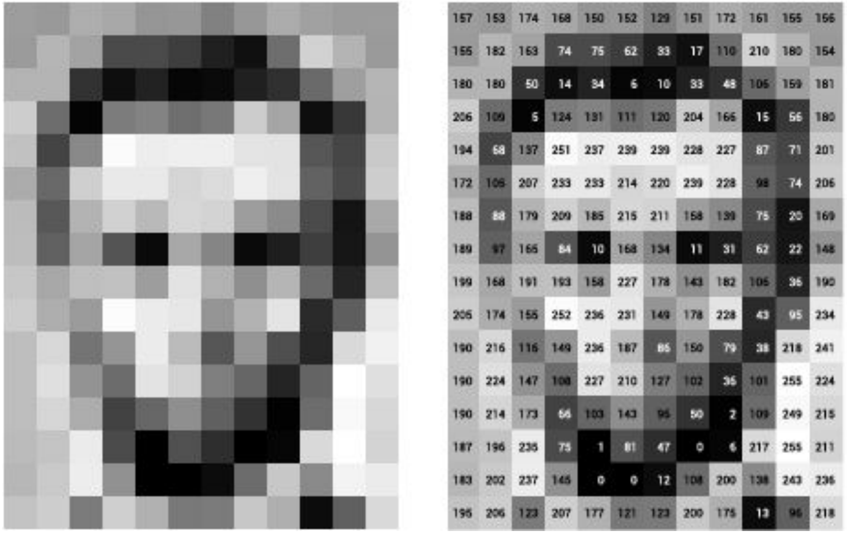
Activity 12
Feature Extraction

Marc Jerrone Castro

Features are pieces of information which represent particular properties, both visual and inherent, of a particular data point either as a boolean, a number, or as a vector.



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Figure 1.0 Converting an image to pixel intensity values which can later be used for feature extraction.

Feature extraction deals with the parsing of these information through various means such as histogram equalization and analysis, blob analysis, and color space analysis. In this experiment, we aim to extract relevant features from a number of fruit images collected from Google Images and qualitatively assess its relevance for differentiating these fruits into classes.

Features

For this specific study, we will be extracting three specific features from each of our inputs.

- Mean RGB Value
  - Average pixel value of all three RGB channels
- Convexity
  - Quantitative value for determining convexity of a particular object. It scores the image from 0 (concave shape) to 1 (convex shape).
- Inertia Ratio
  - A measure for how elongated an object is with respect to the x and y axis length.

Features

Mean RGB Value

The reasoning for pursuing the mean RGB values of an image as a feature for this particular study is because of the distinct color difference between our three classes: apples, oranges, and bananas.

I performed this particular feature extraction using the commands,

```
fruit_rgb_mean = np.mean(fruit_c/256)
```

Which simply takes the mean value of each class and divides it with 256 such that we limit the value for this particular feature from zero to one.

Convexity

The reasoning for pursuing the convexity of the input data is due to the distinct difference between the shapes of each class. Although at some cases, the difference of convexity of the oranges and apples is not distinct. The inherent concavity of the banana would pose this feature as particularly of interest.

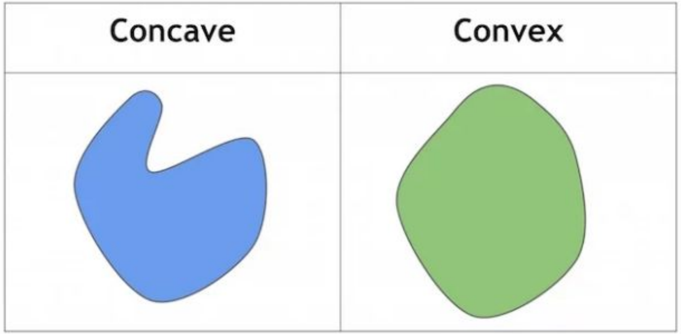


Figure 2.0 Convex shape vs a concave shape

We can determine a score for convexity by simply determining the ratio of the area that the object covers and a superimposed convex shape which encloses all relevant pixels in the image.

IMAGE SOURCE: https://www.learnopencv.com/blob-detection-using-opencv-python-c/

Inertia Ratio

The reasoning for pursuing the inertia ratio of the image is primarily my interest in the particular difference between a circular orange in contrast to an elongated banana. Although the difference between the inertia of an orange and an apple is not that obvious, some particular samples show that apples are particularly more elongated to oranges - which makes the inertia ratio ideal for differentiating these fruits.

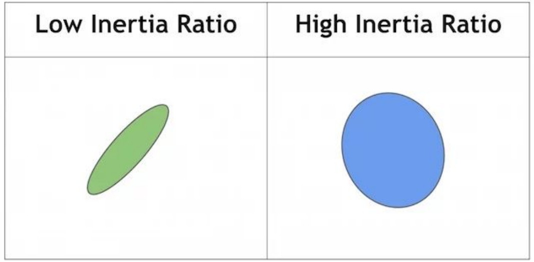


Figure 3.0 Comparison of high and low inertia ratios

One calculates for this ratio by determining the minimum and maximum inertia tensor eigenvalues and calculating the ratio between these two. In simpler terms, it determines a ratio between the longest distance for each axis.

Data Used

The data used for this study is less than ideal, as the general image dimensions are not uniform - however pre-processing steps such as cropping via contour detection and use of zero-dimensionality features limit these factors in our results. The images used were collected from various sources listed in Google Images. A particular characteristic of each image that I have pre-selected is that it should be an image imposed on a white background as it is generally easier to parse out the white background in contrast to noisy backgrounds. As I have no control over the general resolution of the images, I had opted to assume this as a plausible source of error.



