

```
In [1]: import os
import numpy as np
import pandas as pd
from skimage import io, color, exposure, filters
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans, BisectingKMeans, SpectralClustering, Agglomerative
from sklearn.metrics import fowlkes_mallows_score, silhouette_score

import warnings
warnings.filterwarnings("ignore")
```

```
In [2]: crop_folder=r'Cropped'
classes=os.listdir(crop_folder)
def edge_histogram(images):
    imgs=[]
    classs=[]
    for idx,pt in enumerate(images):
        path = os.path.join(crop_folder,pt)
        for img in os.listdir(path):
            src_pt = os.path.join(path, img)
            image = io.imread(src_pt)
            gray_img = color.rgb2gray(image)
            dx, dy = filters.sobel_h(gray_img), filters.sobel_v(gray_img)
            angle_sobel = np.mod(np.arctan2(dy, dx), np.pi)
            hist, bins = exposure.histogram(angle_sobel, nbins=36)
            hist=hist/np.sum(hist)
            imgs.append(hist)
            classs.append(idx)

    imgs=np.array(imgs)
    classs=np.array(classs)
    return imgs,classs

imgs,classs=edge_histogram(classes)
```

```
In [3]: pca=PCA(n_components=2)
imgs=pca.fit_transform(imgs)
```

```
In [5]: #K-mean clustering and its variants for K = 4: init=random

model= KMeans(init="random",n_clusters=4)
model.fit(imgs)
print("fowlkes_mallows_score :"+str(fowlkes_mallows_score(classs,model.labels_)))
print("silhouette_score :"+str(silhouette_score(imgs,model.labels_)))

fowlkes_mallows_score :0.30626322179745674
silhouette_score :0.396541331157873
```

```
In [6]: #init = kmeans++
model= KMeans(init="k-means++",n_clusters=4)
model.fit(imgs)
print("fowlkes_mallows_score :"+str(fowlkes_mallows_score(classs,model.labels_)))
print("silhouette_score :"+str(silhouette_score(imgs,model.labels_)))
```

```
fowlkes_mallows_score :0.3070067472513664
silhouette_score :0.39670360483112527
```

```
In [7]: #Bisecting K-means
model= BisectingKMeans(init="random",n_clusters=4)
model.fit(imgs)
print("fowlkes_mallows_score :"+str(fowlkes_mallows_score(classss,model.labels_)))
print("silhouette_score :"+str(silhouette_score(imgs,model.labels_)))
```

```
fowlkes_mallows_score :0.306198048094803
silhouette_score :0.3679841850451812
```

```
In [8]: #spectral clustering
model= SpectralClustering(n_clusters=4)
model.fit(imgs)
print("fowlkes_mallows_score :"+str(fowlkes_mallows_score(classss,model.labels_)))
print("silhouette_score :"+str(silhouette_score(imgs,model.labels_)))
```

```
fowlkes_mallows_score :0.3490218187997712
silhouette_score :-0.007977167481009094
```

```
In [43]: #DBSCAN
model = DBSCAN(eps=0.0169, min_samples=2).fit(imgs)
labels = model.labels_
# Number of clusters in Labels, ignoring noise if present.
n_clusters_ = len(set(labels)) - (1 if -1 in labels else 0)
print("Estimated number of clusters: %d" % n_clusters_)
print("eps_used : %.4f" % model.eps)
print("min_samples : %d " % model.min_samples)
print("fowlkes_mallows_score :"+str(fowlkes_mallows_score(classss,labels)))
print("silhouette_score :"+str(silhouette_score(imgs,labels)))
```

```
Estimated number of clusters: 4
eps_used : 0.0169
min_samples : 2
fowlkes_mallows_score :0.5058453438610407
silhouette_score :0.550628398316885
```

```
In [45]: #Agglomerative clustering
link=[]
sc=[]
fol=[]
for init in [ "single","complete", "average","ward"]:
    model = AgglomerativeClustering(n_clusters=4,linkage=init)
    predict=model.fit_predict(imgs)
    link.append(init)
    sc.append(silhouette_score(imgs,predict))
    fol.append(fowlkes_mallows_score(classss,predict))
score=pd.DataFrame({'link': link, 'silhouette': sc , 'fowlkes': fol})
score
```

Out[45]:

	link	silhouette	fowlkes
0	single	0.663786	0.513015
1	complete	0.331314	0.441801
2	average	0.618421	0.505890
3	ward	0.448057	0.337882

Based on fowlkes score best to worst : agglomerative single
,Average,DBSCAN,complete,spectral,ward,k-means++ ,random,bisecting

Based on fowlkes score best to worst : agglomerative single ,Average,DBSCAN,ward,k-
means++ ,random,bisecting,complete,spectral

In []: