conv4  (384, 192, 3, 3) (384,)
conv5  (256, 192, 3, 3) (256,)
fc6    (4096, 9216) (4096,)
fc7    (4096, 4096) (4096,)
fc8    (1000, 4096) (1000,)
```

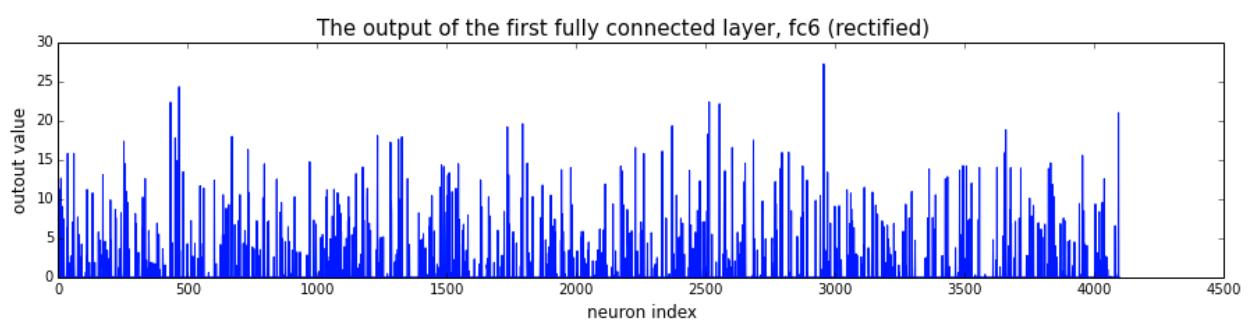
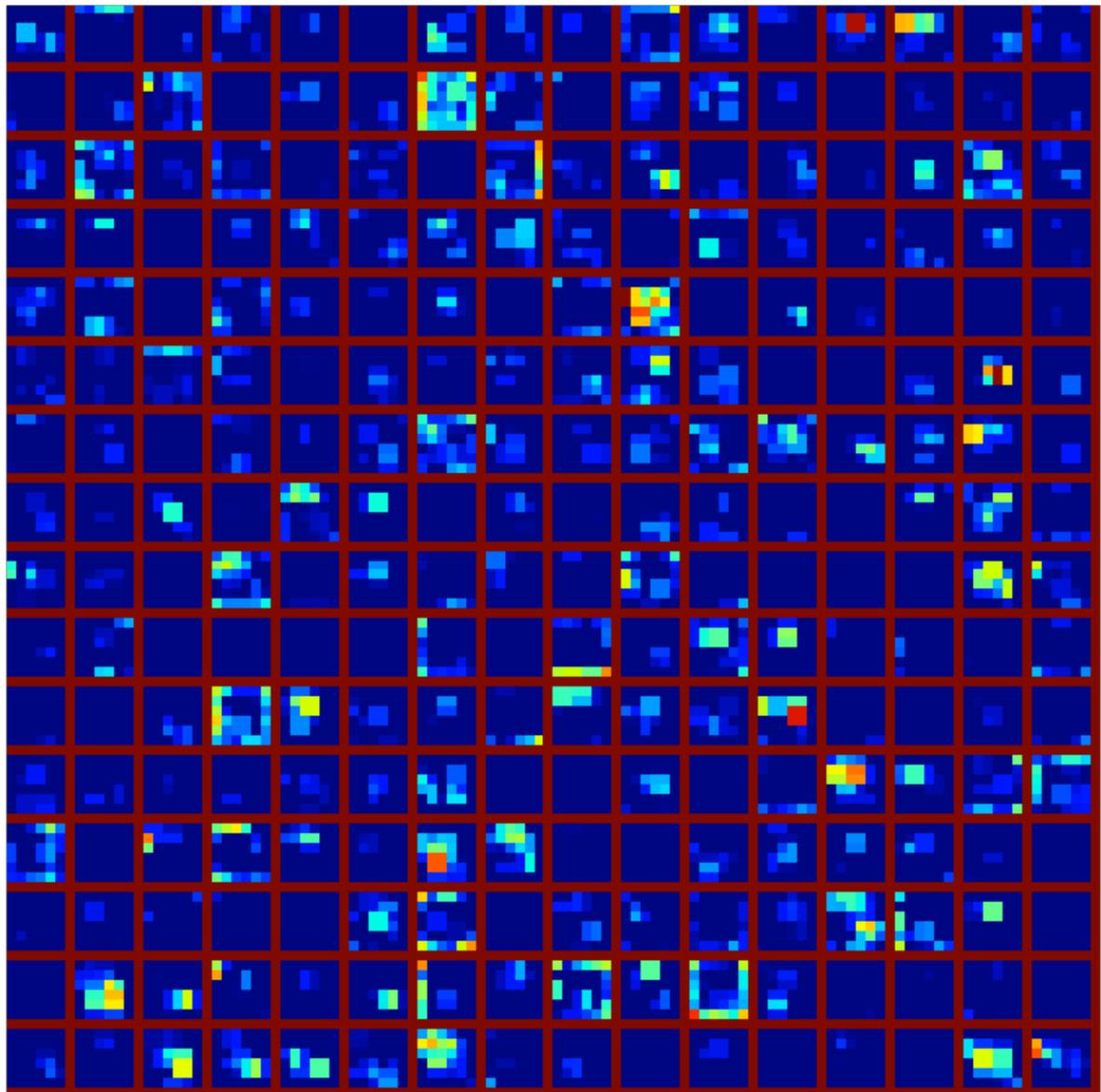
the parameters (weights) of the first conv Layer

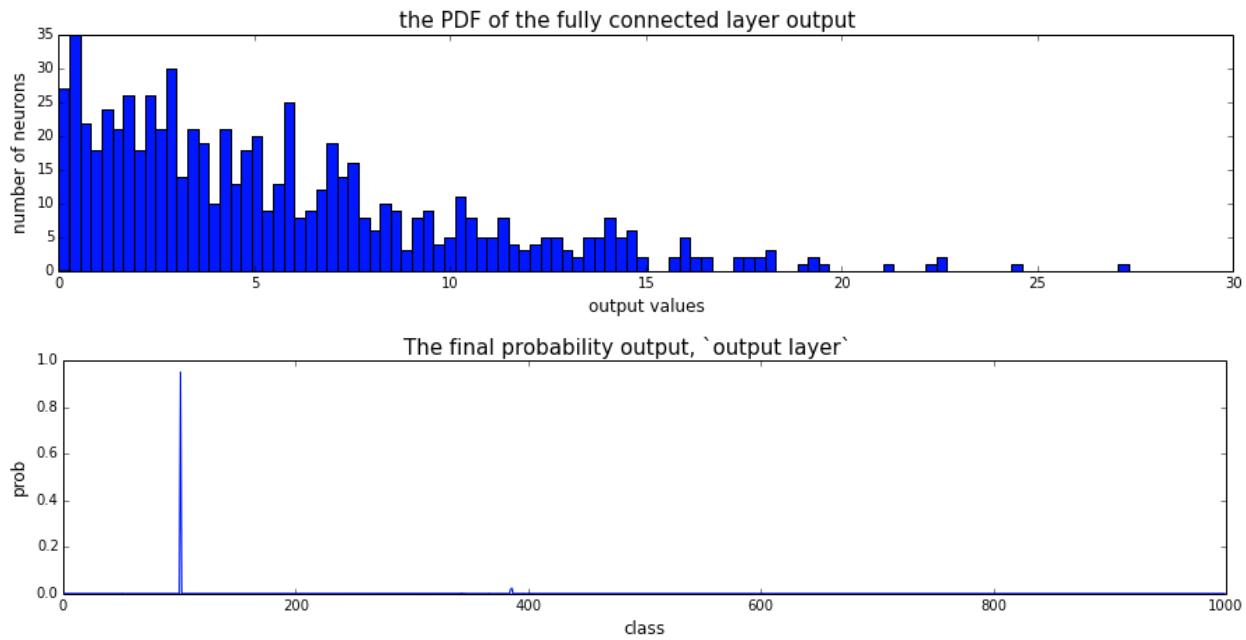


The first layer output, conv1 (rectified responses of the filters above, first 36 only)



The fifth layer after pooling, pool5



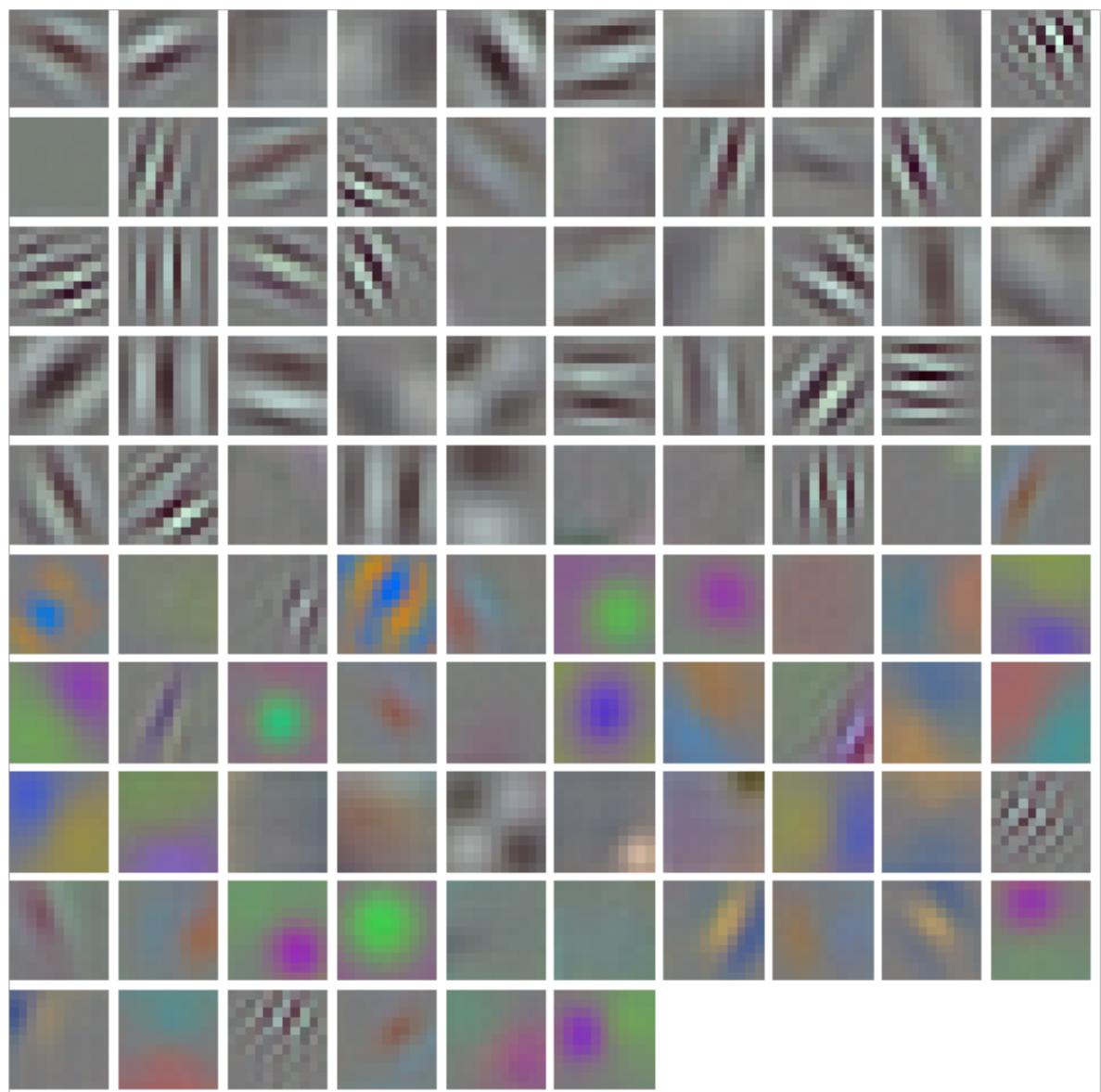


### CaffeNet:

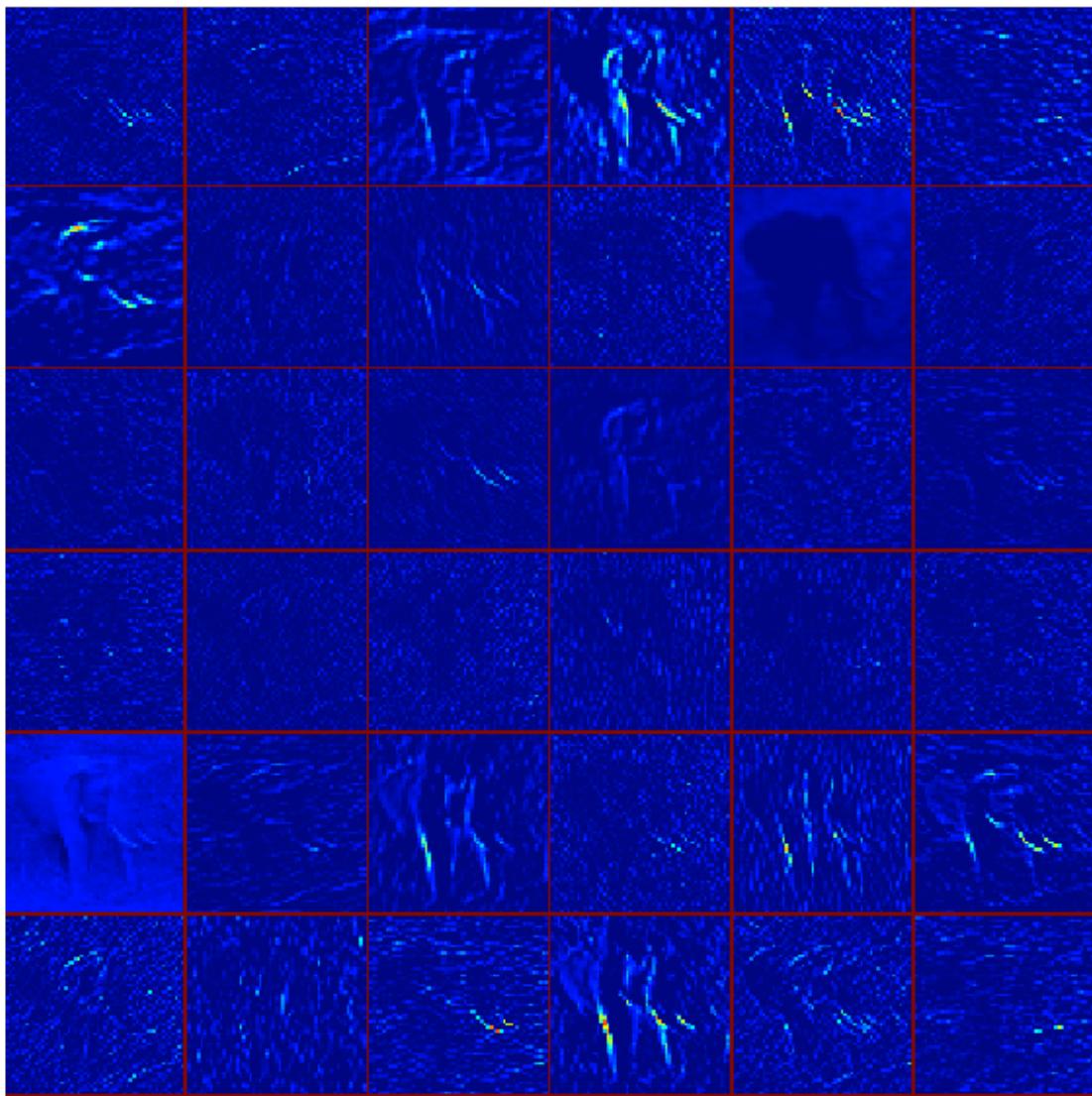
```
Network data shape visualization
data      (10, 3, 227, 227)
conv1    (10, 96, 55, 55)
pool1    (10, 96, 27, 27)
norm1    (10, 96, 27, 27)
conv2    (10, 256, 27, 27)
pool2    (10, 256, 13, 13)
norm2    (10, 256, 13, 13)
conv3    (10, 384, 13, 13)
conv4    (10, 384, 13, 13)
conv5    (10, 256, 13, 13)
pool5    (10, 256, 6, 6)
fc6      (10, 4096)
fc7      (10, 4096)
fc8      (10, 1000)
prob     (10, 1000)
```

```
Network parameters shape visualization
conv1    (96, 3, 11, 11) (96,)
conv2    (256, 48, 5, 5) (256,)
conv3    (384, 256, 3, 3) (384,)
conv4    (384, 192, 3, 3) (384,)
conv5    (256, 192, 3, 3) (256,)
fc6      (4096, 9216) (4096,)
fc7      (4096, 4096) (4096,)
fc8      (1000, 4096) (1000,)
```

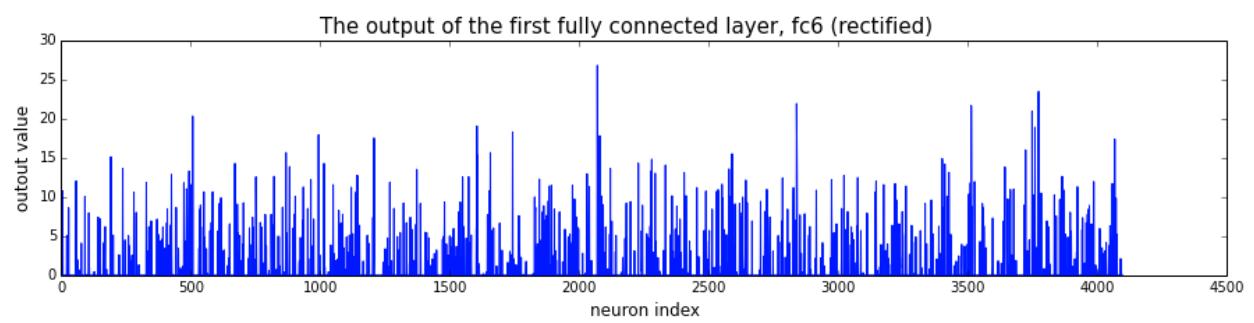
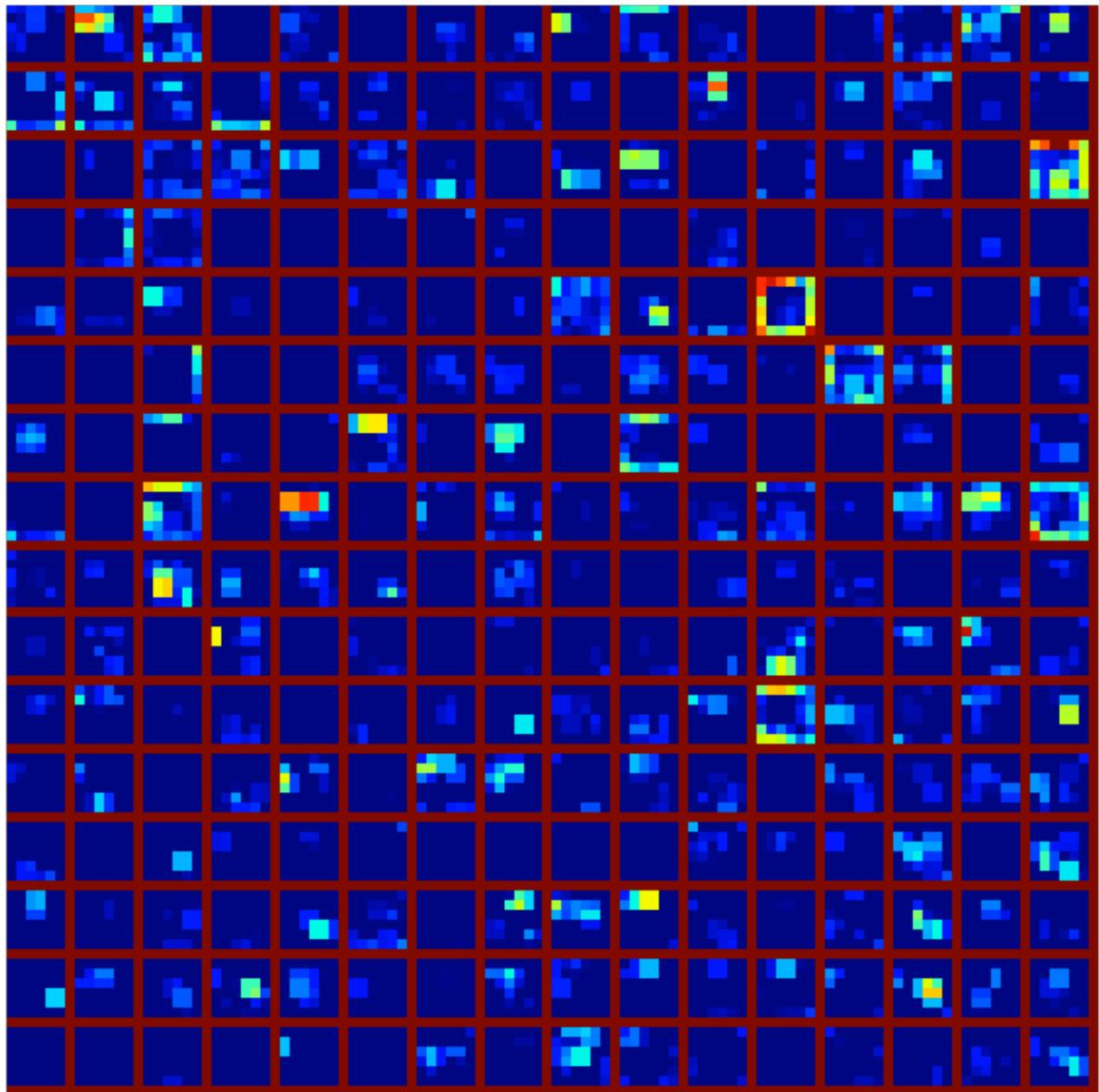
the parameters (weights) of the first conv Layer

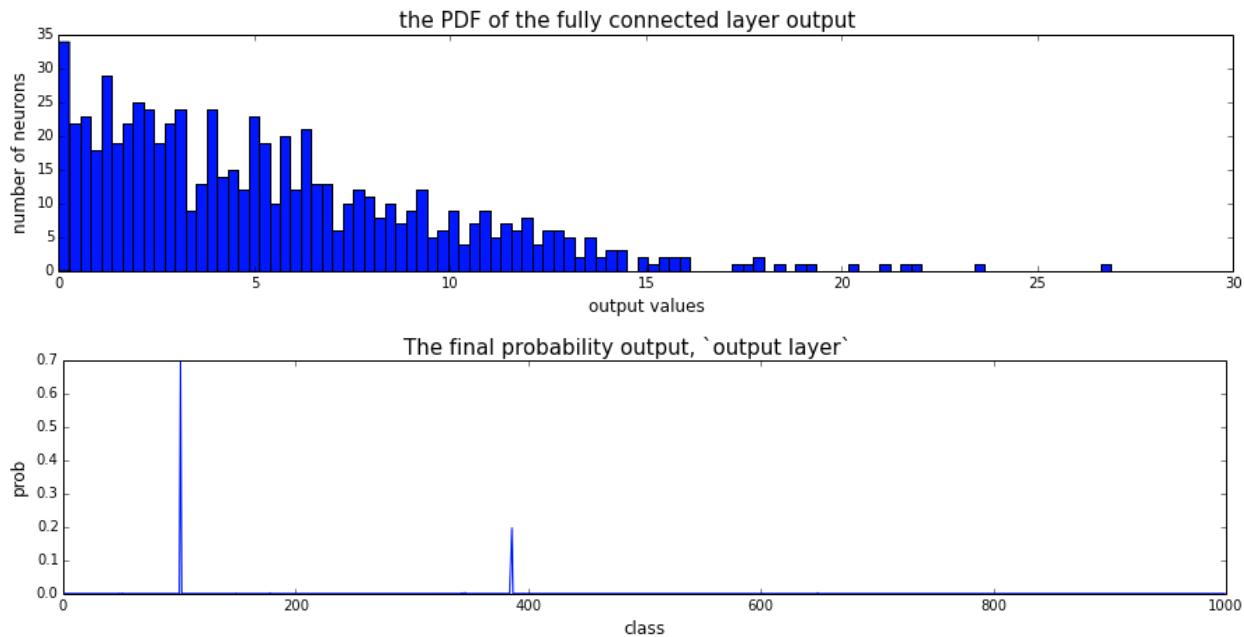


The first layer output, conv1 (rectified responses of the filters above, first 36 only)



The fifth layer after pooling, pool5





NIN:

Network data shape visualization

