Department of Computer Science

California State University, Channel Islands

COMP-546: Pattern Recognition

Lab Report

Lesson28 phys546 Classify HW 2

Student Name: Sandipta Subir Khare
Student Major: Masters in Computer Science

Write an m-file to automatically detect and classfy the fruit in the image fruit.tif.

Trace the boundaries of the objects and display the original image with a one-pixel-thick boundary overlaid on each object. The colors of the boundaries are important: Red indicates apple, green indicates grapefruit, and yellow indicates banana. Classification may be done using any method of your choosing, including region size, elongation, convexity, etc. (Note: The apple boundaries will not come out good with a single threshold. One solution is to use a double threshold).

Hints: Use grayscale open to get rid of small artifacts. Smooth (median) to improve apples. Threshold and remove edge objects. Label. Measure circularities and threshold on circularity to get bananas. (Get outline and overlay on original image).Take out the bananas, label the remaining fruit, and threshold on size to get grapefruit (oranges); overlay on original image (different colors).

Answer

20 answers

Methodology

- Firstly, we will access pre-acquired image datasets containing various images of fruits.
- Since the training is a time-expensive process, the possibility of GPU computing in a standalone or in cloud compute engine would be used (tentative).
- Then we would train our system with images using ALEXNET CNN architecture.
- The training would be carried out with around 450 images in each class chosen randomly.
- The inference system would then use the remaining images in the dataset to validate the classification process.

We will have 3 types of classes

- apple : red
- grapefruit : green
- banana: yellow

Code

%%Generating Labelled Structured Data%% (Main.m)

```
fruit(1).images = dir(fullfile('path to your complete img folder in which apple images are there'))
fruit(1).name = 'apple'

fruit(2).images = dir(fullfile('path to your complete img folder in which grapefruit images are there'))
fruit(2).name = 'grapefruit'
```

fruit(3).images = dir(fullfile('path to your complete img folder in which banana images are there'))
fruit(3).name = 'banana'

So, for me path is 'C:\Users\Sona\Desktop\image augumentation\apple','*.png')). So, accordingly specify your path

x1 = size(fruit(1).images,1)
x2 = size(fruit(2).images,1)
x3 = size(fruit(3).images,1)

images_train_apple = datasample(fruit(1).images,0.6*(x1),'Replace',false)
images_valid_apple= datasample(fruit(1).images,0.2*(x1),'Replace',false)
images_test_apple = datasample(fruit(1).images,0.2*(x1),'Replace',false)

Do the above for 2 more fruits (banana and grapefruit)

// **for apple** (%%Image Augmentation%%)

fruit_apple=cell(1,length(fruit(1).images))

for k1=1:length(fruit(1).images)

i=[fruit(1).images(k1).name]

  fruit_apple{k1}=imread(i);

end

// **for grapefruit** (%%Image Augmentation%%)

fruit_grapefruit=cell(1,length(fruit(2).images))

for k2=1:length(fruit(2).images)

i=[fruit(2).images(k2).name]

  fruit_grapefruit{k2}=imread(i);

end

// **for banana** (%%Image Augmentation%%)

fruit_banana=cell(1,length(fruit(3).images))

for k3=1:length(fruit(3).images)

```
i=[fruit(3).images(k3).name]

  fruit_banana{k3}=imread(i);

end
```

%%Neural Network

```
fritz = imageDatastore('C:\Users\Sona\Desktop\image
augumentation\','IncludeSubfolders',true,'LabelSource','foldernames');
[image_train,image_test,image_validate] = splitEachLabel(fritz,20,20,20,'randomized')
anet = alexnet
layers = anet.Layers
frlayer = fullyConnectedLayer(5)
layers(end-2) = frlayer
layers(end) = classificationLayer
options = trainingOptions('sgdm','InitialLearnRate',0.001)
[fruitnet , info] = trainNetwork(image_train, layers, options)
```

Make Validation.m file and write below code to test the images.

```
testImage1 = imread('c.jpg');
testImage = imresize(testImage1,[227 227]);
[label,score] = classify(fruitnet,testImage);
label = string(label);
[score, idx] = max(score);
bbox = [0 0 2 2];
annotation = sprintf('%s: (Confidence = %f)', label, score);
outputImage = insertObjectAnnotation(testImage,'rectangle',bbox,annotation);
figure;
imshow(outputImage)
```