

训练.....	1
测试.....	7

% **data** 分为训练样本集和测试样本集，其中训练样本集中每个标签包含一张图片，测试样本集包含剩下的图片

```
data_dir = '..\data'; % Set this to the location of your data directory
```

```
[train_samples,test_samples] = sample_select(data_dir);
```

```
disp = 1;
```

% 制作滤波器组

```
fb = makeFilterbank();
```

训练

对样本集中的每个图片，求图片的滤波器组响应,返回样本集的响应

```
samples_res = applyFilterbank(train_samples,fb);
```

% 样本集的响应中，对每个像素进行 **k-means** 聚类，**k=20**，推荐使用 **kmeans** 函数

```
k = 20;
```

```
tic;
```

```
[samples_cluster,centers] = applyKmeans(samples_res,k);
```

```
t = toc;
```

% 求取样本集中每个图片的 **texton** 直方图

```
samples_hist = applyHistogram(samples_cluster);
```

```
if disp == 1
```

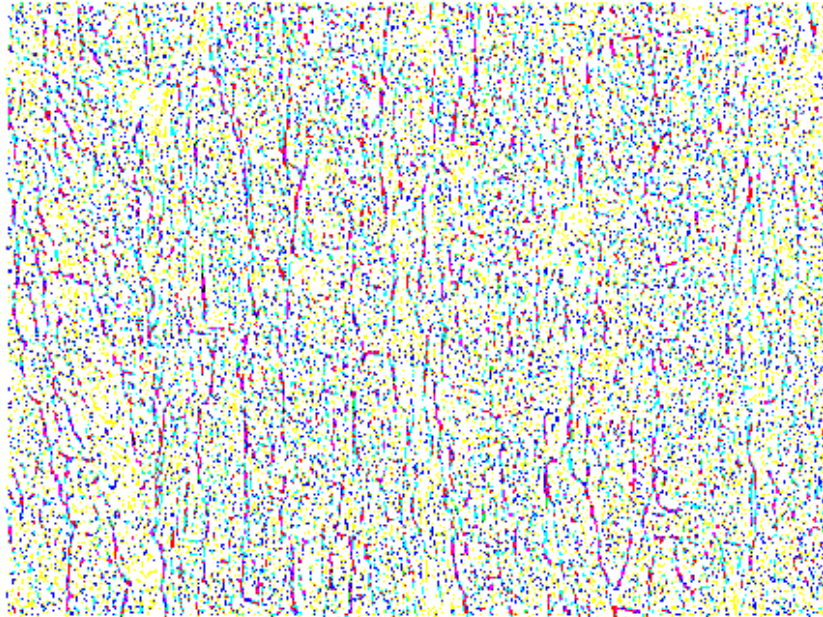
% 可视化聚类结果

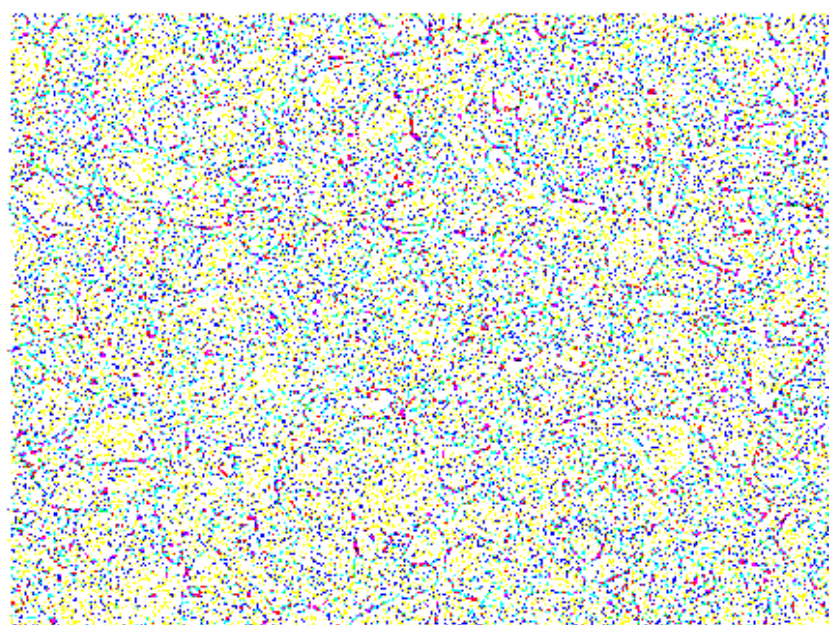
```
visualizeSamples(samples_cluster)
```