```

data      (10, 3, 224, 224)
conv1    (10, 96, 54, 54)
cccp1   (10, 96, 54, 54)
cccp2   (10, 96, 54, 54)
pool0   (10, 96, 27, 27)
conv2    (10, 256, 27, 27)
cccp3   (10, 256, 27, 27)
cccp4   (10, 256, 27, 27)
pool2   (10, 256, 13, 13)
conv3    (10, 384, 13, 13)
cccp5   (10, 384, 13, 13)
cccp6   (10, 384, 13, 13)
pool3   (10, 384, 6, 6)
conv4    (10, 1024, 6, 6)
cccp7   (10, 1024, 6, 6)
cccp8   (10, 1000, 6, 6)
pool4   (10, 1000, 1, 1)

```

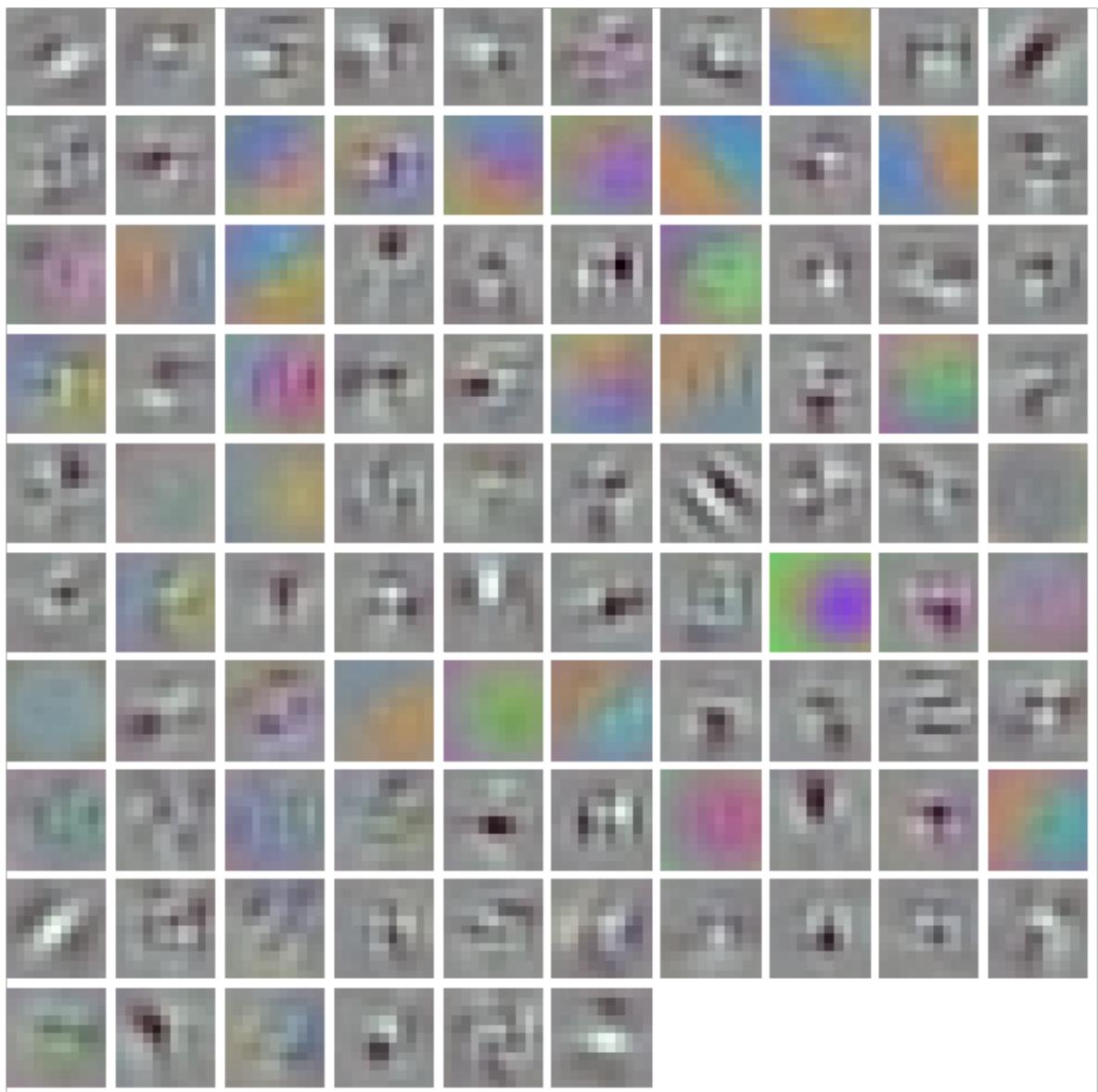
Network parameters shape visualization

```

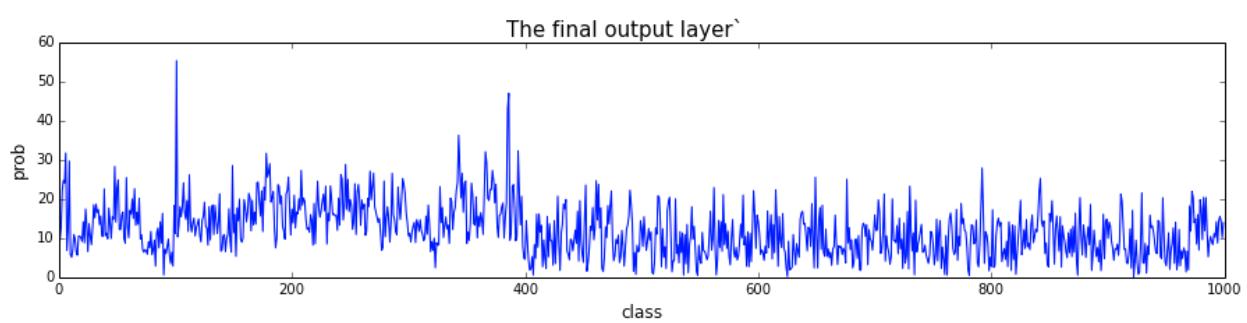
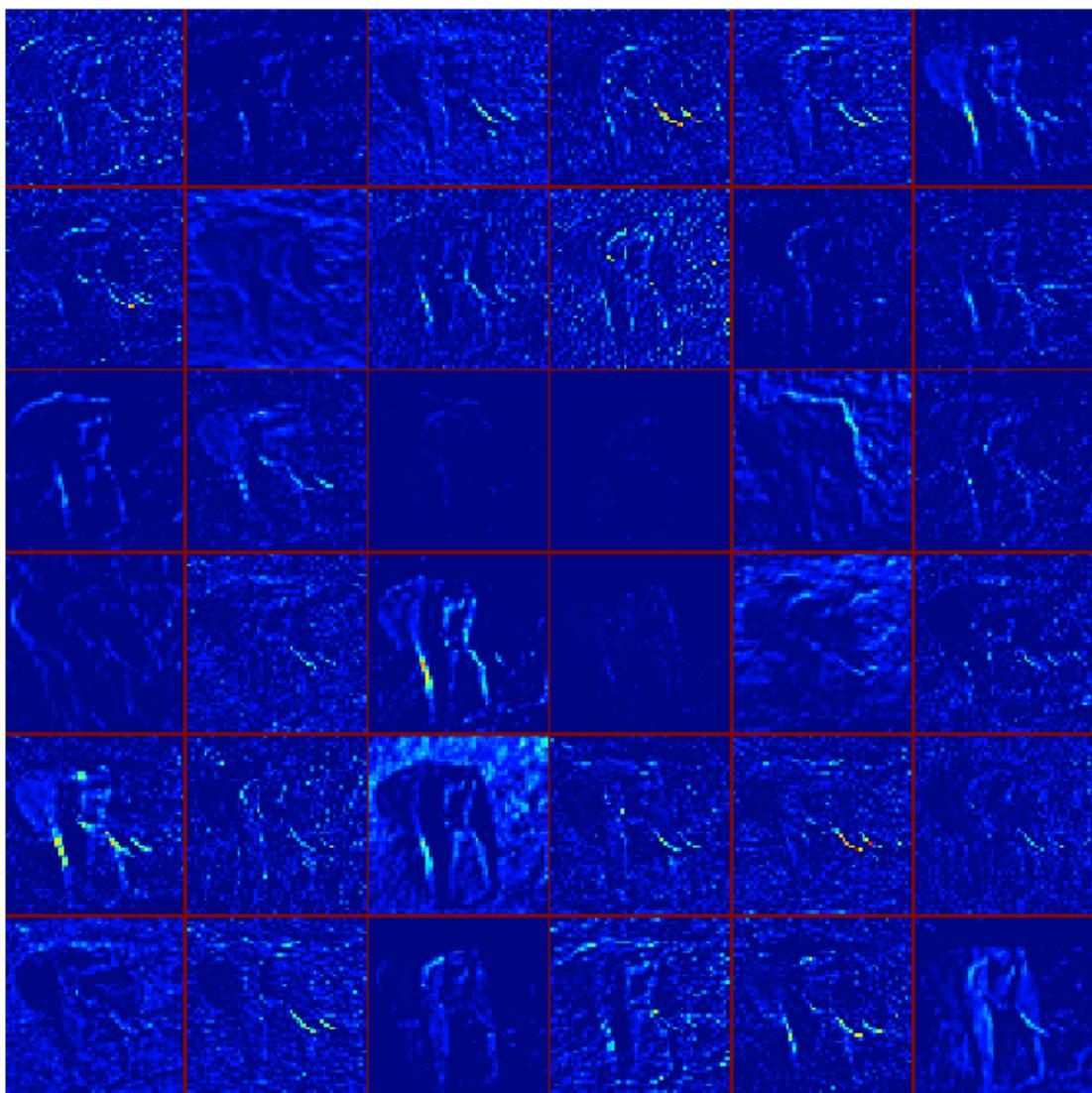
conv1    (96, 3, 11, 11) (96,)
cccp1   (96, 96, 1, 1) (96,)
cccp2   (96, 96, 1, 1) (96,)
conv2    (256, 96, 5, 5) (256,)
cccp3   (256, 256, 1, 1) (256,)
cccp4   (256, 256, 1, 1) (256,)
conv3    (384, 256, 3, 3) (384,)
cccp5   (384, 384, 1, 1) (384,)
cccp6   (384, 384, 1, 1) (384,)
conv4-1024   (1024, 384, 3, 3) (1024,)
cccp7-1024   (1024, 1024, 1, 1) (1024,)
cccp8-1024   (1000, 1024, 1, 1) (1000,)

```

the parameters (weights) of the first conv Layer



The first layer output, conv1 (rectified responses of the filters above, first 36 only)



GoogLeNet:  
Network data shape visualization  
data (10, 3, 224, 224)  
conv1/7x7\_s2 (10, 64, 112, 112)

```
pool1/3x3_s2      (10, 64, 56, 56)
pool1/norm1       (10, 64, 56, 56)
conv2/3x3_reduce   (10, 64, 56, 56)
conv2/3x3          (10, 192, 56, 56)
conv2/norm2        (10, 192, 56, 56)
pool2/3x3_s2      (10, 192, 28, 28)
pool2/3x3_s2_pool2/3x3_s2_0_split_0 (10, 192, 28, 28)
pool2/3x3_s2_pool2/3x3_s2_0_split_1 (10, 192, 28, 28)
pool2/3x3_s2_pool2/3x3_s2_0_split_2 (10, 192, 28, 28)
pool2/3x3_s2_pool2/3x3_s2_0_split_3 (10, 192, 28, 28)
inception_3a/1x1    (10, 64, 28, 28)
inception_3a/3x3_reduce   (10, 96, 28, 28)
inception_3a/3x3          (10, 128, 28, 28)
inception_3a/5x5_reduce   (10, 16, 28, 28)
inception_3a/5x5          (10, 32, 28, 28)
inception_3a/pool        (10, 192, 28, 28)
inception_3a/pool_proj    (10, 32, 28, 28)
inception_3a/output      (10, 256, 28, 28)
inception_3a/output_inception_3a/output_0_split_0 (10, 256, 28, 28)
inception_3a/output_inception_3a/output_0_split_1 (10, 256, 28, 28)
inception_3a/output_inception_3a/output_0_split_2 (10, 256, 28, 28)
inception_3a/output_inception_3a/output_0_split_3 (10, 256, 28, 28)
inception_3b/1x1      (10, 128, 28, 28)
inception_3b/3x3_reduce   (10, 128, 28, 28)
inception_3b/3x3          (10, 192, 28, 28)
inception_3b/5x5_reduce   (10, 32, 28, 28)
inception_3b/5x5          (10, 96, 28, 28)
inception_3b/pool        (10, 256, 28, 28)
inception_3b/pool_proj    (10, 64, 28, 28)
inception_3b/output      (10, 480, 28, 28)
pool3/3x3_s2      (10, 480, 14, 14)
pool3/3x3_s2_pool3/3x3_s2_0_split_0 (10, 480, 14, 14)
pool3/3x3_s2_pool3/3x3_s2_0_split_1 (10, 480, 14, 14)
pool3/3x3_s2_pool3/3x3_s2_0_split_2 (10, 480, 14, 14)
pool3/3x3_s2_pool3/3x3_s2_0_split_3 (10, 480, 14, 14)
inception_4a/1x1      (10, 192, 14, 14)
inception_4a/3x3_reduce   (10, 96, 14, 14)
inception_4a/3x3          (10, 208, 14, 14)
inception_4a/5x5_reduce   (10, 16, 14, 14)
inception_4a/5x5          (10, 48, 14, 14)
inception_4a/pool        (10, 480, 14, 14)
inception_4a/pool_proj    (10, 64, 14, 14)
inception_4a/output      (10, 512, 14, 14)
inception_4a/output_inception_4a/output_0_split_0 (10, 512, 14, 14)
inception_4a/output_inception_4a/output_0_split_1 (10, 512, 14, 14)
inception_4a/output_inception_4a/output_0_split_2 (10, 512, 14, 14)
inception_4a/output_inception_4a/output_0_split_3 (10, 512, 14, 14)
inception_4b/1x1      (10, 160, 14, 14)
inception_4b/3x3_reduce   (10, 112, 14, 14)
inception_4b/3x3          (10, 224, 14, 14)
inception_4b/5x5_reduce   (10, 24, 14, 14)
inception_4b/5x5          (10, 64, 14, 14)
inception_4b/pool        (10, 512, 14, 14)
inception_4b/pool_proj    (10, 64, 14, 14)
inception_4b/output      (10, 512, 14, 14)
```

```
inception_4b/output_inception_4b/output_0_split_0 (10, 512, 14, 14)
inception_4b/output_inception_4b/output_0_split_1 (10, 512, 14, 14)
inception_4b/output_inception_4b/output_0_split_2 (10, 512, 14, 14)
inception_4b/output_inception_4b/output_0_split_3 (10, 512, 14, 14)
inception_4c/1x1 (10, 128, 14, 14)
inception_4c/3x3_reduce (10, 128, 14, 14)
inception_4c/3x3 (10, 256, 14, 14)
inception_4c/5x5_reduce (10, 24, 14, 14)
inception_4c/5x5 (10, 64, 14, 14)
inception_4c/pool (10, 512, 14, 14)
inception_4c/pool_proj (10, 64, 14, 14)
inception_4c/output (10, 512, 14, 14)
inception_4c/output_inception_4c/output_0_split_0 (10, 512, 14, 14)
inception_4c/output_inception_4c/output_0_split_1 (10, 512, 14, 14)
inception_4c/output_inception_4c/output_0_split_2 (10, 512, 14, 14)
inception_4c/output_inception_4c/output_0_split_3 (10, 512, 14, 14)
inception_4d/1x1 (10, 112, 14, 14)
inception_4d/3x3_reduce (10, 144, 14, 14)
inception_4d/3x3 (10, 288, 14, 14)
inception_4d/5x5_reduce (10, 32, 14, 14)
inception_4d/5x5 (10, 64, 14, 14)
inception_4d/pool (10, 512, 14, 14)
inception_4d/pool_proj (10, 64, 14, 14)
inception_4d/output (10, 528, 14, 14)
inception_4d/output_inception_4d/output_0_split_0 (10, 528, 14, 14)
inception_4d/output_inception_4d/output_0_split_1 (10, 528, 14, 14)
inception_4d/output_inception_4d/output_0_split_2 (10, 528, 14, 14)
inception_4d/output_inception_4d/output_0_split_3 (10, 528, 14, 14)
inception_4e/1x1 (10, 256, 14, 14)
inception_4e/3x3_reduce (10, 160, 14, 14)
inception_4e/3x3 (10, 320, 14, 14)
inception_4e/5x5_reduce (10, 32, 14, 14)
inception_4e/5x5 (10, 128, 14, 14)
inception_4e/pool (10, 528, 14, 14)
inception_4e/pool_proj (10, 128, 14, 14)
inception_4e/output (10, 832, 14, 14)
pool4/3x3_s2 (10, 832, 7, 7)
pool4/3x3_s2_pool4/3x3_s2_0_split_0 (10, 832, 7, 7)
pool4/3x3_s2_pool4/3x3_s2_0_split_1 (10, 832, 7, 7)
pool4/3x3_s2_pool4/3x3_s2_0_split_2 (10, 832, 7, 7)
pool4/3x3_s2_pool4/3x3_s2_0_split_3 (10, 832, 7, 7)
inception_5a/1x1 (10, 256, 7, 7)
inception_5a/3x3_reduce (10, 160, 7, 7)
inception_5a/3x3 (10, 320, 7, 7)
inception_5a/5x5_reduce (10, 32, 7, 7)
inception_5a/5x5 (10, 128, 7, 7)
inception_5a/pool (10, 832, 7, 7)
inception_5a/pool_proj (10, 128, 7, 7)
inception_5a/output (10, 832, 7, 7)
inception_5a/output_inception_5a/output_0_split_0 (10, 832, 7, 7)
inception_5a/output_inception_5a/output_0_split_1 (10, 832, 7, 7)
inception_5a/output_inception_5a/output_0_split_2 (10, 832, 7, 7)
inception_5a/output_inception_5a/output_0_split_3 (10, 832, 7, 7)
inception_5b/1x1 (10, 384, 7, 7)
inception_5b/3x3_reduce (10, 192, 7, 7)
```

