1. 创建分类器create\_class\_gmm
2. 读取图像，提取图像中物体的特征，并将特征组合成特征向量，如：

FeatureVector=real([ciculatrity,area])

1. 将各样本的特征向量加入到分类器中

Add\_sample\_class\_gmm(GMMHandle,FeatureVector,0)

1. 训练分类器

Train\_class\_gmm(GMMHandle,100,0,0.01,’training’,0.001,Centers,Item)

1. 分类

5.1 提取图像中物体的特征并组成特征向量FeatureVector

5.2 clasify\_clas\_gmm(GMMHandle,FeatureVector,1,ClassID,ClassProb,Destity,KSignaProb)

6、清除分类器

Class\_class\_gmm

实例：

**\* 创建分类器**

create\_class\_gmm (2, 2, 1, 'spherical', 'normalization', 10, 42, GMMHandle)

\* **读取图像，提取图像中物体的特征，并将特征组合成特征向量，再将各样本的特征向量加入到分类器中**

for i := 1 to 4 by 1

read\_image (Image, 'color/citrus\_fruits\_'+ i$'.2d')

dev\_display (Image)

\* 'Add Samples'

get\_regions (Image, SelectedRegions)

dev\_display (SelectedRegions)

count\_obj (SelectedRegions, NumberObjects)

for j := 1 to NumberObjects by 1

select\_obj (SelectedRegions, ObjectSelected, j)

get\_features (ObjectSelected, WindowHandle, Circularity, Area, RowRegionCenter, ColumnRegionCenter)

FeaturesArea := [FeaturesArea, Area]

FeaturesCircularity := [FeaturesCircularity, Circularity]

FeatureVector := real([Circularity, Area])

if (i<=2)

add\_sample\_class\_gmm (GMMHandle, FeatureVector, 0, 0)

disp\_message (WindowHandle, 'Add to Class:'+ClassName[0], 'window', RowRegionCenter, ColumnRegionCenter-100, 'black', 'true')

else

add\_sample\_class\_gmm (GMMHandle, FeatureVector, 1, 0)

disp\_message (WindowHandle, 'Add to Class:'+ClassName[1], 'window', RowRegionCenter, ColumnRegionCenter-100, 'black', 'true')

endif

endfor

disp\_continue\_message (WindowHandle, 'black', 'true')

stop ()

endfor

dev\_clear\_window ()

\*

\* Visualize the feature space

visualize\_2D\_feature\_space (Cross, Height, Width, WindowHandle, FeaturesArea[0:5], FeaturesCircularity[0:5], 'dim gray', 18)

\* 'oranges', 40, 440

visualize\_2D\_feature\_space (Cross, Height, Width, WindowHandle, FeaturesArea[6:11], FeaturesCircularity[6:11], 'light gray', 18)

\* 'lemons', 70, 440

disp\_continue\_message (WindowHandle, 'black', 'true')

stop ()

**\*训练分类器**

train\_class\_gmm (GMMHandle, 100, 0.001, 'training', 0.0001, Centers, Iter)

**\*分类**

for i := 1 to 15 by 1

read\_image (Image, 'color/citrus\_fruits\_'+ i$'.2d')

dev\_display (Image)

\* 'Classify Image', 10, 10

get\_regions (Image, SelectedRegions)

dev\_display (SelectedRegions)

count\_obj (SelectedRegions, NumberObjects)

for j := 1 to NumberObjects by 1

select\_obj (SelectedRegions, ObjectSelected, j)

get\_features (ObjectSelected, WindowHandle, Circularity, Area, RowRegionCenter, ColumnRegionCenter)

FeaturesArea := [FeaturesArea, Area]

FeaturesCircularity := [FeaturesCircularity, Circularity]

FeatureVector := real([Circularity, Area])

classify\_class\_gmm (GMMHandle, FeatureVector, 1, ClassID, ClassProb, Density, KSigmaProb)

disp\_message (WindowHandle, 'Class: '+ClassName[ClassID], 'window', RowRegionCenter, ColumnRegionCenter-100, 'black', 'true')

disp\_message (WindowHandle, 'KSigmaProb: '+KSigmaProb, 'window', RowRegionCenter+30, ColumnRegionCenter-100, 'black', 'true')

endfor

if (i#15)

disp\_continue\_message (WindowHandle, 'black', 'true')

endif

stop ()

endfor\*

**\* 清除分类器**

clear\_class\_gmm (GMMHandle)