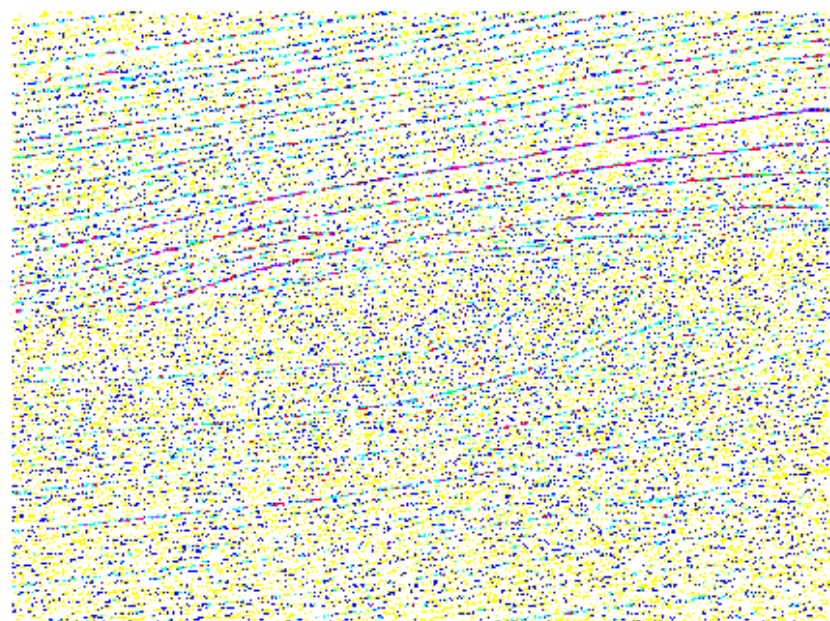
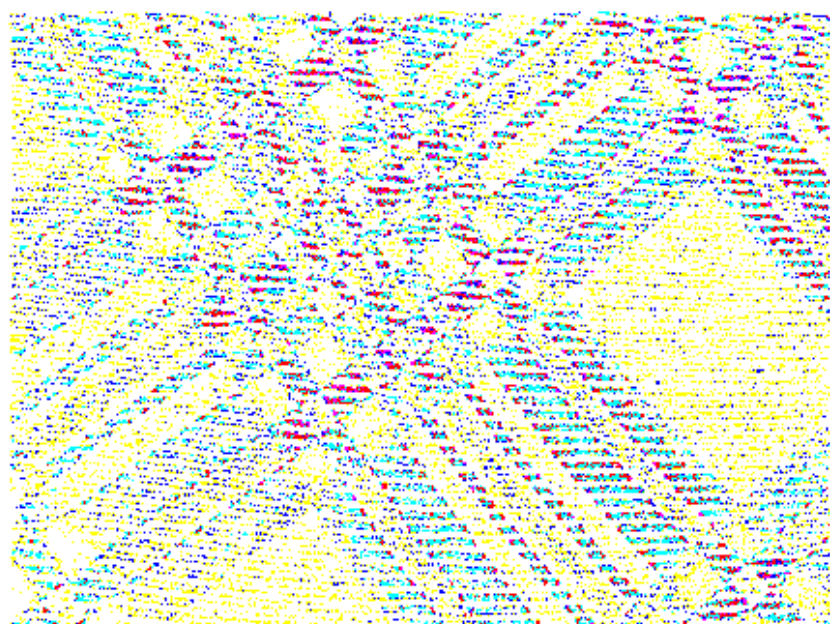
```
%可视化样本集中每个图片的直方图
```