```

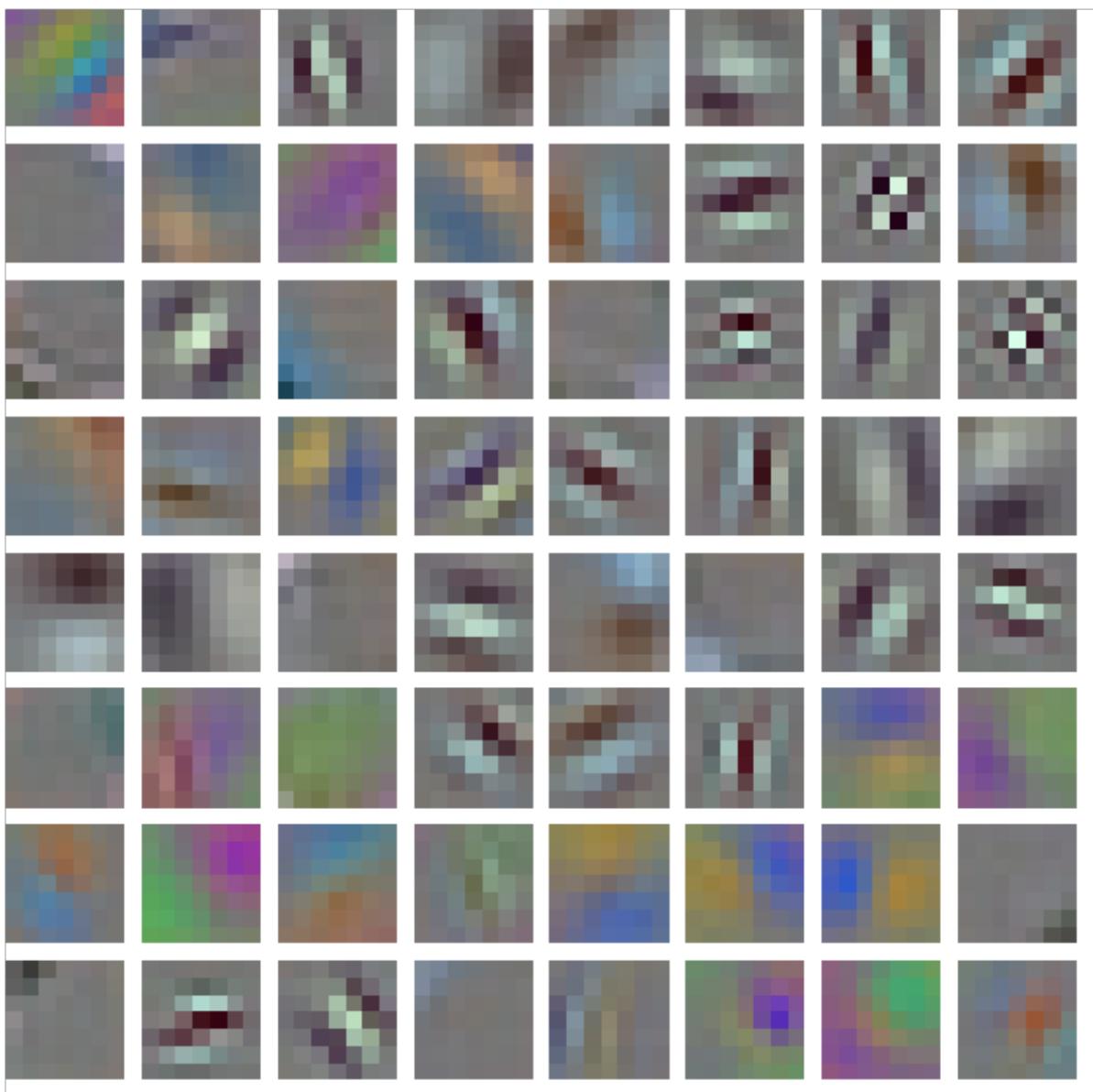
inception_5b/3x3      (10, 384, 7, 7)
inception_5b/5x5_reduce (10, 48, 7, 7)
inception_5b/5x5      (10, 128, 7, 7)
inception_5b/pool     (10, 832, 7, 7)
inception_5b/pool_proj (10, 128, 7, 7)
inception_5b/output   (10, 1024, 7, 7)
pool5/7x7_s1          (10, 1024, 1, 1)
loss3/classifier      (10, 1000)
prob                 (10, 1000)

Network parameters shape visualization
conv1/7x7_s2          (64, 3, 7, 7) (64,)
conv2/3x3_reduce       (64, 64, 1, 1) (64, )
conv2/3x3              (192, 64, 3, 3) (192, )
inception_3a/1x1       (64, 192, 1, 1) (64, )
inception_3a/3x3_reduce (96, 192, 1, 1) (96, )
inception_3a/3x3       (128, 96, 3, 3) (128, )
inception_3a/5x5_reduce (16, 192, 1, 1) (16, )
inception_3a/5x5       (32, 16, 5, 5) (32, )
inception_3a/pool_proj (32, 192, 1, 1) (32, )
inception_3b/1x1       (128, 256, 1, 1) (128, )
inception_3b/3x3_reduce (128, 256, 1, 1) (128, )
inception_3b/3x3       (192, 128, 3, 3) (192, )
inception_3b/5x5_reduce (32, 256, 1, 1) (32, )
inception_3b/5x5       (96, 32, 5, 5) (96, )
inception_3b/pool_proj (64, 256, 1, 1) (64, )
inception_4a/1x1       (192, 480, 1, 1) (192, )
inception_4a/3x3_reduce (96, 480, 1, 1) (96, )
inception_4a/3x3       (208, 96, 3, 3) (208, )
inception_4a/5x5_reduce (16, 480, 1, 1) (16, )
inception_4a/5x5       (48, 16, 5, 5) (48, )
inception_4a/pool_proj (64, 480, 1, 1) (64, )
inception_4b/1x1       (160, 512, 1, 1) (160, )
inception_4b/3x3_reduce (112, 512, 1, 1) (112, )
inception_4b/3x3       (224, 112, 3, 3) (224, )
inception_4b/5x5_reduce (24, 512, 1, 1) (24, )
inception_4b/5x5       (64, 24, 5, 5) (64, )
inception_4b/pool_proj (64, 512, 1, 1) (64, )
inception_4c/1x1       (128, 512, 1, 1) (128, )
inception_4c/3x3_reduce (128, 512, 1, 1) (128, )
inception_4c/3x3       (256, 128, 3, 3) (256, )
inception_4c/5x5_reduce (24, 512, 1, 1) (24, )
inception_4c/5x5       (64, 24, 5, 5) (64, )
inception_4c/pool_proj (64, 512, 1, 1) (64, )
inception_4d/1x1       (112, 512, 1, 1) (112, )
inception_4d/3x3_reduce (144, 512, 1, 1) (144, )
inception_4d/3x3       (288, 144, 3, 3) (288, )
inception_4d/5x5_reduce (32, 512, 1, 1) (32, )
inception_4d/5x5       (64, 32, 5, 5) (64, )
inception_4d/pool_proj (64, 512, 1, 1) (64, )
inception_4e/1x1       (256, 528, 1, 1) (256, )
inception_4e/3x3_reduce (160, 528, 1, 1) (160, )
inception_4e/3x3       (320, 160, 3, 3) (320, )
inception_4e/5x5_reduce (32, 528, 1, 1) (32, )
inception_4e/5x5       (128, 32, 5, 5) (128, )

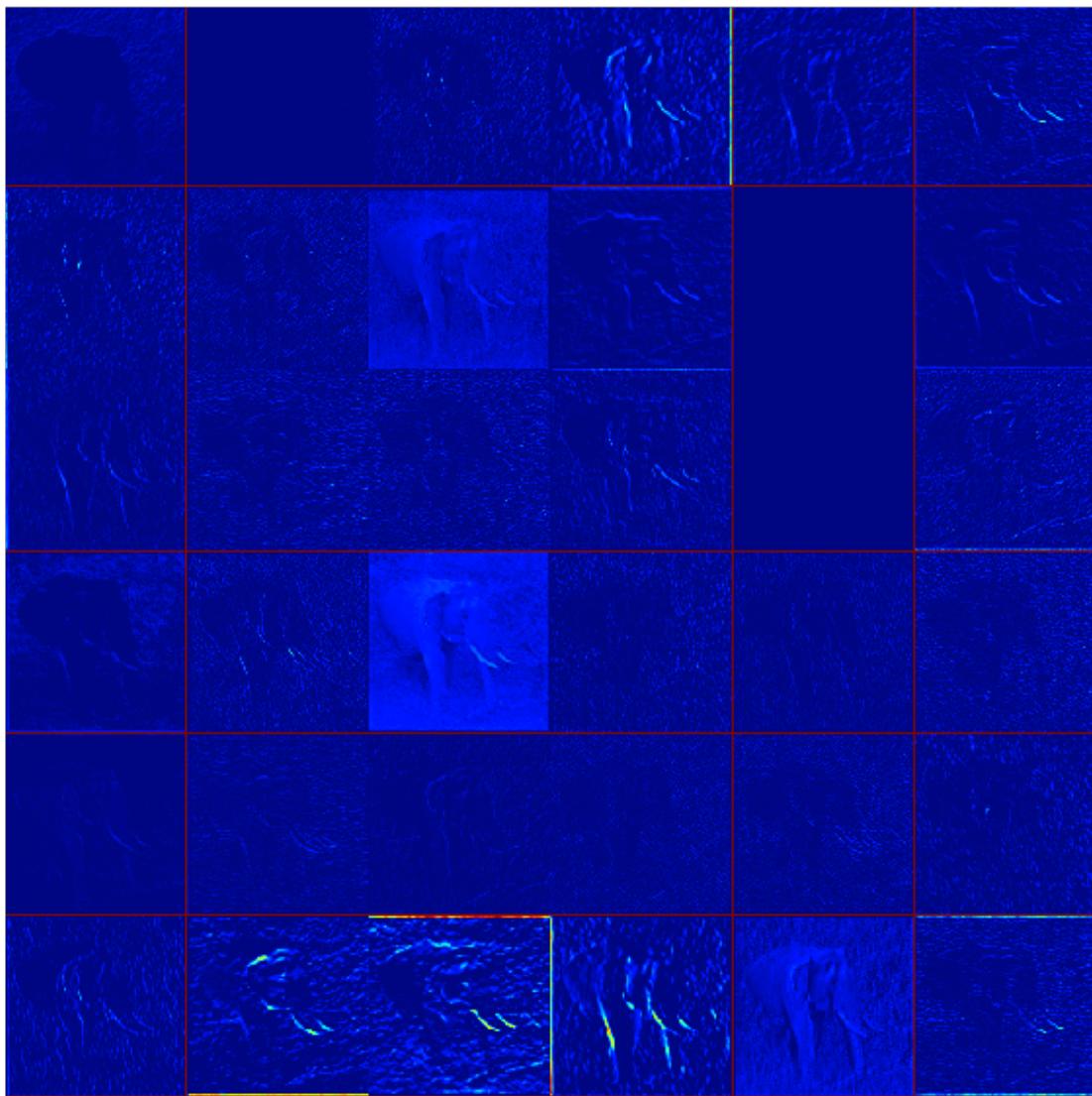
```

```
inception_4e/pool_proj      (128, 528, 1, 1) (128, )
inception_5a/1x1            (256, 832, 1, 1) (256, )
inception_5a/3x3_reduce    (160, 832, 1, 1) (160, )
inception_5a/3x3           (320, 160, 3, 3) (320, )
inception_5a/5x5_reduce    (32, 832, 1, 1) (32, )
inception_5a/5x5           (128, 32, 5, 5) (128, )
inception_5a/pool_proj     (128, 832, 1, 1) (128, )
inception_5b/1x1           (384, 832, 1, 1) (384, )
inception_5b/3x3_reduce   (192, 832, 1, 1) (192, )
inception_5b/3x3           (384, 192, 3, 3) (384, )
inception_5b/5x5_reduce   (48, 832, 1, 1) (48, )
inception_5b/5x5           (128, 48, 5, 5) (128, )
inception_5b/pool_proj     (128, 832, 1, 1) (128, )
loss3/classifier          (1000, 1024) (1000, )
```

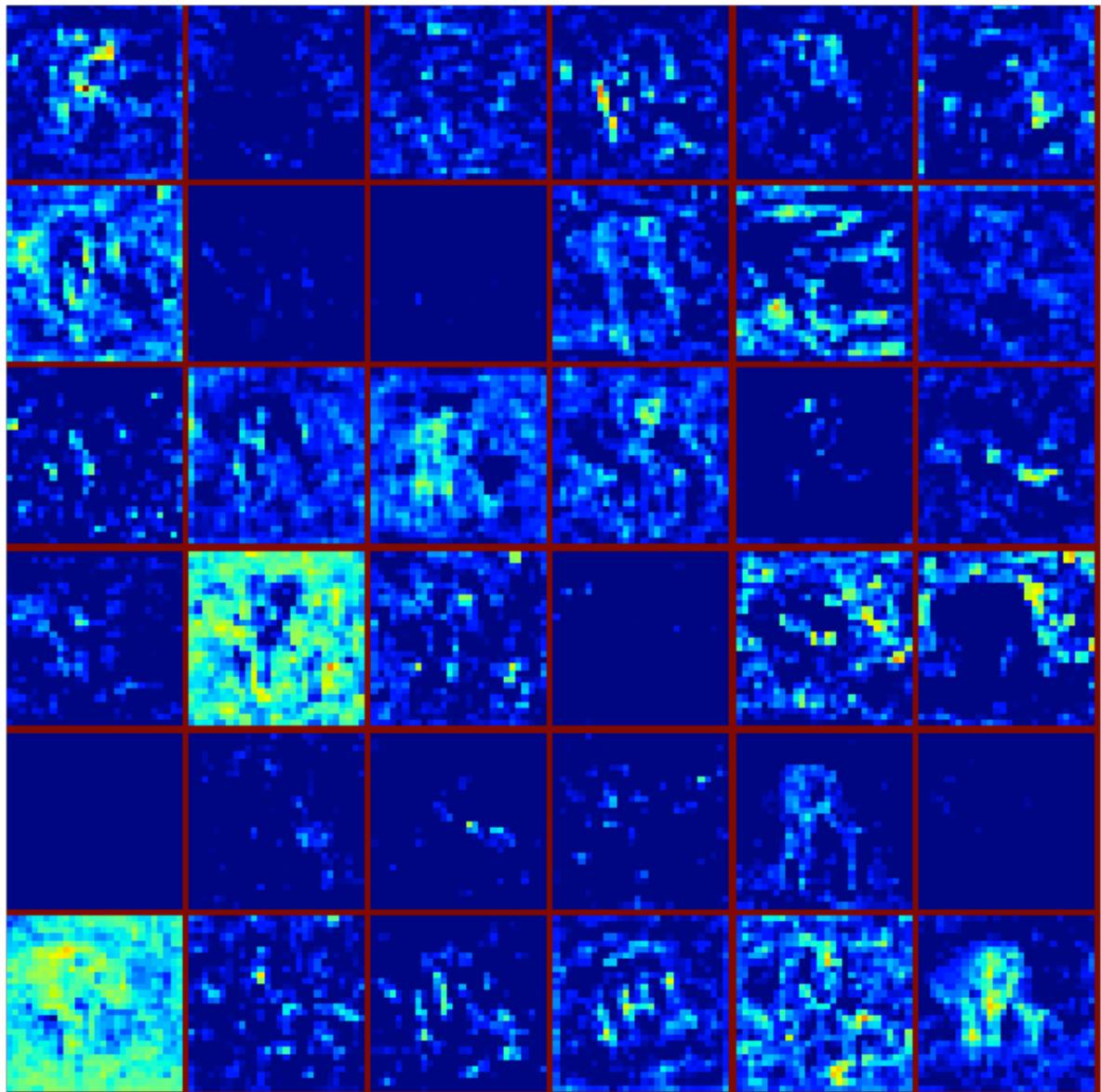
the parameters (weights) of the first conv Layer



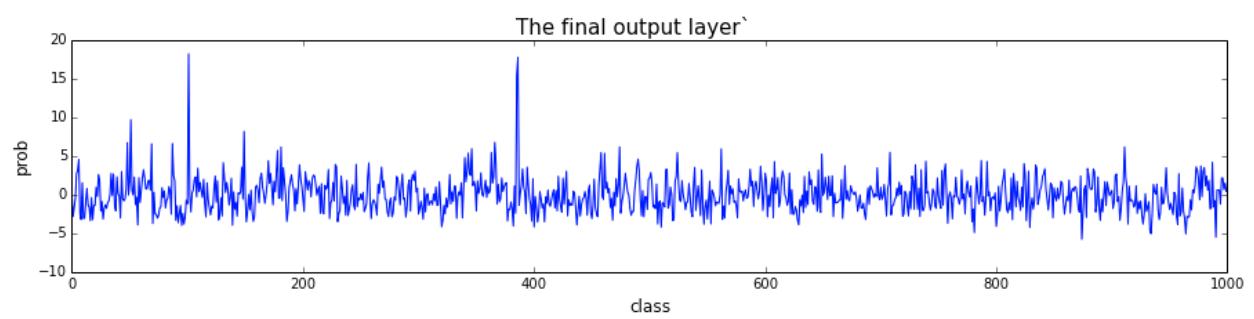
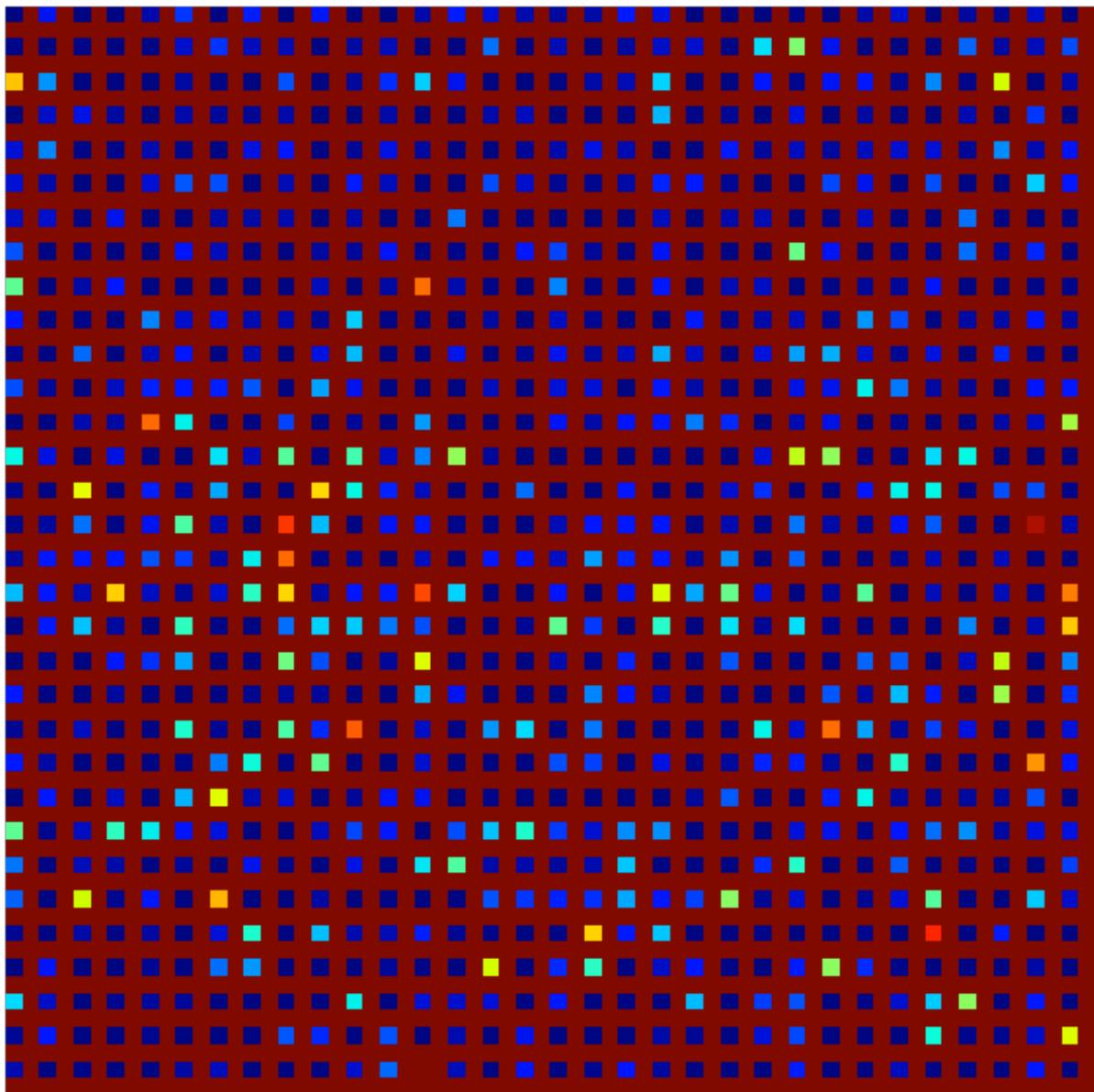
The first layer output, conv1 (rectified responses of the filters above, first 36 only)



inception\_3a/output



The last layer of pooling, pool5/7x7\_s1

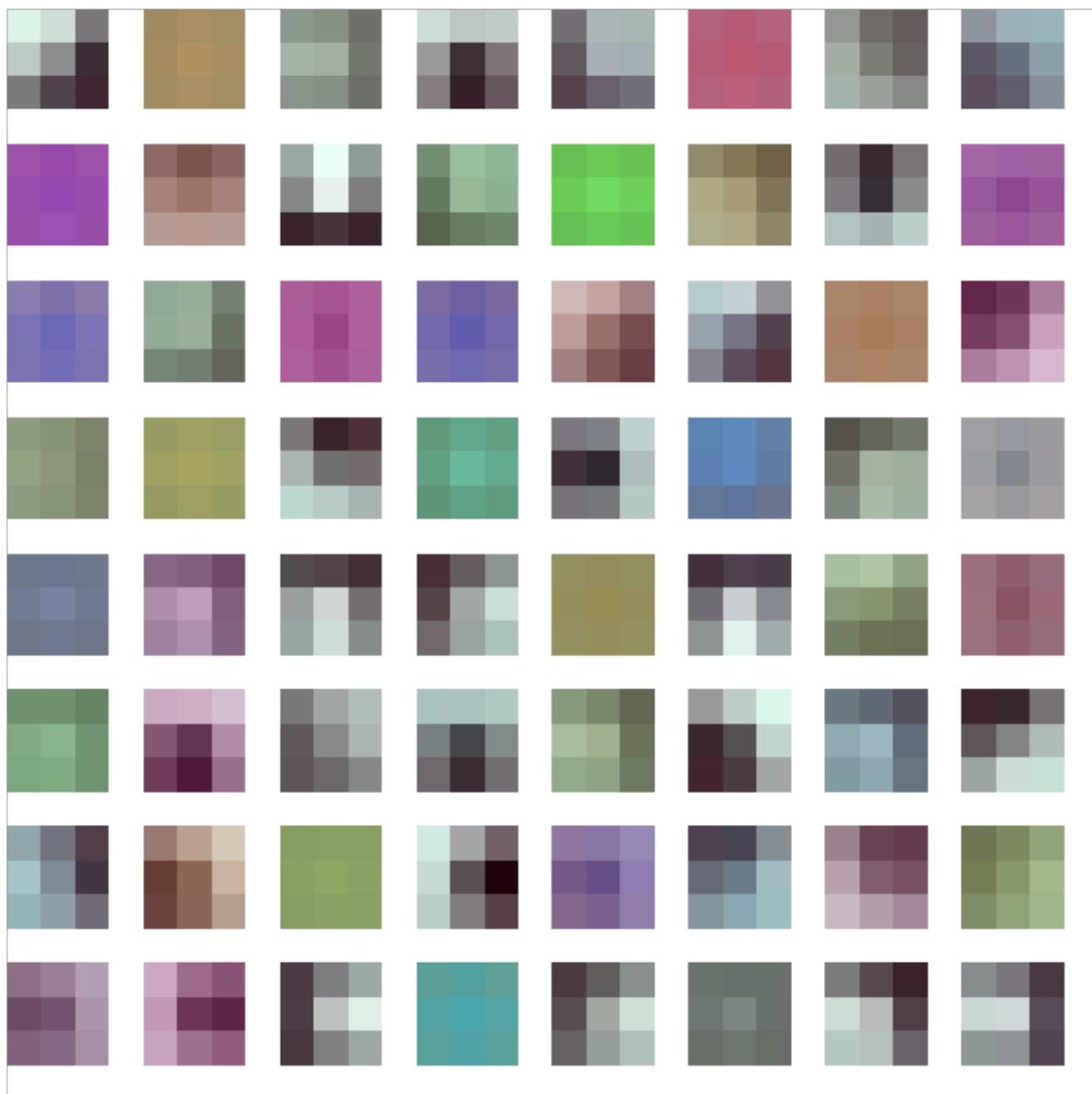


VGG-16:

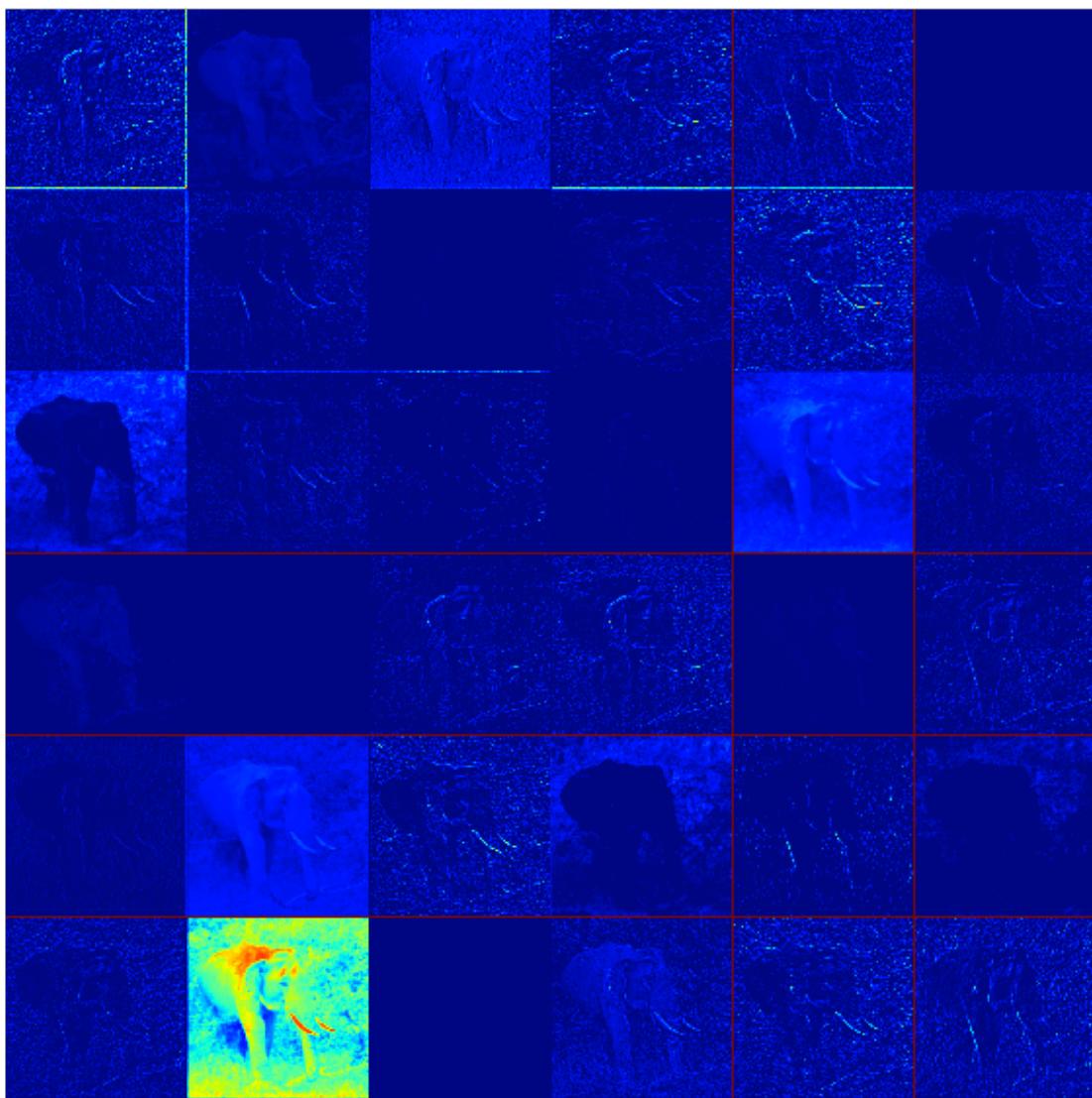
```
Network data shape visualization
data    (10, 3, 224, 224)
conv1_1(10, 64, 224, 224)
conv1_2(10, 64, 224, 224)
pool1   (10, 64, 112, 112)
conv2_1(10, 128, 112, 112)
conv2_2(10, 128, 112, 112)
pool2   (10, 128, 56, 56)
conv3_1(10, 256, 56, 56)
conv3_2(10, 256, 56, 56)
conv3_3(10, 256, 56, 56)
pool3   (10, 256, 28, 28)
conv4_1(10, 512, 28, 28)
conv4_2(10, 512, 28, 28)
conv4_3(10, 512, 28, 28)
pool4   (10, 512, 14, 14)
conv5_1(10, 512, 14, 14)
conv5_2(10, 512, 14, 14)
conv5_3(10, 512, 14, 14)
pool5   (10, 512, 7, 7)
fc6     (10, 4096)
fc7     (10, 4096)
fc8     (10, 1000)
prob    (10, 1000)
```

```
Network parameters shape visualization
conv1_1(64, 3, 3, 3) (64,)
conv1_2(64, 64, 3, 3) (64,)
conv2_1(128, 64, 3, 3) (128,)
conv2_2(128, 128, 3, 3) (128,)
conv3_1(256, 128, 3, 3) (256,)
conv3_2(256, 256, 3, 3) (256,)
conv3_3(256, 256, 3, 3) (256,)
conv4_1(512, 256, 3, 3) (512,)
conv4_2(512, 512, 3, 3) (512,)
conv4_3(512, 512, 3, 3) (512,)
conv5_1(512, 512, 3, 3) (512,)
conv5_2(512, 512, 3, 3) (512,)
conv5_3(512, 512, 3, 3) (512,)
fc6    (4096, 25088) (4096,)
fc7    (4096, 4096) (4096,)
fc8    (1000, 4096) (1000,)
```

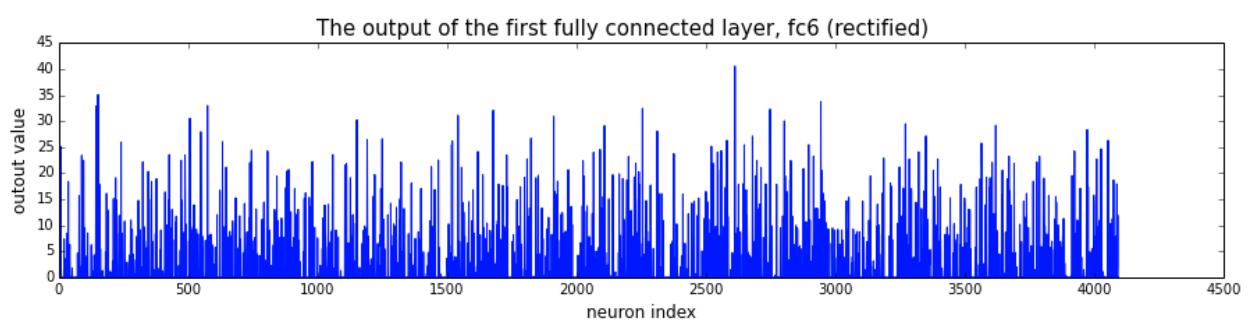
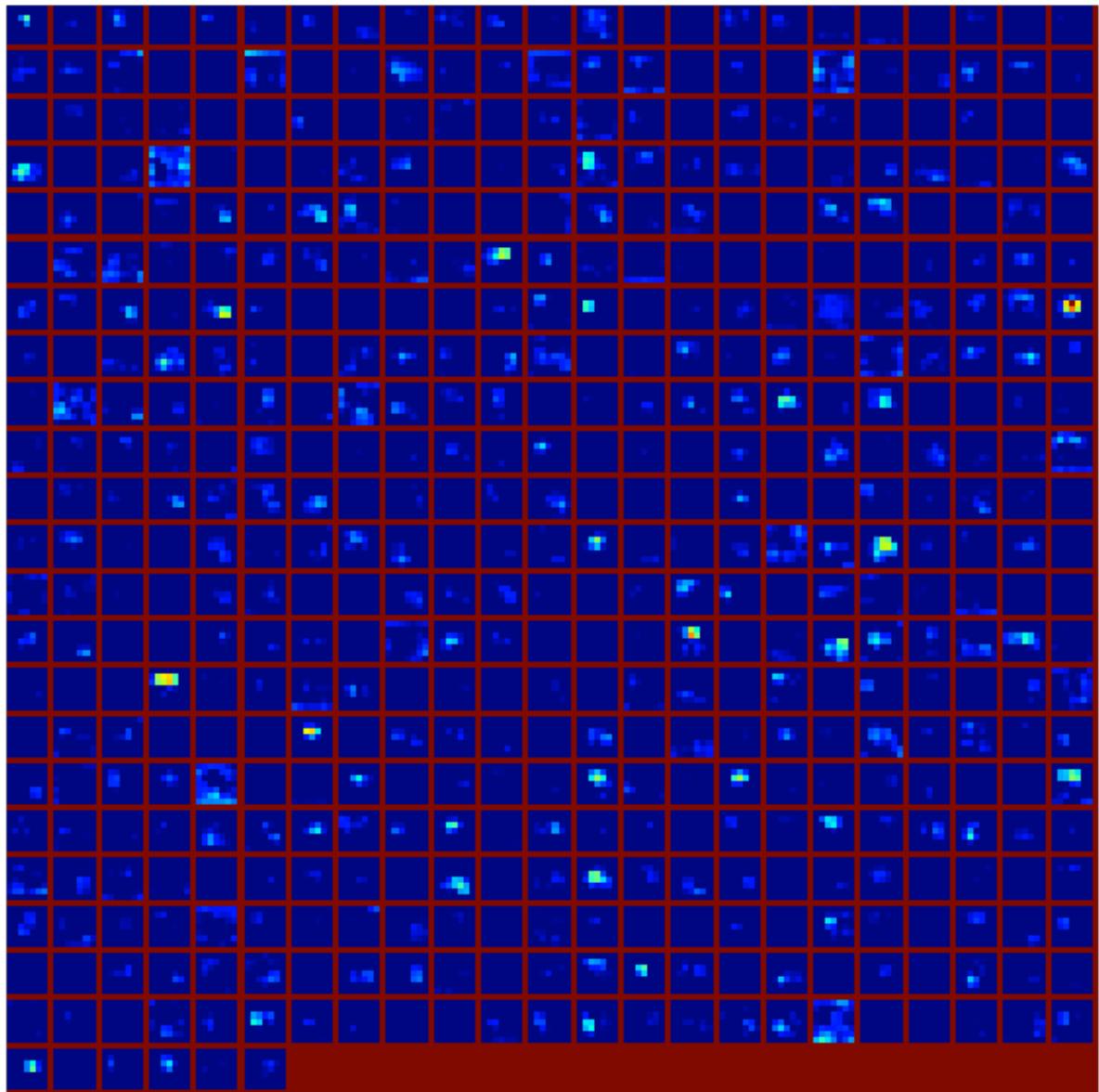
the parameters (weights) of the first conv Layer

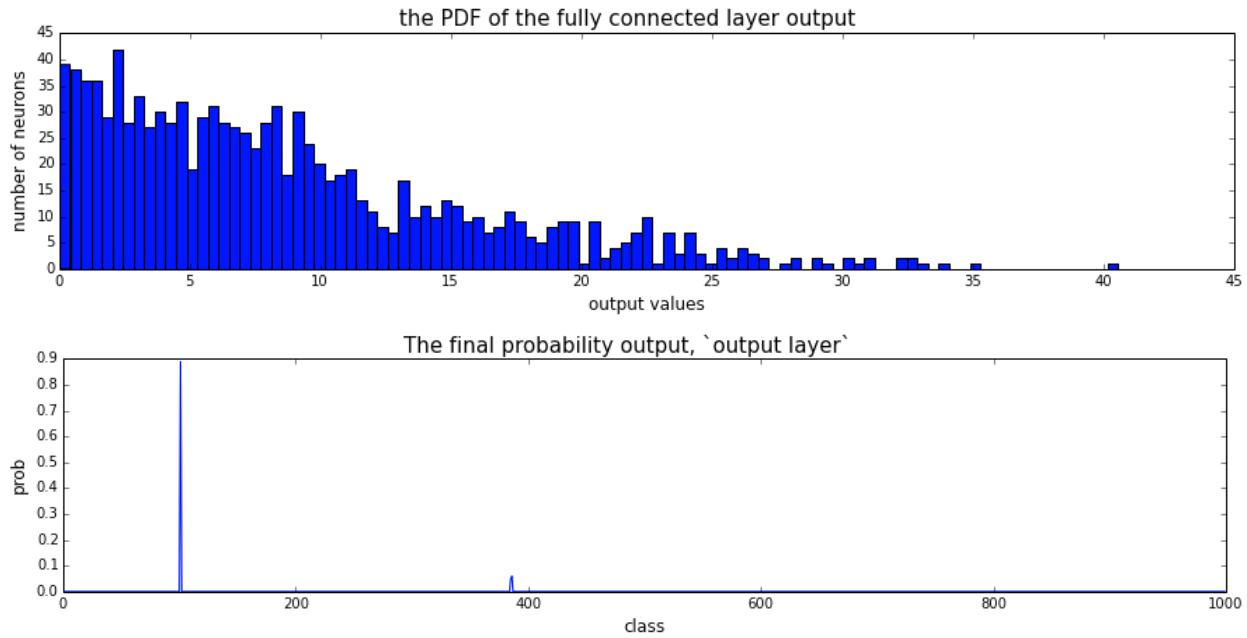


The first layer output, conv1 (rectified responses of the filters above, first 36 only)



The fifth layer after pooling, pool5





### VGG-19:

Network data shape visualization

```

data      (10, 3, 224, 224)
conv1_1(10, 64, 224, 224)
conv1_2(10, 64, 224, 224)
pool1   (10, 64, 112, 112)
conv2_1(10, 128, 112, 112)
conv2_2(10, 128, 112, 112)
pool2   (10, 128, 56, 56)
conv3_1(10, 256, 56, 56)
conv3_2(10, 256, 56, 56)
conv3_3(10, 256, 56, 56)
conv3_4(10, 256, 56, 56)
pool3   (10, 256, 28, 28)
conv4_1(10, 512, 28, 28)
conv4_2(10, 512, 28, 28)
conv4_3(10, 512, 28, 28)
conv4_4(10, 512, 28, 28)
pool4   (10, 512, 14, 14)
conv5_1(10, 512, 14, 14)
conv5_2(10, 512, 14, 14)
conv5_3(10, 512, 14, 14)
conv5_4(10, 512, 14, 14)
pool5   (10, 512, 7, 7)
fc6     (10, 4096)
fc7     (10, 4096)
fc8     (10, 1000)
prob    (10, 1000)

```

Network parameters shape visualization

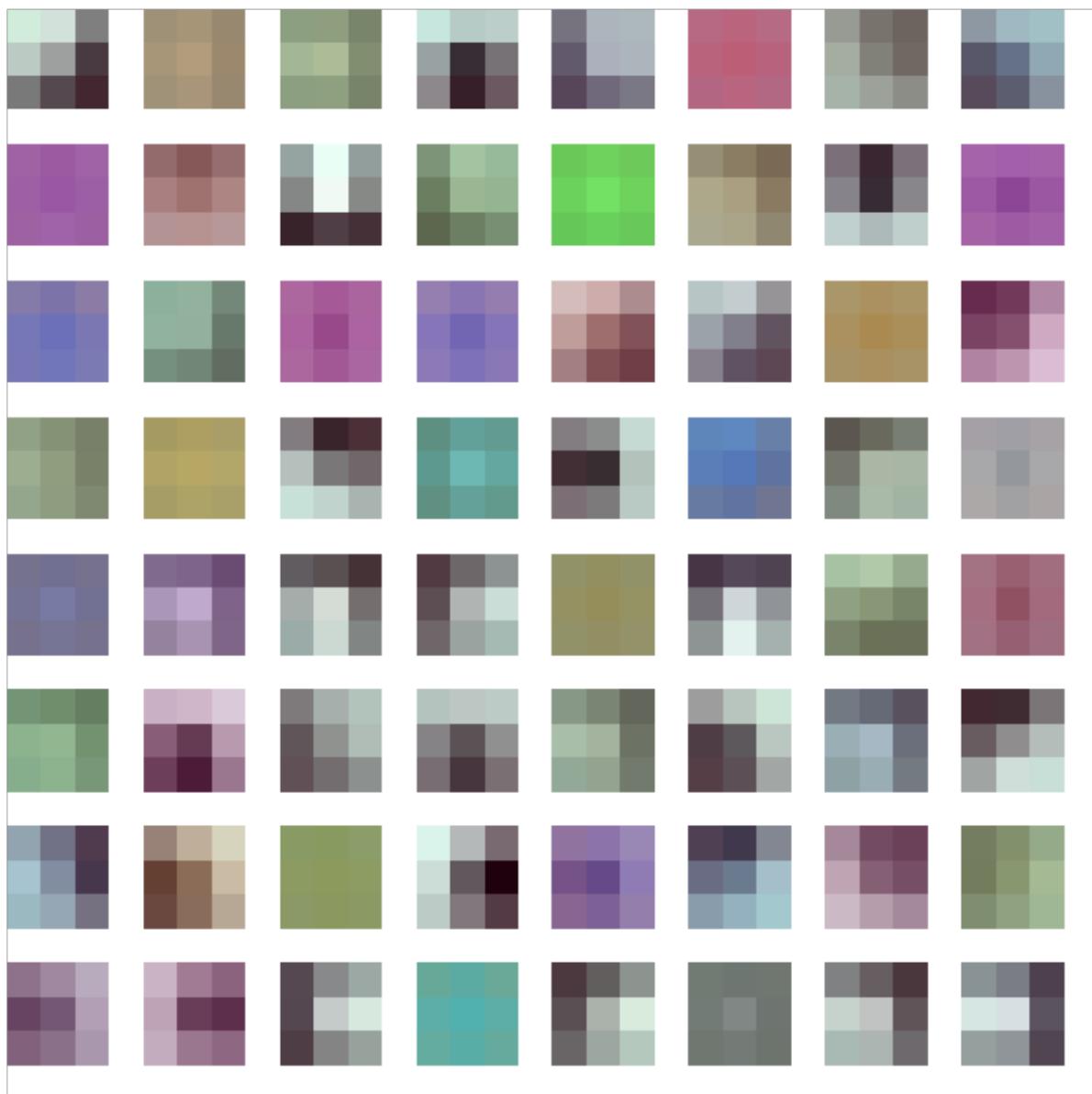
```

conv1_1(64, 3, 3, 3) (64,)
conv1_2(64, 64, 3, 3) (64,)
conv2_1(128, 64, 3, 3) (128,)
conv2_2(128, 128, 3, 3) (128,)

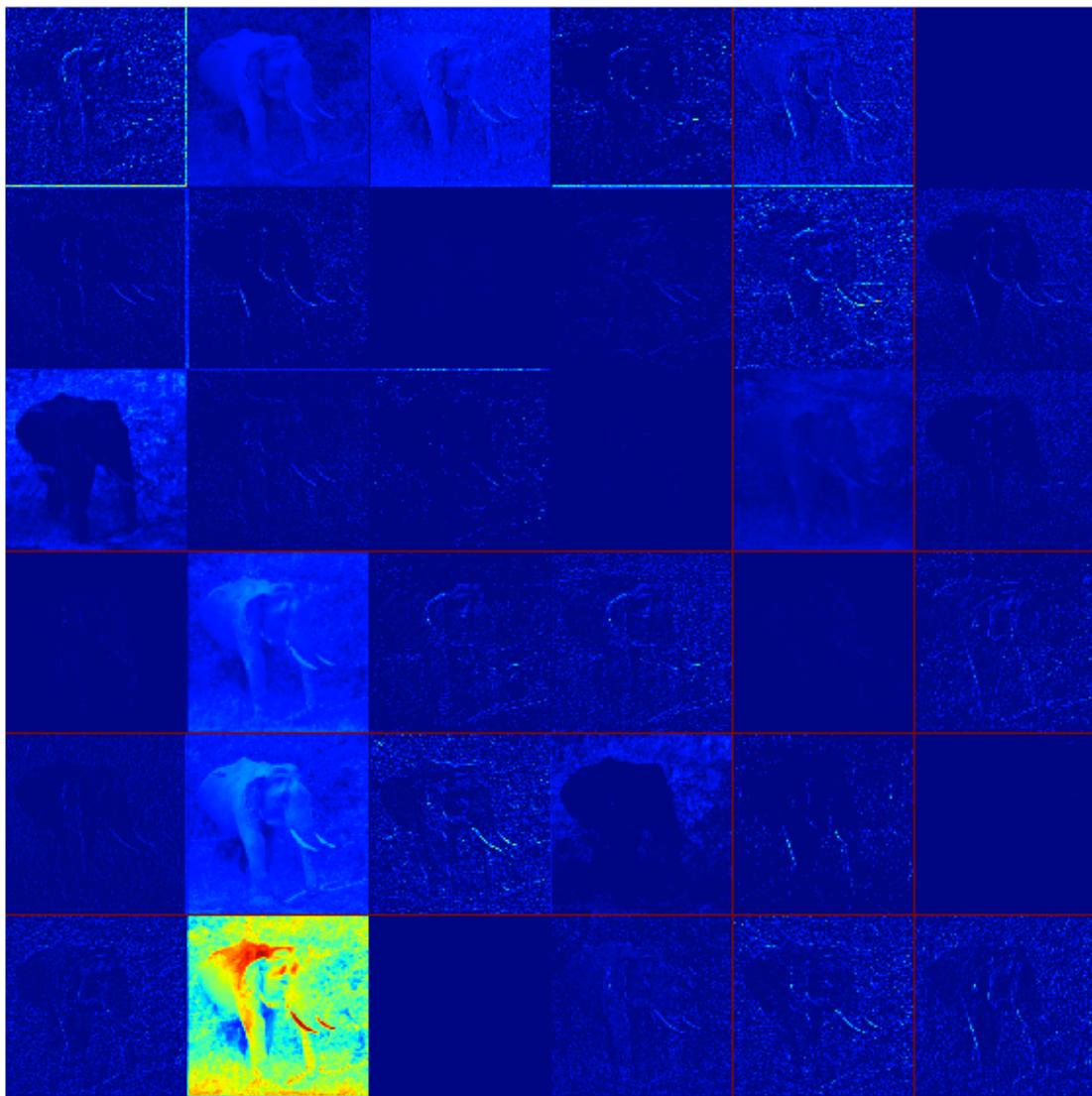
```

```
conv3_1 (256, 128, 3, 3) (256, )
conv3_2 (256, 256, 3, 3) (256, )
conv3_3 (256, 256, 3, 3) (256, )
conv3_4 (256, 256, 3, 3) (256, )
conv4_1 (512, 256, 3, 3) (512, )
conv4_2 (512, 512, 3, 3) (512, )
conv4_3 (512, 512, 3, 3) (512, )
conv4_4 (512, 512, 3, 3) (512, )
conv5_1 (512, 512, 3, 3) (512, )
conv5_2 (512, 512, 3, 3) (512, )
conv5_3 (512, 512, 3, 3) (512, )
conv5_4 (512, 512, 3, 3) (512, )
fc6      (4096, 25088) (4096, )
fc7      (4096, 4096) (4096, )
fc8      (1000, 4096) (1000, )
```

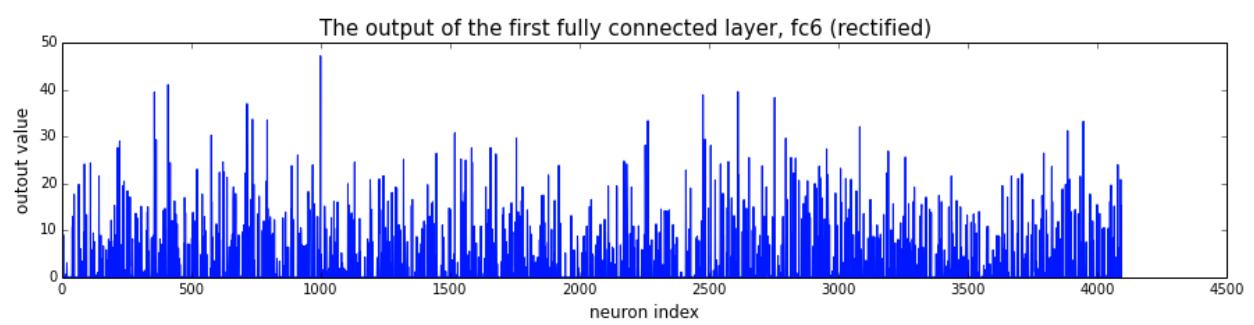
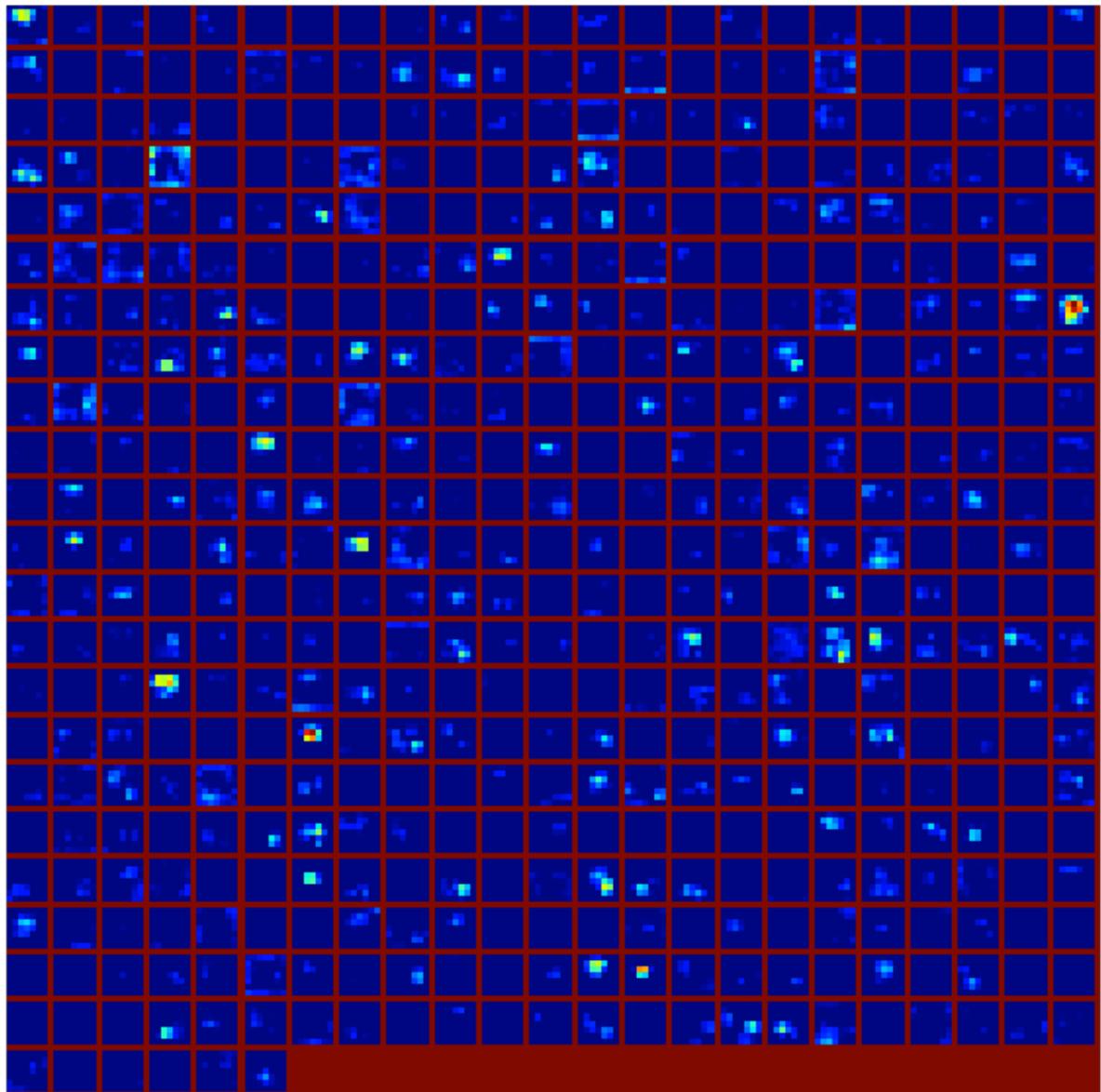
the parameters (weights) of the first conv Layer

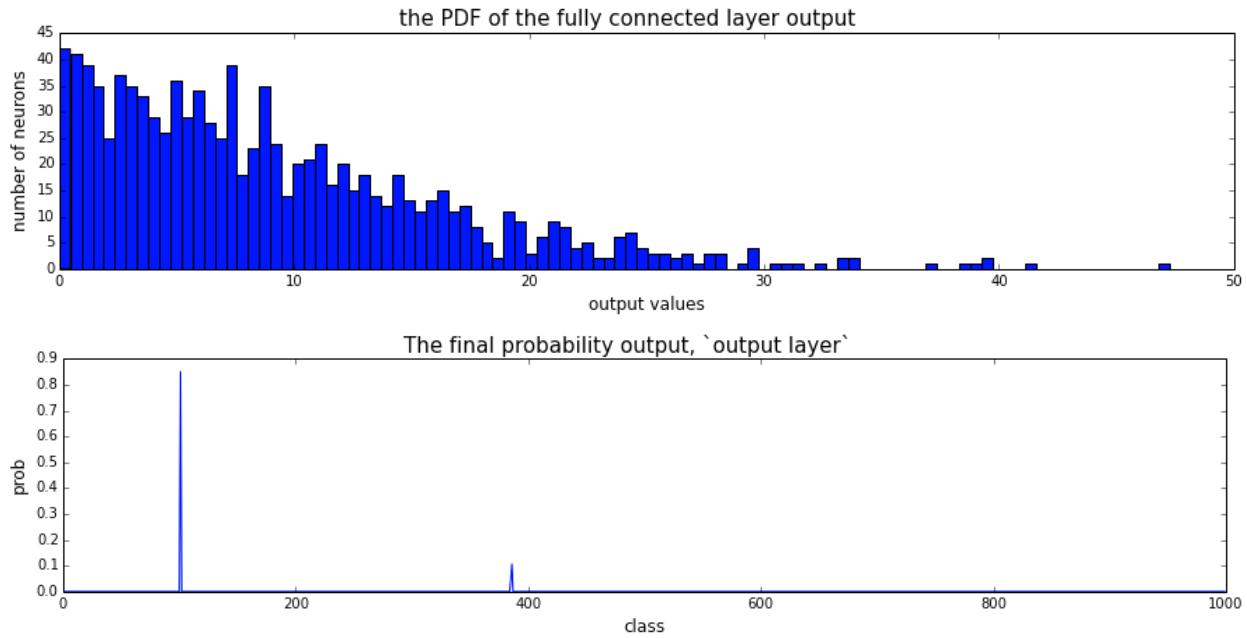


The first layer output, conv1 (rectified responses of the filters above, first 36 only)



The fifth layer after pooling, pool5





ResNet-50:

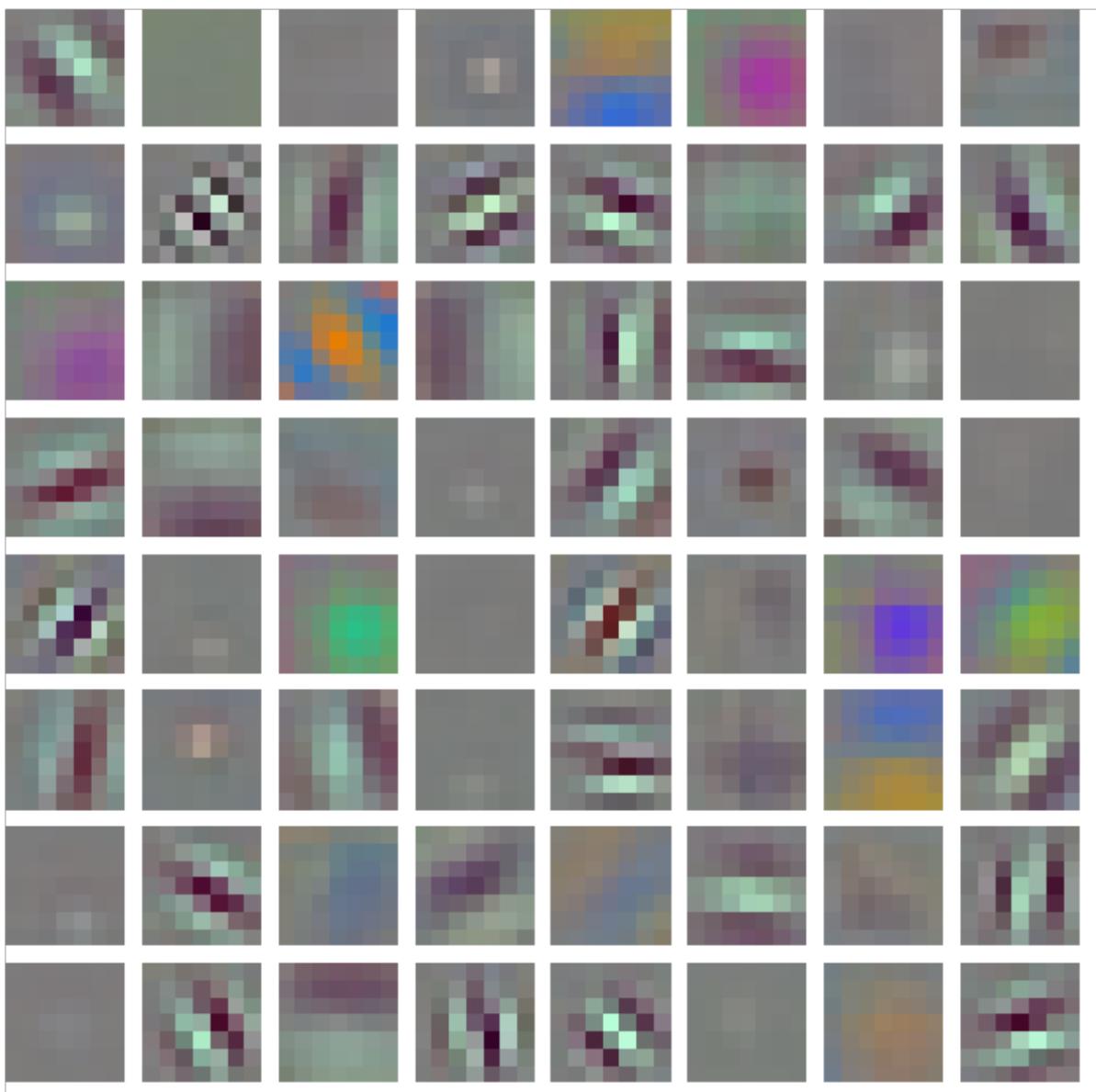
```

Network data shape visualization
data      (1, 3, 224, 224)
conv1    (1, 64, 112, 112)
pool1    (1, 64, 56, 56)
pool1_pool1_0_split_0 (1, 64, 56, 56)
pool1_pool1_0_split_1 (1, 64, 56, 56)
res2a_branch1 (1, 256, 56, 56)
res2a_branch2a (1, 64, 56, 56)
res2a_branch2b (1, 64, 56, 56)
res2a_branch2c (1, 256, 56, 56)
res2a    (1, 256, 56, 56)
res2a_res2a_relu_0_split_0   (1, 256, 56, 56)
res2a_res2a_relu_0_split_1   (1, 256, 56, 56)
res2b_branch2a (1, 64, 56, 56)
res2b_branch2b (1, 64, 56, 56)
res2b_branch2c (1, 256, 56, 56)
res2b    (1, 256, 56, 56)
res2b_res2b_relu_0_split_0   (1, 256, 56, 56)
res2b_res2b_relu_0_split_1   (1, 256, 56, 56)
res2c_branch2a (1, 64, 56, 56)
res2c_branch2b (1, 64, 56, 56)
res2c_branch2c (1, 256, 56, 56)
res2c    (1, 256, 56, 56)
res2c_res2c_relu_0_split_0   (1, 256, 56, 56)
res2c_res2c_relu_0_split_1   (1, 256, 56, 56)
res3a_branch1 (1, 512, 28, 28)
res3a_branch2a (1, 128, 28, 28)
res3a_branch2b (1, 128, 28, 28)
res3a_branch2c (1, 512, 28, 28)
res3a    (1, 512, 28, 28)
res3a_res3a_relu_0_split_0   (1, 512, 28, 28)
res3a_res3a_relu_0_split_1   (1, 512, 28, 28)
res3b_branch2a (1, 128, 28, 28)
```

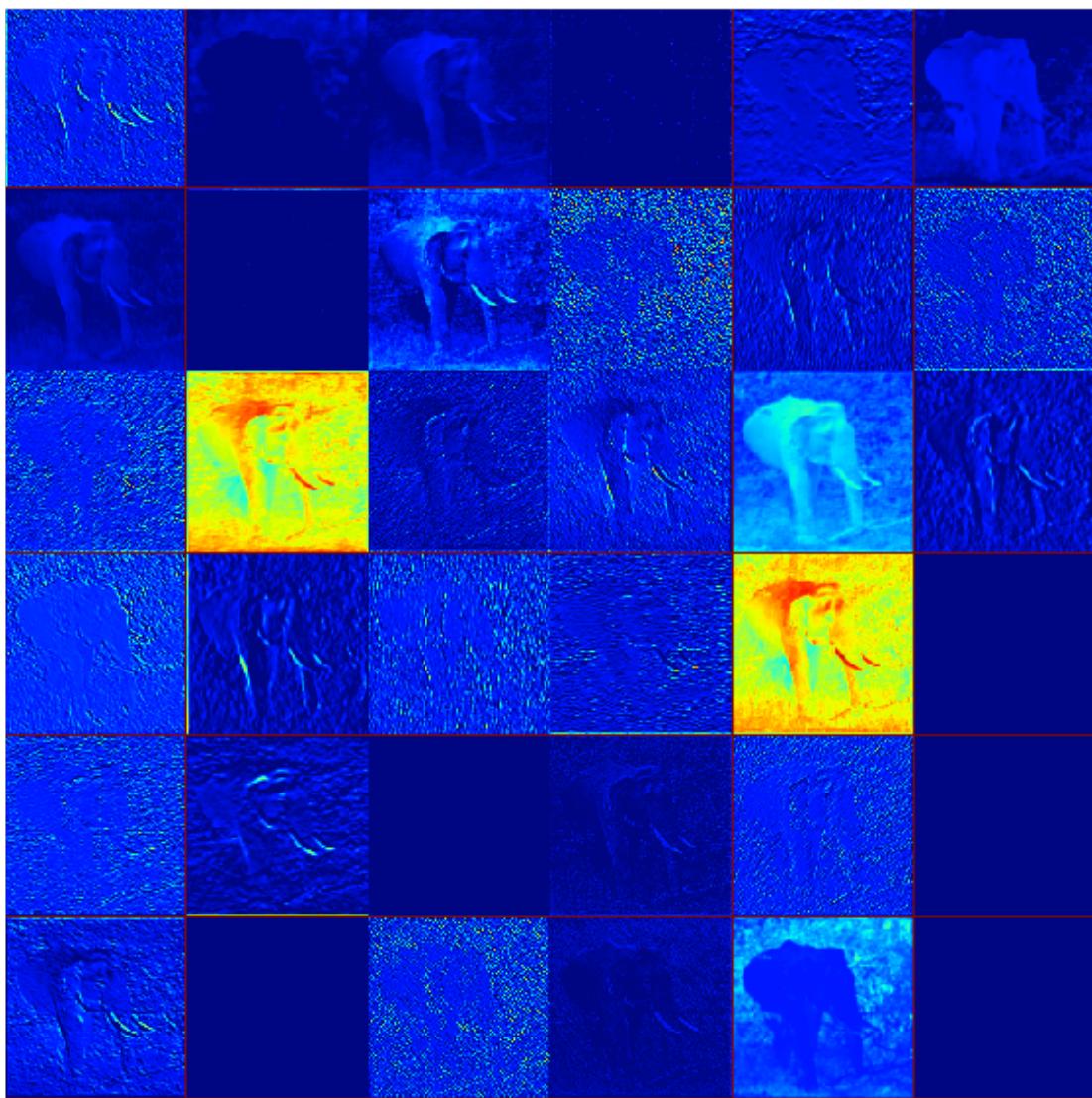
```
res3b_branch2b (1, 128, 28, 28)
res3b_branch2c (1, 512, 28, 28)
res3b (1, 512, 28, 28)
res3b_res3b_relu_0_split_0 (1, 512, 28, 28)
res3b_res3b_relu_0_split_1 (1, 512, 28, 28)
res3c_branch2a (1, 128, 28, 28)
res3c_branch2b (1, 128, 28, 28)
res3c_branch2c (1, 512, 28, 28)
res3c (1, 512, 28, 28)
res3c_res3c_relu_0_split_0 (1, 512, 28, 28)
res3c_res3c_relu_0_split_1 (1, 512, 28, 28)
res3d_branch2a (1, 128, 28, 28)
res3d_branch2b (1, 128, 28, 28)
res3d_branch2c (1, 512, 28, 28)
res3d (1, 512, 28, 28)
res3d_res3d_relu_0_split_0 (1, 512, 28, 28)
res3d_res3d_relu_0_split_1 (1, 512, 28, 28)
res4a_branch1 (1, 1024, 14, 14)
res4a_branch2a (1, 256, 14, 14)
res4a_branch2b (1, 256, 14, 14)
res4a_branch2c (1, 1024, 14, 14)
res4a (1, 1024, 14, 14)
res4a_res4a_relu_0_split_0 (1, 1024, 14, 14)
res4a_res4a_relu_0_split_1 (1, 1024, 14, 14)
res4b_branch2a (1, 256, 14, 14)
res4b_branch2b (1, 256, 14, 14)
res4b_branch2c (1, 1024, 14, 14)
res4b (1, 1024, 14, 14)
res4b_res4b_relu_0_split_0 (1, 1024, 14, 14)
res4b_res4b_relu_0_split_1 (1, 1024, 14, 14)
res4c_branch2a (1, 256, 14, 14)
res4c_branch2b (1, 256, 14, 14)
res4c_branch2c (1, 1024, 14, 14)
res4c (1, 1024, 14, 14)
res4c_res4c_relu_0_split_0 (1, 1024, 14, 14)
res4c_res4c_relu_0_split_1 (1, 1024, 14, 14)
res4d_branch2a (1, 256, 14, 14)
res4d_branch2b (1, 256, 14, 14)
res4d_branch2c (1, 1024, 14, 14)
res4d (1, 1024, 14, 14)
res4d_res4d_relu_0_split_0 (1, 1024, 14, 14)
res4d_res4d_relu_0_split_1 (1, 1024, 14, 14)
res4e_branch2a (1, 256, 14, 14)
res4e_branch2b (1, 256, 14, 14)
res4e_branch2c (1, 1024, 14, 14)
res4e (1, 1024, 14, 14)
res4e_res4e_relu_0_split_0 (1, 1024, 14, 14)
res4e_res4e_relu_0_split_1 (1, 1024, 14, 14)
res4f_branch2a (1, 256, 14, 14)
res4f_branch2b (1, 256, 14, 14)
res4f_branch2c (1, 1024, 14, 14)
res4f (1, 1024, 14, 14)
res4f_res4f_relu_0_split_0 (1, 1024, 14, 14)
res4f_res4f_relu_0_split_1 (1, 1024, 14, 14)
res5a_branch1 (1, 2048, 7, 7)
```