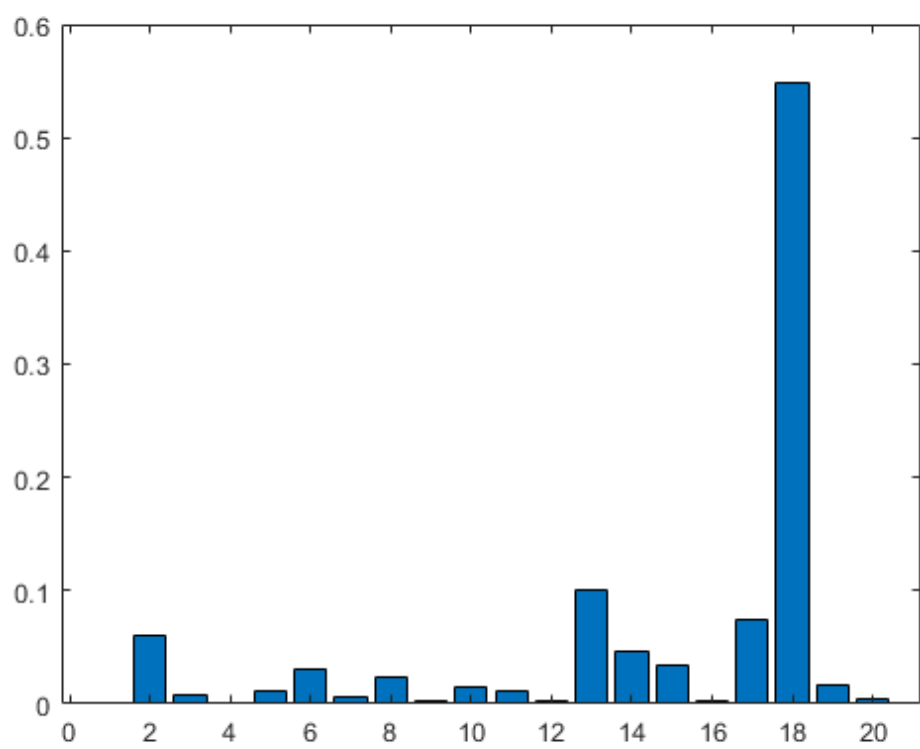
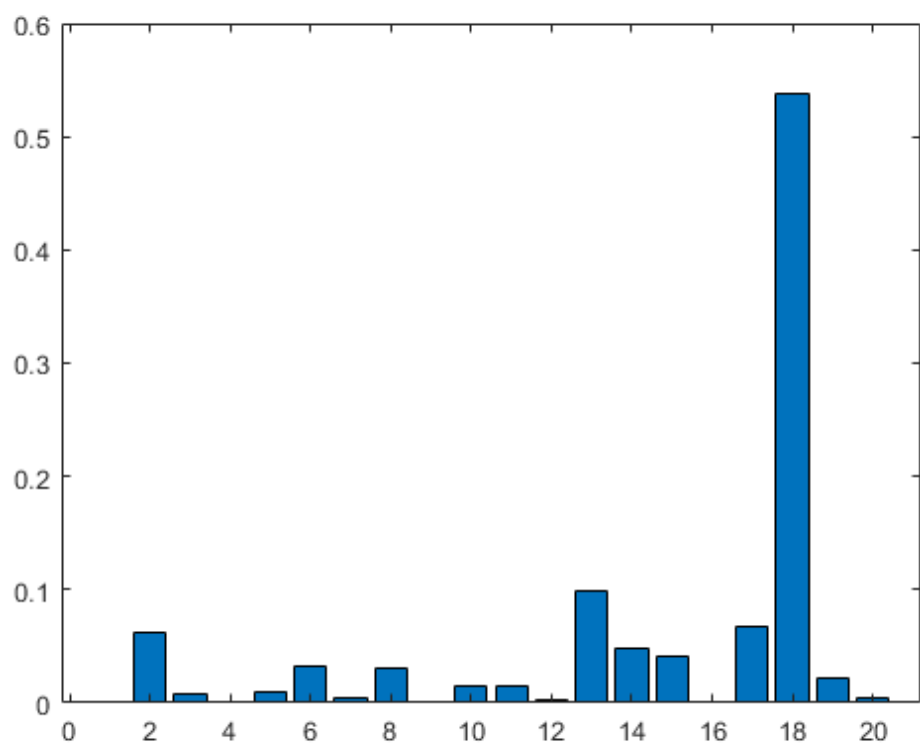
```
visualizeHistogram(samples_hist);
```

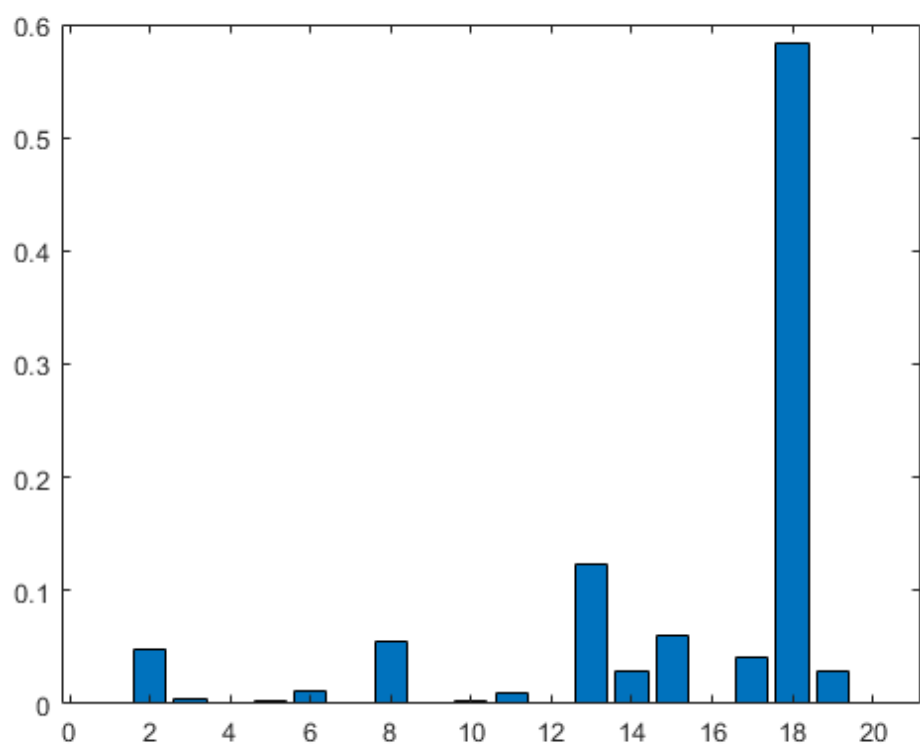
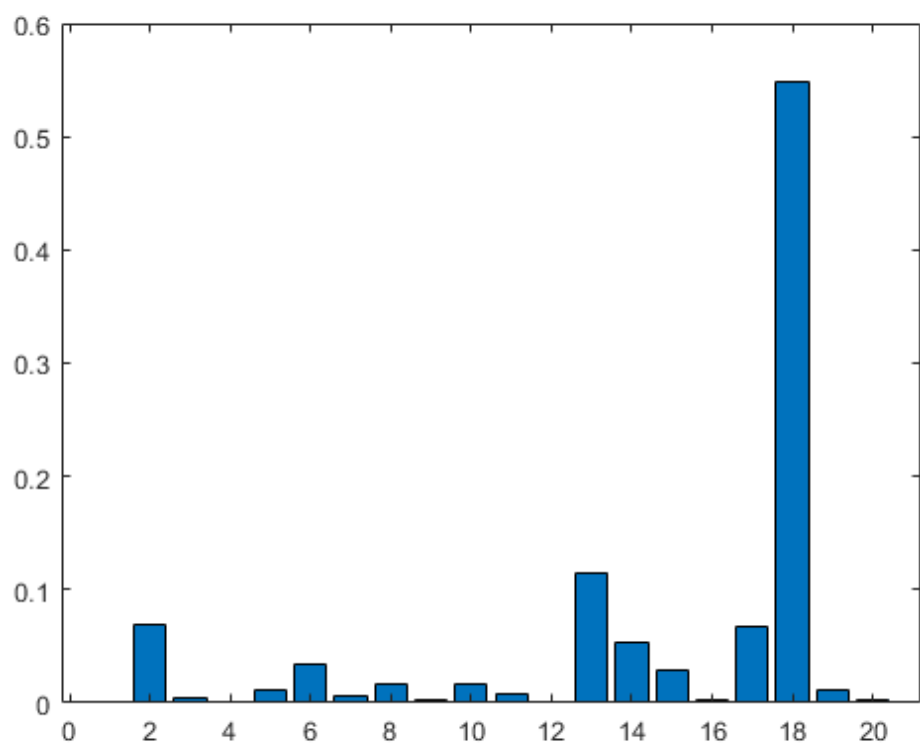
```
end
```