```
res5a_branch2a (1, 512, 7, 7)
res5a_branch2b (1, 512, 7, 7)
res5a_branch2c (1, 2048, 7, 7)
res5a (1, 2048, 7, 7)
res5a_res5a_relu_0_split_0 (1, 2048, 7, 7)
res5a_res5a_relu_0_split_1 (1, 2048, 7, 7)
res5b_branch2a (1, 512, 7, 7)
res5b_branch2b (1, 512, 7, 7)
res5b_branch2c (1, 2048, 7, 7)
res5b (1, 2048, 7, 7)
res5b_res5b_relu_0_split_0 (1, 2048, 7, 7)
res5b_res5b_relu_0_split_1 (1, 2048, 7, 7)
res5c_branch2a (1, 512, 7, 7)
res5c_branch2b (1, 512, 7, 7)
res5c_branch2c (1, 2048, 7, 7)
res5c (1, 2048, 7, 7)
pool5 (1, 2048, 1, 1)
fc1000 (1, 1000)
prob (1, 1000)
```

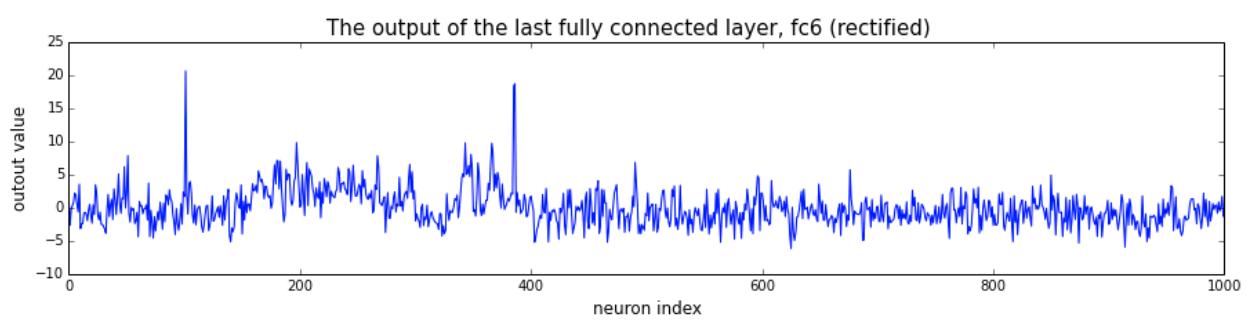
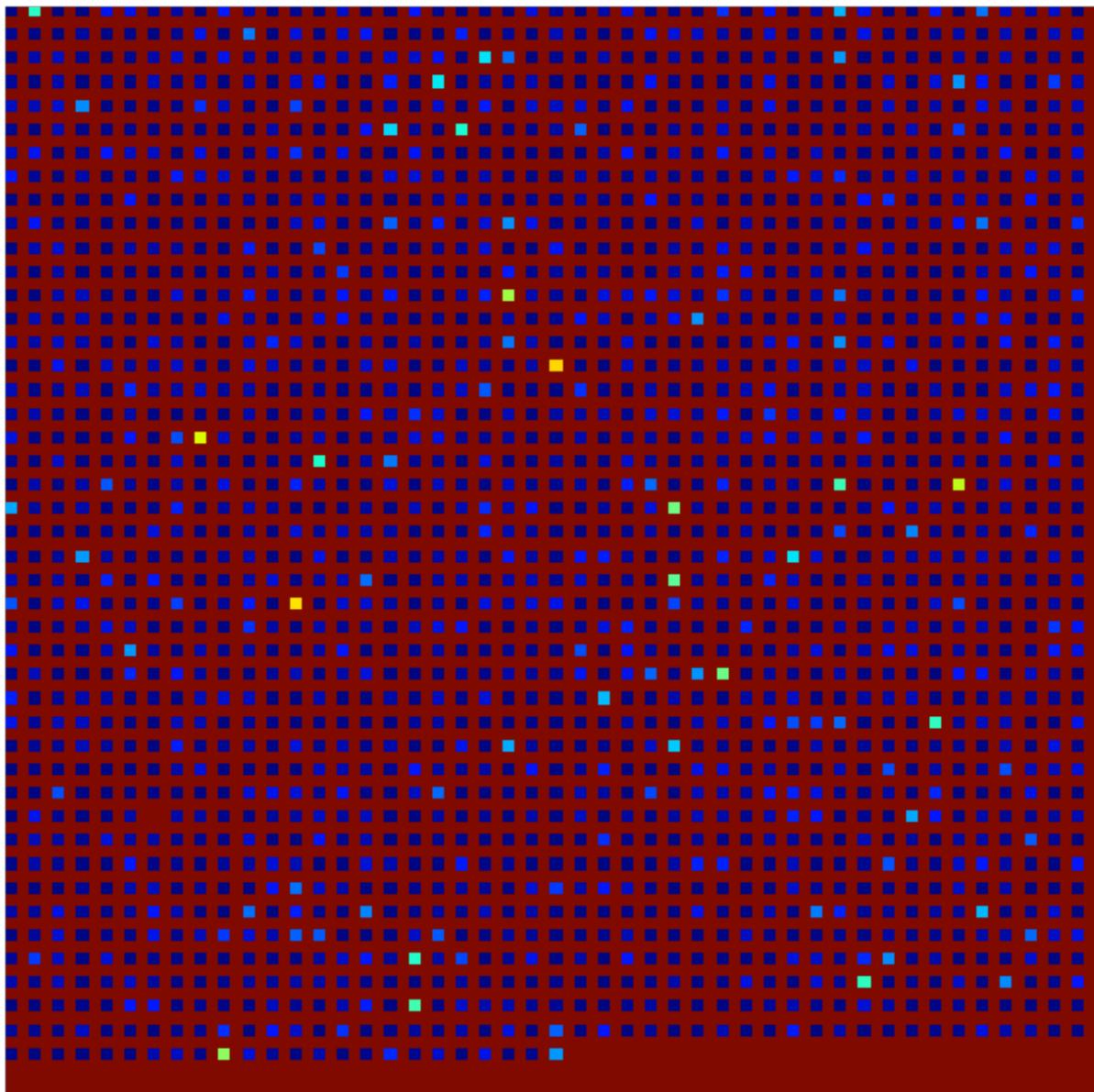
the parameters (weights) of the first conv Layer

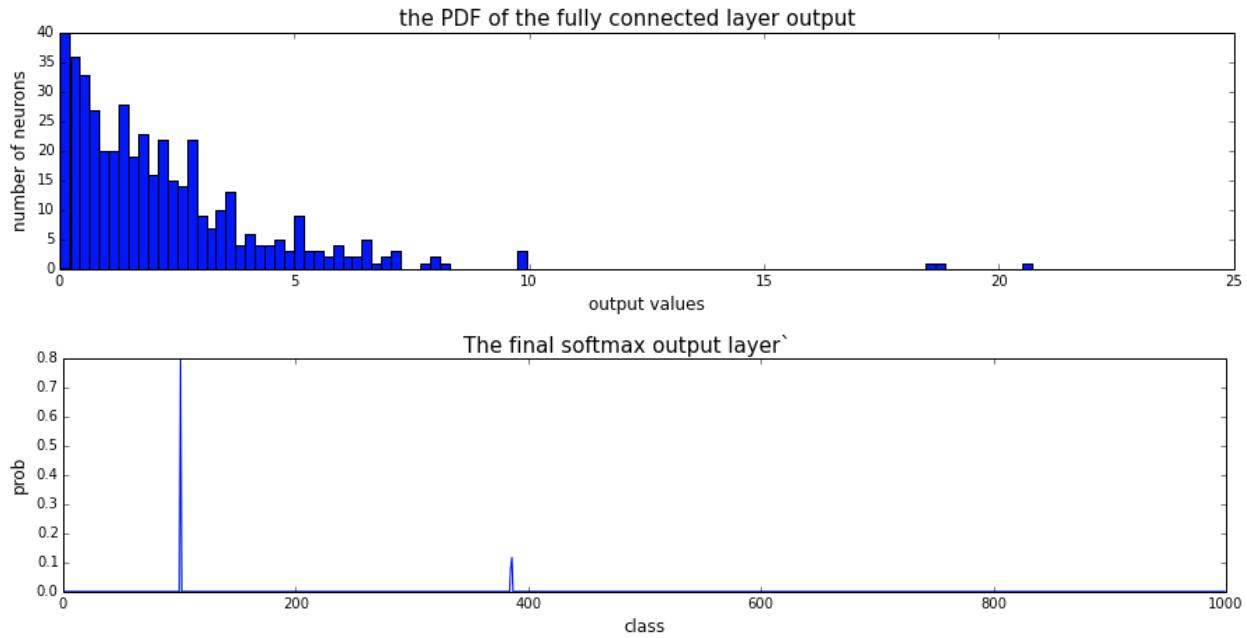


The first layer output, conv1 (rectified responses of the filters above, first 36 only)



The last layer of pooling, pool5/7x7\_s1





### ResNet-101:

Network data shape visualization

```

data      (1, 3, 224, 224)
conv1    (1, 64, 112, 112)
pool1    (1, 64, 56, 56)
pool1_pool1_0_split_0 (1, 64, 56, 56)
pool1_pool1_0_split_1 (1, 64, 56, 56)
res2a_branch1 (1, 256, 56, 56)
res2a_branch2a (1, 64, 56, 56)
res2a_branch2b (1, 64, 56, 56)
res2a_branch2c (1, 256, 56, 56)
res2a    (1, 256, 56, 56)
res2a_res2a_relu_0_split_0   (1, 256, 56, 56)
res2a_res2a_relu_0_split_1   (1, 256, 56, 56)
res2b_branch2a (1, 64, 56, 56)
res2b_branch2b (1, 64, 56, 56)
res2b_branch2c (1, 256, 56, 56)
res2b    (1, 256, 56, 56)
res2b_res2b_relu_0_split_0   (1, 256, 56, 56)
res2b_res2b_relu_0_split_1   (1, 256, 56, 56)
res2c_branch2a (1, 64, 56, 56)
res2c_branch2b (1, 64, 56, 56)
res2c_branch2c (1, 256, 56, 56)
res2c    (1, 256, 56, 56)
res2c_res2c_relu_0_split_0   (1, 256, 56, 56)
res2c_res2c_relu_0_split_1   (1, 256, 56, 56)
res3a_branch1 (1, 512, 28, 28)
res3a_branch2a (1, 128, 28, 28)
res3a_branch2b (1, 128, 28, 28)
res3a_branch2c (1, 512, 28, 28)
res3a    (1, 512, 28, 28)
res3a_res3a_relu_0_split_0   (1, 512, 28, 28)
res3a_res3a_relu_0_split_1   (1, 512, 28, 28)
res3b1_branch2a   (1, 128, 28, 28)
```

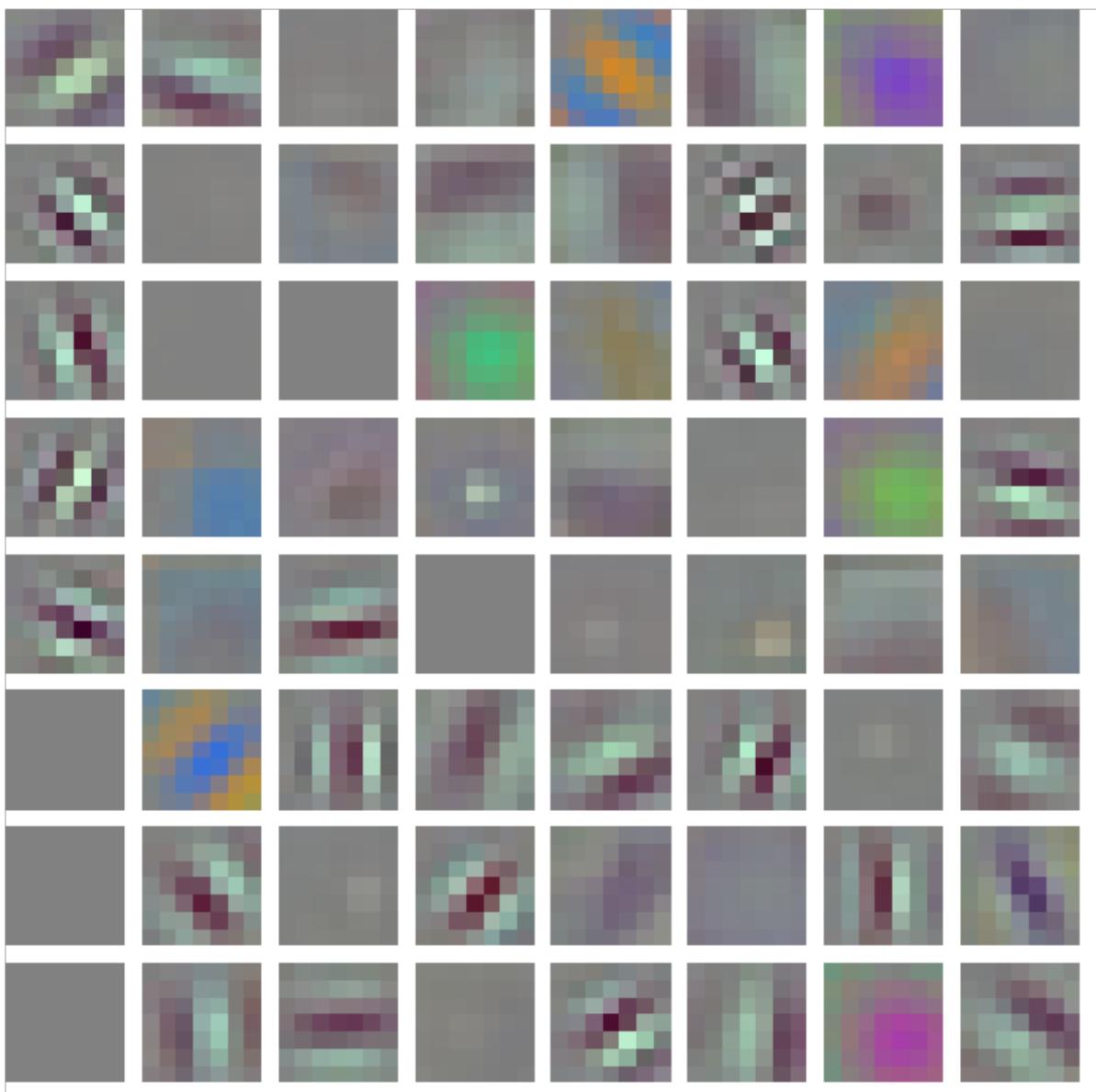
```
res3b1_branch2b      (1, 128, 28, 28)
res3b1_branch2c      (1, 512, 28, 28)
res3b1 (1, 512, 28, 28)
res3b1_res3b1_relu_0_split_0 (1, 512, 28, 28)
res3b1_res3b1_relu_0_split_1 (1, 512, 28, 28)
res3b2_branch2a      (1, 128, 28, 28)
res3b2_branch2b      (1, 128, 28, 28)
res3b2_branch2c      (1, 512, 28, 28)
res3b2 (1, 512, 28, 28)
res3b2_res3b2_relu_0_split_0 (1, 512, 28, 28)
res3b2_res3b2_relu_0_split_1 (1, 512, 28, 28)
res3b3_branch2a      (1, 128, 28, 28)
res3b3_branch2b      (1, 128, 28, 28)
res3b3_branch2c      (1, 512, 28, 28)
res3b3 (1, 512, 28, 28)
res3b3_res3b3_relu_0_split_0 (1, 512, 28, 28)
res3b3_res3b3_relu_0_split_1 (1, 512, 28, 28)
res4a_branch1 (1, 1024, 14, 14)
res4a_branch2a (1, 256, 14, 14)
res4a_branch2b (1, 256, 14, 14)
res4a_branch2c (1, 1024, 14, 14)
res4a (1, 1024, 14, 14)
res4a_res4a_relu_0_split_0 (1, 1024, 14, 14)
res4a_res4a_relu_0_split_1 (1, 1024, 14, 14)
res4b1_branch2a      (1, 256, 14, 14)
res4b1_branch2b      (1, 256, 14, 14)
res4b1_branch2c      (1, 1024, 14, 14)
res4b1 (1, 1024, 14, 14)
res4b1_res4b1_relu_0_split_0 (1, 1024, 14, 14)
res4b1_res4b1_relu_0_split_1 (1, 1024, 14, 14)
res4b2_branch2a      (1, 256, 14, 14)
res4b2_branch2b      (1, 256, 14, 14)
res4b2_branch2c      (1, 1024, 14, 14)
res4b2 (1, 1024, 14, 14)
res4b2_res4b2_relu_0_split_0 (1, 1024, 14, 14)
res4b2_res4b2_relu_0_split_1 (1, 1024, 14, 14)
res4b3_branch2a      (1, 256, 14, 14)
res4b3_branch2b      (1, 256, 14, 14)
res4b3_branch2c      (1, 1024, 14, 14)
res4b3 (1, 1024, 14, 14)
res4b3_res4b3_relu_0_split_0 (1, 1024, 14, 14)
res4b3_res4b3_relu_0_split_1 (1, 1024, 14, 14)
res4b4_branch2a      (1, 256, 14, 14)
res4b4_branch2b      (1, 256, 14, 14)
res4b4_branch2c      (1, 1024, 14, 14)
res4b4 (1, 1024, 14, 14)
res4b4_res4b4_relu_0_split_0 (1, 1024, 14, 14)
res4b4_res4b4_relu_0_split_1 (1, 1024, 14, 14)
res4b5_branch2a      (1, 256, 14, 14)
res4b5_branch2b      (1, 256, 14, 14)
res4b5_branch2c      (1, 1024, 14, 14)
res4b5 (1, 1024, 14, 14)
res4b5_res4b5_relu_0_split_0 (1, 1024, 14, 14)
res4b5_res4b5_relu_0_split_1 (1, 1024, 14, 14)
res4b6_branch2a      (1, 256, 14, 14)
```

```
res4b6_branch2b      (1, 256, 14, 14)
res4b6_branch2c      (1, 1024, 14, 14)
res4b6 (1, 1024, 14, 14)
res4b6_res4b6_relu_0_split_0 (1, 1024, 14, 14)
res4b6_res4b6_relu_0_split_1 (1, 1024, 14, 14)
res4b7_branch2a      (1, 256, 14, 14)
res4b7_branch2b      (1, 256, 14, 14)
res4b7_branch2c      (1, 1024, 14, 14)
res4b7 (1, 1024, 14, 14)
res4b7_res4b7_relu_0_split_0 (1, 1024, 14, 14)
res4b7_res4b7_relu_0_split_1 (1, 1024, 14, 14)
res4b8_branch2a      (1, 256, 14, 14)
res4b8_branch2b      (1, 256, 14, 14)
res4b8_branch2c      (1, 1024, 14, 14)
res4b8 (1, 1024, 14, 14)
res4b8_res4b8_relu_0_split_0 (1, 1024, 14, 14)
res4b8_res4b8_relu_0_split_1 (1, 1024, 14, 14)
res4b9_branch2a      (1, 256, 14, 14)
res4b9_branch2b      (1, 256, 14, 14)
res4b9_branch2c      (1, 1024, 14, 14)
res4b9 (1, 1024, 14, 14)
res4b9_res4b9_relu_0_split_0 (1, 1024, 14, 14)
res4b9_res4b9_relu_0_split_1 (1, 1024, 14, 14)
res4b10_branch2a     (1, 256, 14, 14)
res4b10_branch2b     (1, 256, 14, 14)
res4b10_branch2c     (1, 1024, 14, 14)
res4b10 (1, 1024, 14, 14)
res4b10_res4b10_relu_0_split_0      (1, 1024, 14, 14)
res4b10_res4b10_relu_0_split_1      (1, 1024, 14, 14)
res4b11_branch2a     (1, 256, 14, 14)
res4b11_branch2b     (1, 256, 14, 14)
res4b11_branch2c     (1, 1024, 14, 14)
res4b11 (1, 1024, 14, 14)
res4b11_res4b11_relu_0_split_0      (1, 1024, 14, 14)
res4b11_res4b11_relu_0_split_1      (1, 1024, 14, 14)
res4b12_branch2a     (1, 256, 14, 14)
res4b12_branch2b     (1, 256, 14, 14)
res4b12_branch2c     (1, 1024, 14, 14)
res4b12 (1, 1024, 14, 14)
res4b12_res4b12_relu_0_split_0      (1, 1024, 14, 14)
res4b12_res4b12_relu_0_split_1      (1, 1024, 14, 14)
res4b13_branch2a     (1, 256, 14, 14)
res4b13_branch2b     (1, 256, 14, 14)
res4b13_branch2c     (1, 1024, 14, 14)
res4b13 (1, 1024, 14, 14)
res4b13_res4b13_relu_0_split_0      (1, 1024, 14, 14)
res4b13_res4b13_relu_0_split_1      (1, 1024, 14, 14)
res4b14_branch2a     (1, 256, 14, 14)
res4b14_branch2b     (1, 256, 14, 14)
res4b14_branch2c     (1, 1024, 14, 14)
res4b14 (1, 1024, 14, 14)
res4b14_res4b14_relu_0_split_0      (1, 1024, 14, 14)
res4b14_res4b14_relu_0_split_1      (1, 1024, 14, 14)
res4b15_branch2a     (1, 256, 14, 14)
res4b15_branch2b     (1, 256, 14, 14)
```

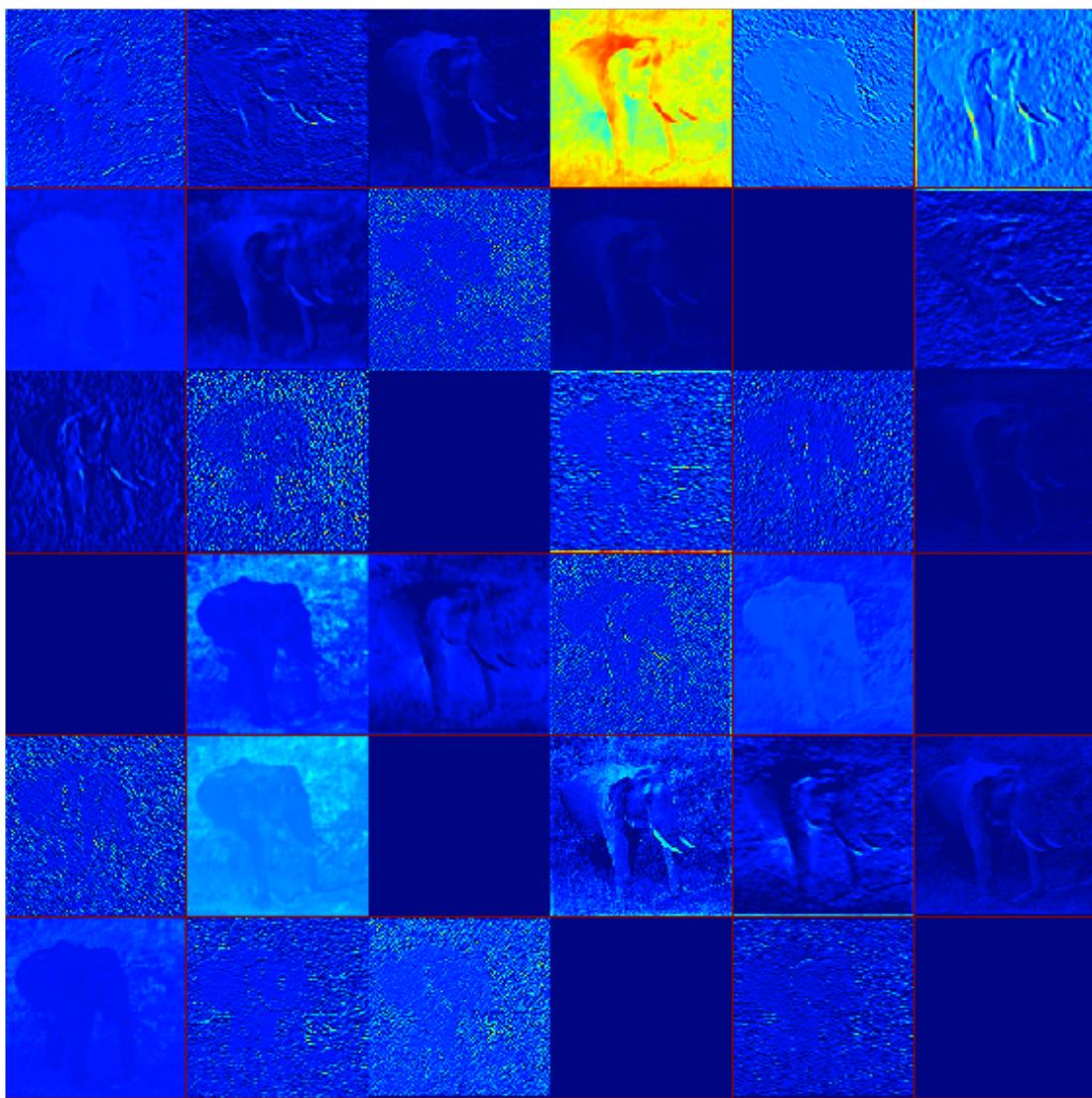
```
res4b15_branch2c      (1, 1024, 14, 14)
res4b15(1, 1024, 14, 14)
res4b15_res4b15_relu_0_split_0      (1, 1024, 14, 14)
res4b15_res4b15_relu_0_split_1      (1, 1024, 14, 14)
res4b16_branch2a      (1, 256, 14, 14)
res4b16_branch2b      (1, 256, 14, 14)
res4b16_branch2c      (1, 1024, 14, 14)
res4b16(1, 1024, 14, 14)
res4b16_res4b16_relu_0_split_0      (1, 1024, 14, 14)
res4b16_res4b16_relu_0_split_1      (1, 1024, 14, 14)
res4b17_branch2a      (1, 256, 14, 14)
res4b17_branch2b      (1, 256, 14, 14)
res4b17_branch2c      (1, 1024, 14, 14)
res4b17(1, 1024, 14, 14)
res4b17_res4b17_relu_0_split_0      (1, 1024, 14, 14)
res4b17_res4b17_relu_0_split_1      (1, 1024, 14, 14)
res4b18_branch2a      (1, 256, 14, 14)
res4b18_branch2b      (1, 256, 14, 14)
res4b18_branch2c      (1, 1024, 14, 14)
res4b18(1, 1024, 14, 14)
res4b18_res4b18_relu_0_split_0      (1, 1024, 14, 14)
res4b18_res4b18_relu_0_split_1      (1, 1024, 14, 14)
res4b19_branch2a      (1, 256, 14, 14)
res4b19_branch2b      (1, 256, 14, 14)
res4b19_branch2c      (1, 1024, 14, 14)
res4b19(1, 1024, 14, 14)
res4b19_res4b19_relu_0_split_0      (1, 1024, 14, 14)
res4b19_res4b19_relu_0_split_1      (1, 1024, 14, 14)
res4b20_branch2a      (1, 256, 14, 14)
res4b20_branch2b      (1, 256, 14, 14)
res4b20_branch2c      (1, 1024, 14, 14)
res4b20(1, 1024, 14, 14)
res4b20_res4b20_relu_0_split_0      (1, 1024, 14, 14)
res4b20_res4b20_relu_0_split_1      (1, 1024, 14, 14)
res4b21_branch2a      (1, 256, 14, 14)
res4b21_branch2b      (1, 256, 14, 14)
res4b21_branch2c      (1, 1024, 14, 14)
res4b21(1, 1024, 14, 14)
res4b21_res4b21_relu_0_split_0      (1, 1024, 14, 14)
res4b21_res4b21_relu_0_split_1      (1, 1024, 14, 14)
res4b22_branch2a      (1, 256, 14, 14)
res4b22_branch2b      (1, 256, 14, 14)
res4b22_branch2c      (1, 1024, 14, 14)
res4b22(1, 1024, 14, 14)
res4b22_res4b22_relu_0_split_0      (1, 1024, 14, 14)
res4b22_res4b22_relu_0_split_1      (1, 1024, 14, 14)
res5a_branch1(1, 2048, 7, 7)
res5a_branch2a(1, 512, 7, 7)
res5a_branch2b(1, 512, 7, 7)
res5a_branch2c(1, 2048, 7, 7)
res5a(1, 2048, 7, 7)
res5a_res5a_relu_0_split_0      (1, 2048, 7, 7)
res5a_res5a_relu_0_split_1      (1, 2048, 7, 7)
res5b_branch2a(1, 512, 7, 7)
res5b_branch2b(1, 512, 7, 7)
```

```
res5b_branch2c (1, 2048, 7, 7)
res5b (1, 2048, 7, 7)
res5b_res5b_relu_0_split_0 (1, 2048, 7, 7)
res5b_res5b_relu_0_split_1 (1, 2048, 7, 7)
res5c_branch2a (1, 512, 7, 7)
res5c_branch2b (1, 512, 7, 7)
res5c_branch2c (1, 2048, 7, 7)
res5c (1, 2048, 7, 7)
pool5 (1, 2048, 1, 1)
fc1000 (1, 1000)
prob (1, 1000)
```

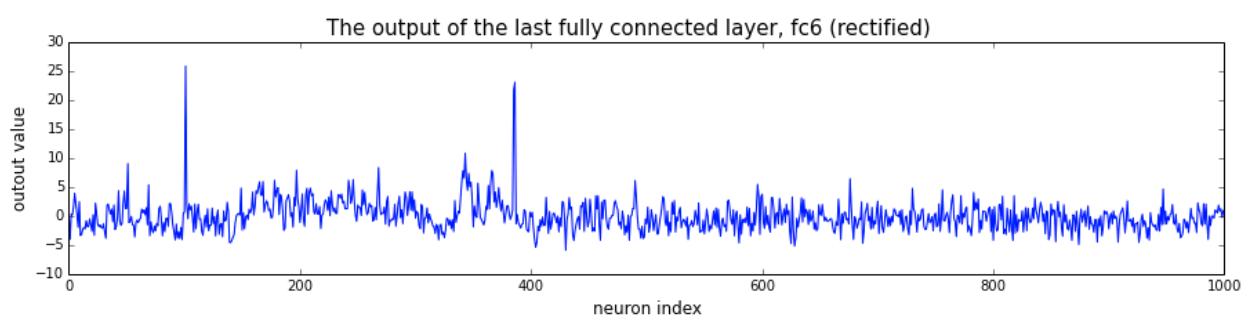
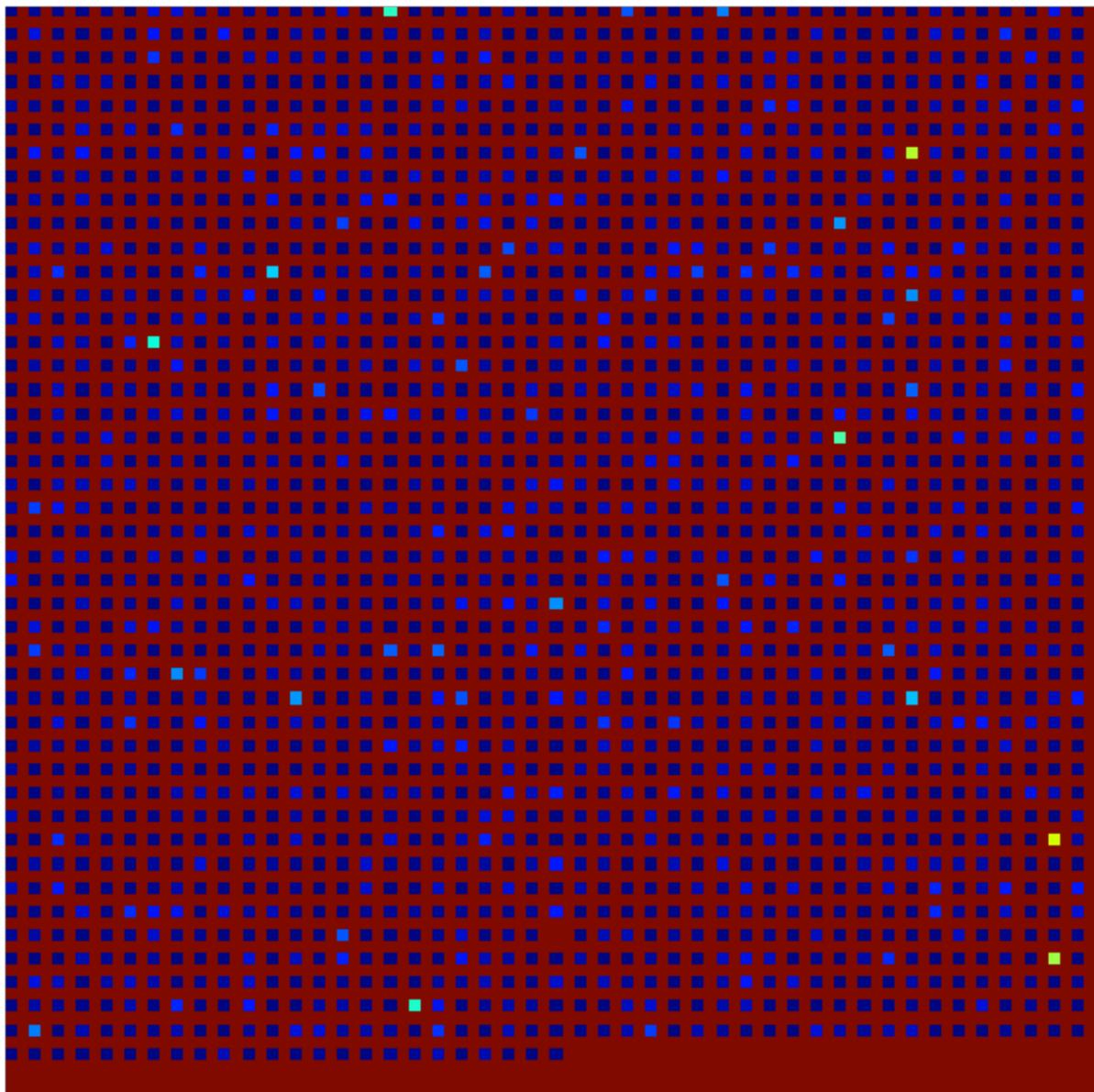
the parameters (weights) of the first conv Layer

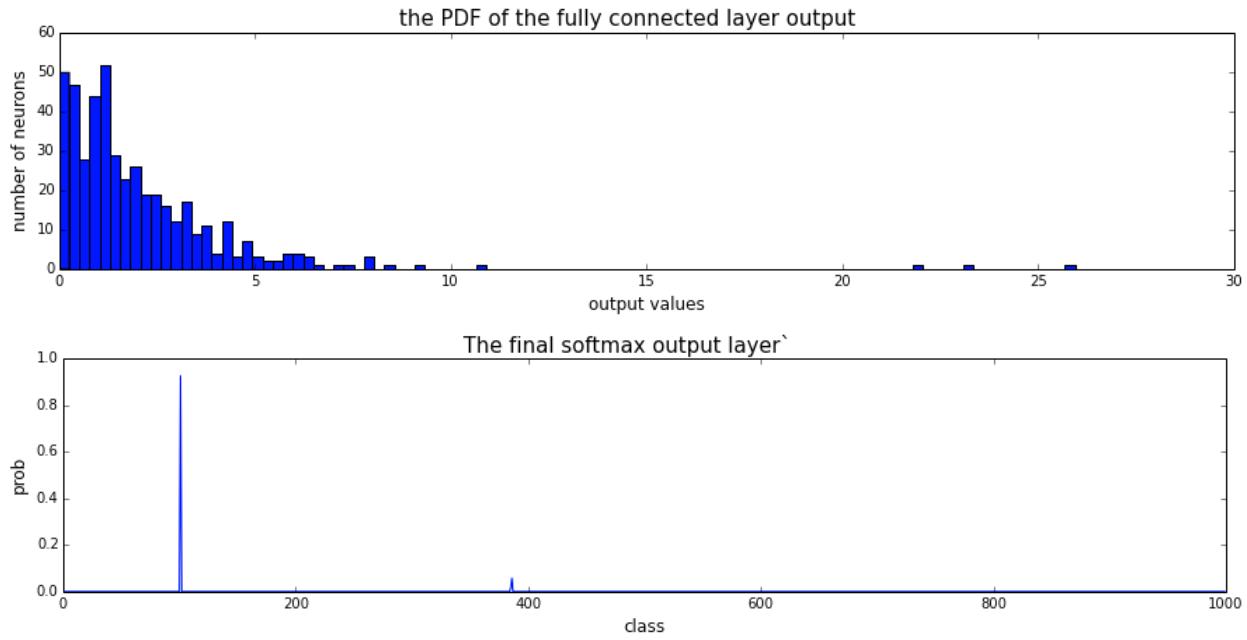


The first layer output, conv1 (rectified responses of the filters above, first 36 only)



The last layer of pooling, pool5/7x7\_s1





### ResNet-152:

Network data shape visualization

```

data      (1, 3, 224, 224)
conv1    (1, 64, 112, 112)
pool1    (1, 64, 56, 56)
pool1_pool1_0_split_0 (1, 64, 56, 56)
pool1_pool1_0_split_1 (1, 64, 56, 56)
res2a_branch1 (1, 256, 56, 56)
res2a_branch2a (1, 64, 56, 56)
res2a_branch2b (1, 64, 56, 56)
res2a_branch2c (1, 256, 56, 56)
res2a    (1, 256, 56, 56)
res2a_res2a_relu_0_split_0   (1, 256, 56, 56)
res2a_res2a_relu_0_split_1   (1, 256, 56, 56)
res2b_branch2a (1, 64, 56, 56)
res2b_branch2b (1, 64, 56, 56)
res2b_branch2c (1, 256, 56, 56)
res2b    (1, 256, 56, 56)
res2b_res2b_relu_0_split_0   (1, 256, 56, 56)
res2b_res2b_relu_0_split_1   (1, 256, 56, 56)
res2c_branch2a (1, 64, 56, 56)
res2c_branch2b (1, 64, 56, 56)
res2c_branch2c (1, 256, 56, 56)
res2c    (1, 256, 56, 56)
res2c_res2c_relu_0_split_0   (1, 256, 56, 56)
res2c_res2c_relu_0_split_1   (1, 256, 56, 56)
res3a_branch1 (1, 512, 28, 28)
res3a_branch2a (1, 128, 28, 28)
res3a_branch2b (1, 128, 28, 28)
res3a_branch2c (1, 512, 28, 28)
res3a    (1, 512, 28, 28)
res3a_res3a_relu_0_split_0   (1, 512, 28, 28)
res3a_res3a_relu_0_split_1   (1, 512, 28, 28)
res3b1_branch2a   (1, 128, 28, 28)
```

```
res3b1_branch2b      (1, 128, 28, 28)
res3b1_branch2c      (1, 512, 28, 28)
res3b1 (1, 512, 28, 28)
res3b1_res3b1_relu_0_split_0 (1, 512, 28, 28)
res3b1_res3b1_relu_0_split_1 (1, 512, 28, 28)
res3b2_branch2a      (1, 128, 28, 28)
res3b2_branch2b      (1, 128, 28, 28)
res3b2_branch2c      (1, 512, 28, 28)
res3b2 (1, 512, 28, 28)
res3b2_res3b2_relu_0_split_0 (1, 512, 28, 28)
res3b2_res3b2_relu_0_split_1 (1, 512, 28, 28)
res3b3_branch2a      (1, 128, 28, 28)
res3b3_branch2b      (1, 128, 28, 28)
res3b3_branch2c      (1, 512, 28, 28)
res3b3 (1, 512, 28, 28)
res3b3_res3b3_relu_0_split_0 (1, 512, 28, 28)
res3b3_res3b3_relu_0_split_1 (1, 512, 28, 28)
res3b4_branch2a      (1, 128, 28, 28)
res3b4_branch2b      (1, 128, 28, 28)
res3b4_branch2c      (1, 512, 28, 28)
res3b4 (1, 512, 28, 28)
res3b4_res3b4_relu_0_split_0 (1, 512, 28, 28)
res3b4_res3b4_relu_0_split_1 (1, 512, 28, 28)
res3b5_branch2a      (1, 128, 28, 28)
res3b5_branch2b      (1, 128, 28, 28)
res3b5_branch2c      (1, 512, 28, 28)
res3b5 (1, 512, 28, 28)
res3b5_res3b5_relu_0_split_0 (1, 512, 28, 28)
res3b5_res3b5_relu_0_split_1 (1, 512, 28, 28)
res3b6_branch2a      (1, 128, 28, 28)
res3b6_branch2b      (1, 128, 28, 28)
res3b6_branch2c      (1, 512, 28, 28)
res3b6 (1, 512, 28, 28)
res3b6_res3b6_relu_0_split_0 (1, 512, 28, 28)
res3b6_res3b6_relu_0_split_1 (1, 512, 28, 28)
res3b7_branch2a      (1, 128, 28, 28)
res3b7_branch2b      (1, 128, 28, 28)
res3b7_branch2c      (1, 512, 28, 28)
res3b7 (1, 512, 28, 28)
res3b7_res3b7_relu_0_split_0 (1, 512, 28, 28)
res3b7_res3b7_relu_0_split_1 (1, 512, 28, 28)
res4a_branch1 (1, 1024, 14, 14)
res4a_branch2a (1, 256, 14, 14)
res4a_branch2b (1, 256, 14, 14)
res4a_branch2c (1, 1024, 14, 14)
res4a (1, 1024, 14, 14)
res4a_res4a_relu_0_split_0 (1, 1024, 14, 14)
res4a_res4a_relu_0_split_1 (1, 1024, 14, 14)
res4b1_branch2a      (1, 256, 14, 14)
res4b1_branch2b      (1, 256, 14, 14)
res4b1_branch2c      (1, 1024, 14, 14)
res4b1 (1, 1024, 14, 14)
res4b1_res4b1_relu_0_split_0 (1, 1024, 14, 14)
res4b1_res4b1_relu_0_split_1 (1, 1024, 14, 14)
res4b2_branch2a      (1, 256, 14, 14)
```