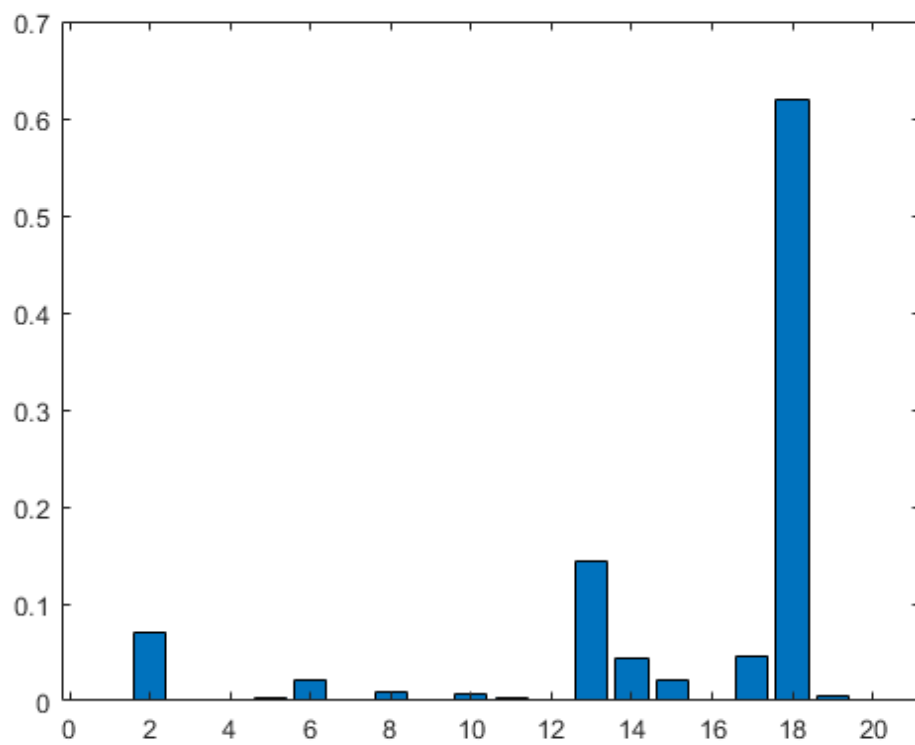












测试

对测试集中的每个图片，求滤波器组响应

```
test_samples_res = applyFilterbank(test_samples,fb);
```

```
% 对每个像素进行归类,推荐使用 pdist2 函数
```

```
test_samples_cluster = predictClasses(test_samples_res,centers);
```

```
% 求取测试图片的 texon 直方图
```

```
test_samples_hist = applyHistogram(test_samples_cluster);
```

```
% 最近邻算法预测类别
```

```
test_samples_predict = predictLabels(test_samples_hist,samples_hist);
```

```
% 计算准确率
```

```
accuracy = evalPredict(test_samples_predict,test_samples);
```

Published with MATLAB® R2018b