```
res4b2_branch2b      (1, 256, 14, 14)
res4b2_branch2c      (1, 1024, 14, 14)
res4b2 (1, 1024, 14, 14)
res4b2_res4b2_relu_0_split_0 (1, 1024, 14, 14)
res4b2_res4b2_relu_0_split_1 (1, 1024, 14, 14)
res4b3_branch2a      (1, 256, 14, 14)
res4b3_branch2b      (1, 256, 14, 14)
res4b3_branch2c      (1, 1024, 14, 14)
res4b3 (1, 1024, 14, 14)
res4b3_res4b3_relu_0_split_0 (1, 1024, 14, 14)
res4b3_res4b3_relu_0_split_1 (1, 1024, 14, 14)
res4b4_branch2a      (1, 256, 14, 14)
res4b4_branch2b      (1, 256, 14, 14)
res4b4_branch2c      (1, 1024, 14, 14)
res4b4 (1, 1024, 14, 14)
res4b4_res4b4_relu_0_split_0 (1, 1024, 14, 14)
res4b4_res4b4_relu_0_split_1 (1, 1024, 14, 14)
res4b5_branch2a      (1, 256, 14, 14)
res4b5_branch2b      (1, 256, 14, 14)
res4b5_branch2c      (1, 1024, 14, 14)
res4b5 (1, 1024, 14, 14)
res4b5_res4b5_relu_0_split_0 (1, 1024, 14, 14)
res4b5_res4b5_relu_0_split_1 (1, 1024, 14, 14)
res4b6_branch2a      (1, 256, 14, 14)
res4b6_branch2b      (1, 256, 14, 14)
res4b6_branch2c      (1, 1024, 14, 14)
res4b6 (1, 1024, 14, 14)
res4b6_res4b6_relu_0_split_0 (1, 1024, 14, 14)
res4b6_res4b6_relu_0_split_1 (1, 1024, 14, 14)
res4b7_branch2a      (1, 256, 14, 14)
res4b7_branch2b      (1, 256, 14, 14)
res4b7_branch2c      (1, 1024, 14, 14)
res4b7 (1, 1024, 14, 14)
res4b7_res4b7_relu_0_split_0 (1, 1024, 14, 14)
res4b7_res4b7_relu_0_split_1 (1, 1024, 14, 14)
res4b8_branch2a      (1, 256, 14, 14)
res4b8_branch2b      (1, 256, 14, 14)
res4b8_branch2c      (1, 1024, 14, 14)
res4b8 (1, 1024, 14, 14)
res4b8_res4b8_relu_0_split_0 (1, 1024, 14, 14)
res4b8_res4b8_relu_0_split_1 (1, 1024, 14, 14)
res4b9_branch2a      (1, 256, 14, 14)
res4b9_branch2b      (1, 256, 14, 14)
res4b9_branch2c      (1, 1024, 14, 14)
res4b9 (1, 1024, 14, 14)
res4b9_res4b9_relu_0_split_0 (1, 1024, 14, 14)
res4b9_res4b9_relu_0_split_1 (1, 1024, 14, 14)
res4b10_branch2a     (1, 256, 14, 14)
res4b10_branch2b     (1, 256, 14, 14)
res4b10_branch2c     (1, 1024, 14, 14)
res4b10 (1, 1024, 14, 14)
res4b10_res4b10_relu_0_split_0 (1, 1024, 14, 14)
res4b10_res4b10_relu_0_split_1 (1, 1024, 14, 14)
res4b11_branch2a     (1, 256, 14, 14)
res4b11_branch2b     (1, 256, 14, 14)
```

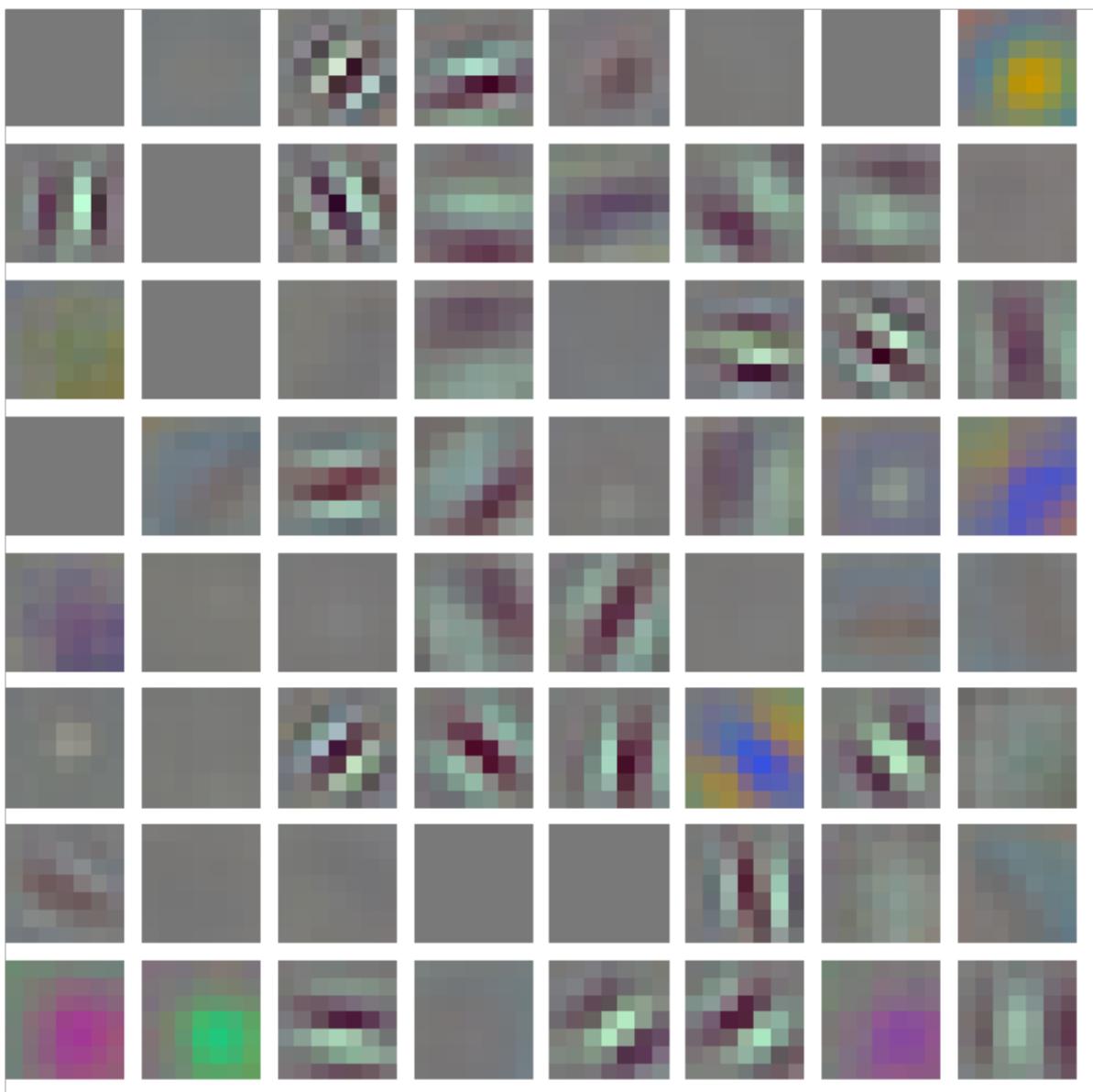
```
res4b11_branch2c      (1, 1024, 14, 14)
res4b11(1, 1024, 14, 14)
res4b11_res4b11_relu_0_split_0      (1, 1024, 14, 14)
res4b11_res4b11_relu_0_split_1      (1, 1024, 14, 14)
res4b12_branch2a      (1, 256, 14, 14)
res4b12_branch2b      (1, 256, 14, 14)
res4b12_branch2c      (1, 1024, 14, 14)
res4b12(1, 1024, 14, 14)
res4b12_res4b12_relu_0_split_0      (1, 1024, 14, 14)
res4b12_res4b12_relu_0_split_1      (1, 1024, 14, 14)
res4b13_branch2a      (1, 256, 14, 14)
res4b13_branch2b      (1, 256, 14, 14)
res4b13_branch2c      (1, 1024, 14, 14)
res4b13(1, 1024, 14, 14)
res4b13_res4b13_relu_0_split_0      (1, 1024, 14, 14)
res4b13_res4b13_relu_0_split_1      (1, 1024, 14, 14)
res4b14_branch2a      (1, 256, 14, 14)
res4b14_branch2b      (1, 256, 14, 14)
res4b14_branch2c      (1, 1024, 14, 14)
res4b14(1, 1024, 14, 14)
res4b14_res4b14_relu_0_split_0      (1, 1024, 14, 14)
res4b14_res4b14_relu_0_split_1      (1, 1024, 14, 14)
res4b15_branch2a      (1, 256, 14, 14)
res4b15_branch2b      (1, 256, 14, 14)
res4b15_branch2c      (1, 1024, 14, 14)
res4b15(1, 1024, 14, 14)
res4b15_res4b15_relu_0_split_0      (1, 1024, 14, 14)
res4b15_res4b15_relu_0_split_1      (1, 1024, 14, 14)
res4b16_branch2a      (1, 256, 14, 14)
res4b16_branch2b      (1, 256, 14, 14)
res4b16_branch2c      (1, 1024, 14, 14)
res4b16(1, 1024, 14, 14)
res4b16_res4b16_relu_0_split_0      (1, 1024, 14, 14)
res4b16_res4b16_relu_0_split_1      (1, 1024, 14, 14)
res4b17_branch2a      (1, 256, 14, 14)
res4b17_branch2b      (1, 256, 14, 14)
res4b17_branch2c      (1, 1024, 14, 14)
res4b17(1, 1024, 14, 14)
res4b17_res4b17_relu_0_split_0      (1, 1024, 14, 14)
res4b17_res4b17_relu_0_split_1      (1, 1024, 14, 14)
res4b18_branch2a      (1, 256, 14, 14)
res4b18_branch2b      (1, 256, 14, 14)
res4b18_branch2c      (1, 1024, 14, 14)
res4b18(1, 1024, 14, 14)
res4b18_res4b18_relu_0_split_0      (1, 1024, 14, 14)
res4b18_res4b18_relu_0_split_1      (1, 1024, 14, 14)
res4b19_branch2a      (1, 256, 14, 14)
res4b19_branch2b      (1, 256, 14, 14)
res4b19_branch2c      (1, 1024, 14, 14)
res4b19(1, 1024, 14, 14)
res4b19_res4b19_relu_0_split_0      (1, 1024, 14, 14)
res4b19_res4b19_relu_0_split_1      (1, 1024, 14, 14)
res4b20_branch2a      (1, 256, 14, 14)
res4b20_branch2b      (1, 256, 14, 14)
res4b20_branch2c      (1, 1024, 14, 14)
```

```
res4b20 (1, 1024, 14, 14)
res4b20_res4b20_relu_0_split_0      (1, 1024, 14, 14)
res4b20_res4b20_relu_0_split_1      (1, 1024, 14, 14)
res4b21_branch2a      (1, 256, 14, 14)
res4b21_branch2b      (1, 256, 14, 14)
res4b21_branch2c      (1, 1024, 14, 14)
res4b21 (1, 1024, 14, 14)
res4b21_res4b21_relu_0_split_0      (1, 1024, 14, 14)
res4b21_res4b21_relu_0_split_1      (1, 1024, 14, 14)
res4b22_branch2a      (1, 256, 14, 14)
res4b22_branch2b      (1, 256, 14, 14)
res4b22_branch2c      (1, 1024, 14, 14)
res4b22 (1, 1024, 14, 14)
res4b22_res4b22_relu_0_split_0      (1, 1024, 14, 14)
res4b22_res4b22_relu_0_split_1      (1, 1024, 14, 14)
res4b23_branch2a      (1, 256, 14, 14)
res4b23_branch2b      (1, 256, 14, 14)
res4b23_branch2c      (1, 1024, 14, 14)
res4b23 (1, 1024, 14, 14)
res4b23_res4b23_relu_0_split_0      (1, 1024, 14, 14)
res4b23_res4b23_relu_0_split_1      (1, 1024, 14, 14)
res4b24_branch2a      (1, 256, 14, 14)
res4b24_branch2b      (1, 256, 14, 14)
res4b24_branch2c      (1, 1024, 14, 14)
res4b24 (1, 1024, 14, 14)
res4b24_res4b24_relu_0_split_0      (1, 1024, 14, 14)
res4b24_res4b24_relu_0_split_1      (1, 1024, 14, 14)
res4b25_branch2a      (1, 256, 14, 14)
res4b25_branch2b      (1, 256, 14, 14)
res4b25_branch2c      (1, 1024, 14, 14)
res4b25 (1, 1024, 14, 14)
res4b25_res4b25_relu_0_split_0      (1, 1024, 14, 14)
res4b25_res4b25_relu_0_split_1      (1, 1024, 14, 14)
res4b26_branch2a      (1, 256, 14, 14)
res4b26_branch2b      (1, 256, 14, 14)
res4b26_branch2c      (1, 1024, 14, 14)
res4b26 (1, 1024, 14, 14)
res4b26_res4b26_relu_0_split_0      (1, 1024, 14, 14)
res4b26_res4b26_relu_0_split_1      (1, 1024, 14, 14)
res4b27_branch2a      (1, 256, 14, 14)
res4b27_branch2b      (1, 256, 14, 14)
res4b27_branch2c      (1, 1024, 14, 14)
res4b27 (1, 1024, 14, 14)
res4b27_res4b27_relu_0_split_0      (1, 1024, 14, 14)
res4b27_res4b27_relu_0_split_1      (1, 1024, 14, 14)
res4b28_branch2a      (1, 256, 14, 14)
res4b28_branch2b      (1, 256, 14, 14)
res4b28_branch2c      (1, 1024, 14, 14)
res4b28 (1, 1024, 14, 14)
res4b28_res4b28_relu_0_split_0      (1, 1024, 14, 14)
res4b28_res4b28_relu_0_split_1      (1, 1024, 14, 14)
res4b29_branch2a      (1, 256, 14, 14)
res4b29_branch2b      (1, 256, 14, 14)
res4b29_branch2c      (1, 1024, 14, 14)
res4b29 (1, 1024, 14, 14)
```

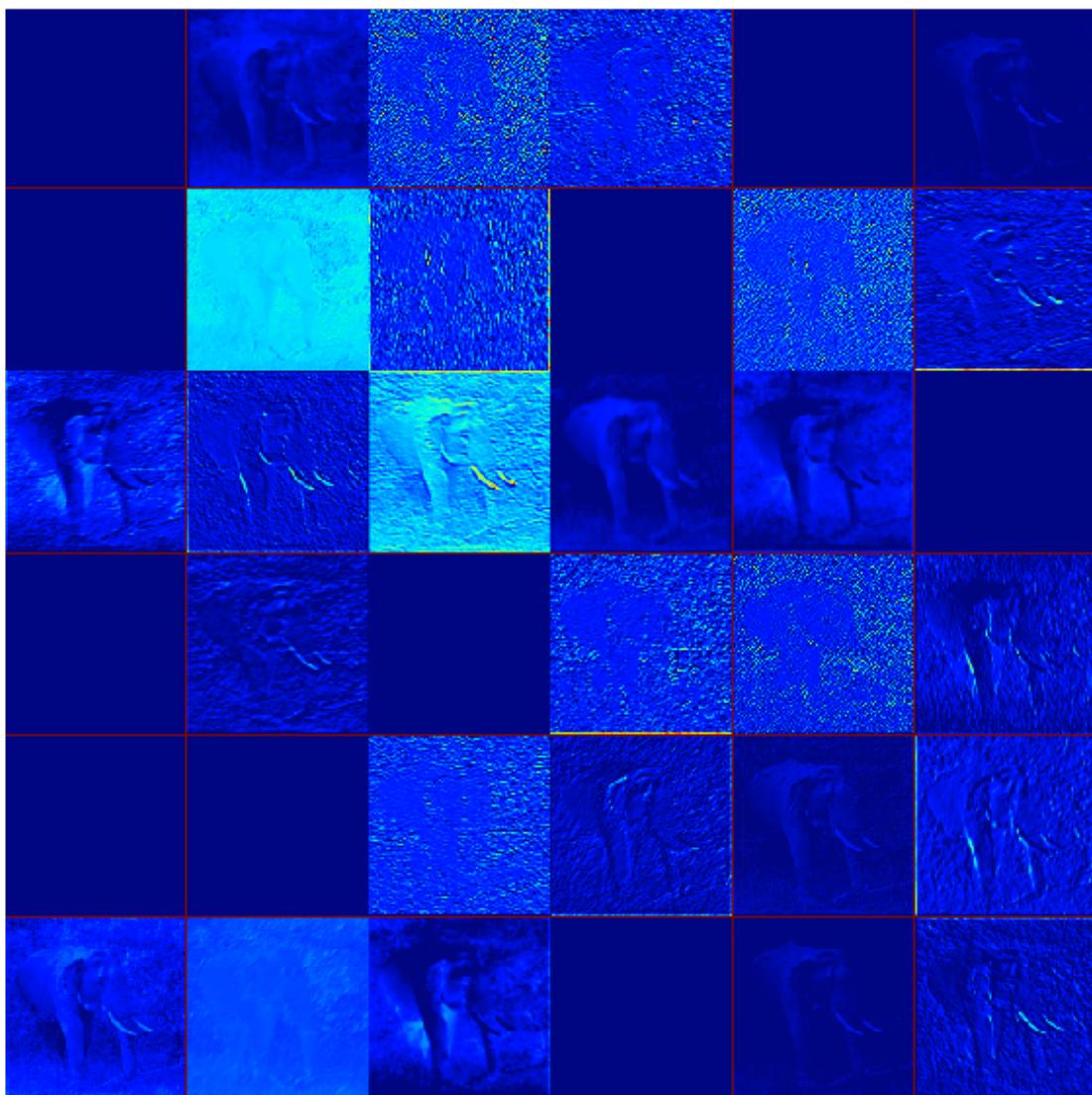
```
res4b29_res4b29_relu_0_split_0      (1, 1024, 14, 14)
res4b29_res4b29_relu_0_split_1      (1, 1024, 14, 14)
res4b30_branch2a      (1, 256, 14, 14)
res4b30_branch2b      (1, 256, 14, 14)
res4b30_branch2c      (1, 1024, 14, 14)
res4b30(1, 1024, 14, 14)
res4b30_res4b30_relu_0_split_0      (1, 1024, 14, 14)
res4b30_res4b30_relu_0_split_1      (1, 1024, 14, 14)
res4b31_branch2a      (1, 256, 14, 14)
res4b31_branch2b      (1, 256, 14, 14)
res4b31_branch2c      (1, 1024, 14, 14)
res4b31(1, 1024, 14, 14)
res4b31_res4b31_relu_0_split_0      (1, 1024, 14, 14)
res4b31_res4b31_relu_0_split_1      (1, 1024, 14, 14)
res4b32_branch2a      (1, 256, 14, 14)
res4b32_branch2b      (1, 256, 14, 14)
res4b32_branch2c      (1, 1024, 14, 14)
res4b32(1, 1024, 14, 14)
res4b32_res4b32_relu_0_split_0      (1, 1024, 14, 14)
res4b32_res4b32_relu_0_split_1      (1, 1024, 14, 14)
res4b33_branch2a      (1, 256, 14, 14)
res4b33_branch2b      (1, 256, 14, 14)
res4b33_branch2c      (1, 1024, 14, 14)
res4b33(1, 1024, 14, 14)
res4b33_res4b33_relu_0_split_0      (1, 1024, 14, 14)
res4b33_res4b33_relu_0_split_1      (1, 1024, 14, 14)
res4b34_branch2a      (1, 256, 14, 14)
res4b34_branch2b      (1, 256, 14, 14)
res4b34_branch2c      (1, 1024, 14, 14)
res4b34(1, 1024, 14, 14)
res4b34_res4b34_relu_0_split_0      (1, 1024, 14, 14)
res4b34_res4b34_relu_0_split_1      (1, 1024, 14, 14)
res4b35_branch2a      (1, 256, 14, 14)
res4b35_branch2b      (1, 256, 14, 14)
res4b35_branch2c      (1, 1024, 14, 14)
res4b35(1, 1024, 14, 14)
res4b35_res4b35_relu_0_split_0      (1, 1024, 14, 14)
res4b35_res4b35_relu_0_split_1      (1, 1024, 14, 14)
res5a_branch1 (1, 2048, 7, 7)
res5a_branch2a (1, 512, 7, 7)
res5a_branch2b (1, 512, 7, 7)
res5a_branch2c (1, 2048, 7, 7)
res5a (1, 2048, 7, 7)
res5a_res5a_relu_0_split_0 (1, 2048, 7, 7)
res5a_res5a_relu_0_split_1 (1, 2048, 7, 7)
res5b_branch2a (1, 512, 7, 7)
res5b_branch2b (1, 512, 7, 7)
res5b_branch2c (1, 2048, 7, 7)
res5b (1, 2048, 7, 7)
res5b_res5b_relu_0_split_0 (1, 2048, 7, 7)
res5b_res5b_relu_0_split_1 (1, 2048, 7, 7)
res5c_branch2a (1, 512, 7, 7)
res5c_branch2b (1, 512, 7, 7)
res5c_branch2c (1, 2048, 7, 7)
res5c (1, 2048, 7, 7)
```

```
pool5  (1, 2048, 1, 1)
fc1000 (1, 1000)
prob   (1, 1000)
```

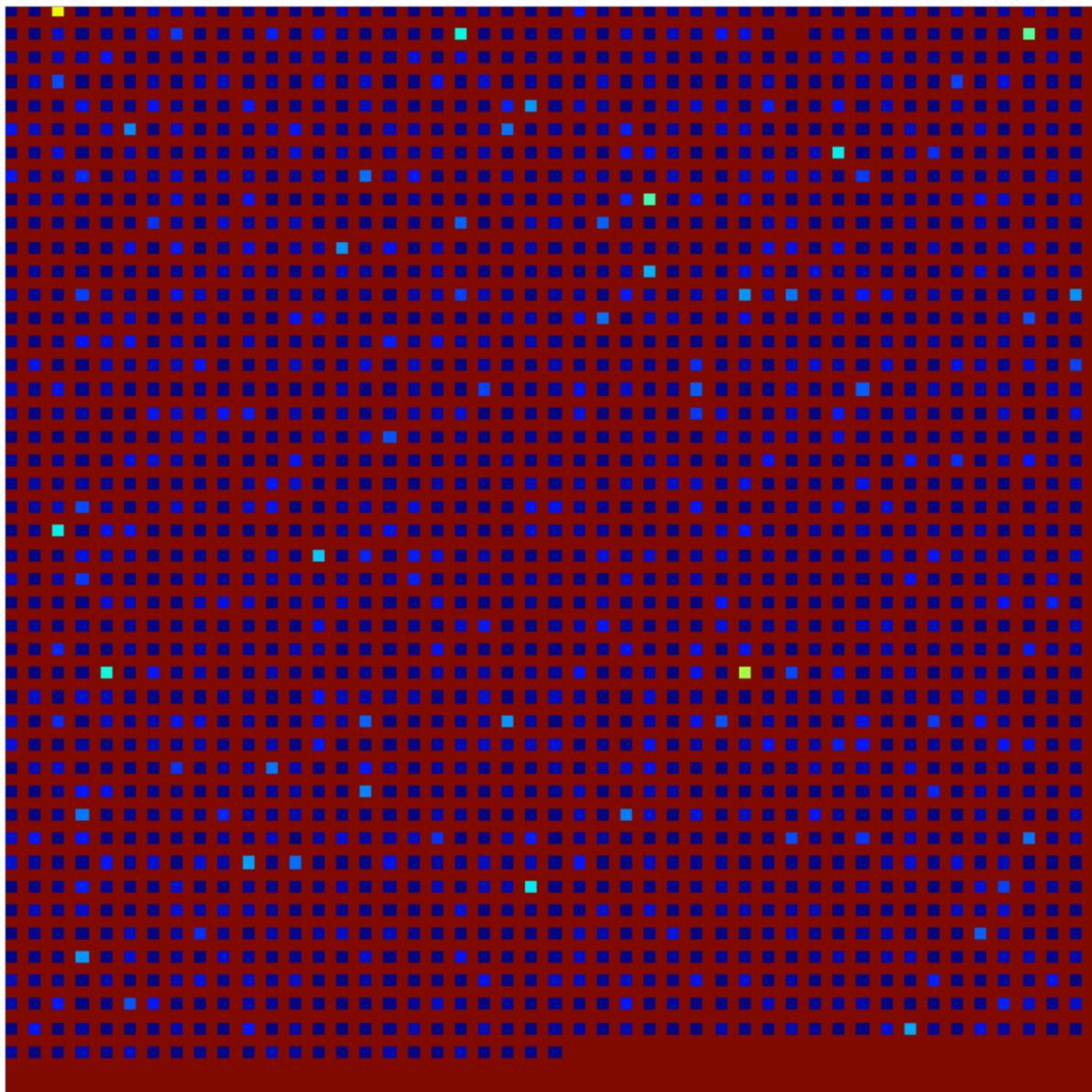
the parameters (weights) of the first conv Layer



The first layer output, conv1 (rectified responses of the filters above, first 36 only)



The last layer of pooling, pool5/7x7\_s1



The output of the last fully connected layer, fc6 (rectified)