Pre-processing through cropping

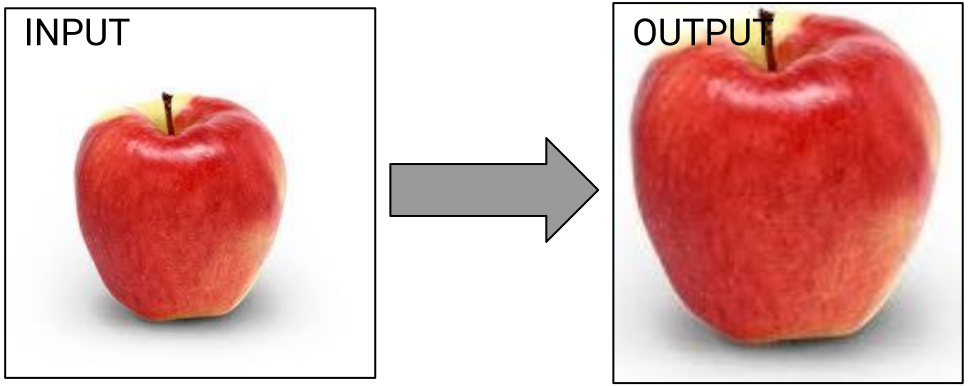


Figure 4.0 Comparison of input images after cropping via contour detection.

As for my pre-processing step, I had performed cropping via contour detection before any analysis as to limit the amount of white spaces found within the background. This would allow for the mean RGB value of the image to have minimal dependency on the white pixels found within the image.

```
th, threshed = cv2.threshold(gray, 240, 255, cv2.THRESH_BINARY_INV)
kernel = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (10,10))
morphed = cv2.morphologyEx(threshed, cv2.MORPH_CLOSE, kernel)
cnts = cv2.findContours(morphed, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)[-2]
cnt = sorted(cnts, key=cv2.contourArea)[-1]
x,y,w,h = cv2.boundingRect(cnt)
dst = fruit[y:y+h, x:x+w]
```

Thresholding & Blob Detection

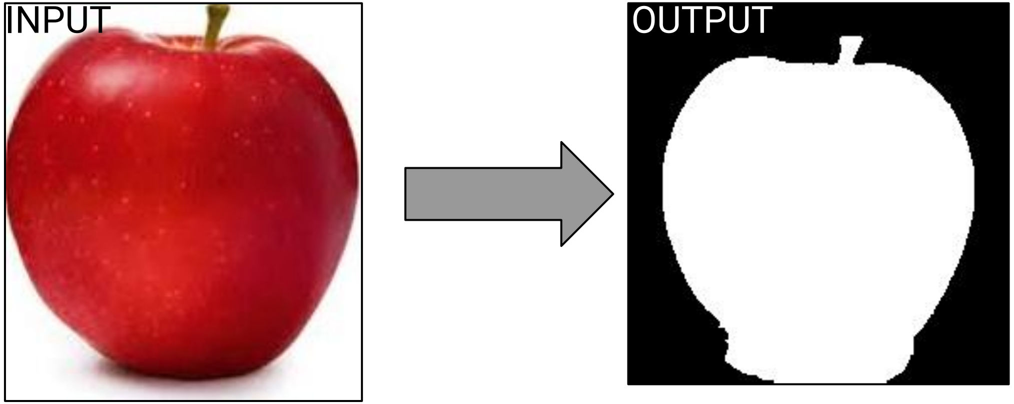


Figure 5.0 Thresholding of image using binary conversion. One must take note that due to time constraints and in attempt to simplify the study - shadows and stalks of the fruit, as well as, leaves of the fruits remained in the binary image.

After cropping the image, I performed blob detection on a binary-transformed version of the image. This was done in reference to Activity 10 and 11 where we extracted relevant information from a "blob" of ROI and qualitatively assess its properties. Extracting features of interests - area, convex area, perimeter, and inertia eigenvalues.

```
area = reg[0].area
conv_area = reg[0].convex_area
perimeter = reg[0].perimeter
inertia_a = reg[0].inertia_tensor_eigvals[0]
inertia_b = reg[0].inertia_tensor_eigvals[1]
```

Features Extracted

	file name	fruit type	dominant_color	convexity	circularity	inertia ratio	hsvmean	rgbmean
0	01	red	0.0	0.925760	0.512439	0.664910	105.550144	0.515804
1	02	red	0.0	0.999024	0.892545	0.885877	15.526212	0.595774
2	03	red	0.0	0.965941	0.633915	0.726095	49.530879	0.529208
3	08	red	0.0	0.991004	0.878732	0.827307	71.614742	0.497882
4	09	red	0.0	0.962301	0.749591	0.826559	99.081343	0.489888
5	15	red	0.0	0.985413	0.826993	0.907376	8.920236	0.627684
6	16	red	1.0	0.976841	0.789151	0.965360	14.547421	0.616134
7	19	red	0.0	0.907978	0.629531	0.588133	64.443734	0.585084
8	22	red	0.0	0.854527	0.409290	0.550018	81.045480	0.603666
9	25	red	0.0	0.935736	0.729294	0.725295	29.821999	0.617304
10	26	red	0.0	0.977636	0.821655	0.964853	52.485871	0.498262

After using the values extracted from the images, the following features were determined.

Although some features mentioned earlier, features such as - dominant color and hsv mean value were also determined as they were plausible sources for differentiating the fruits.

However, as there is significant overlap between red, yellow, and orange hue values in the 360 degree wheel. The use of a mean hsv values was disregarded.

Features Extracted

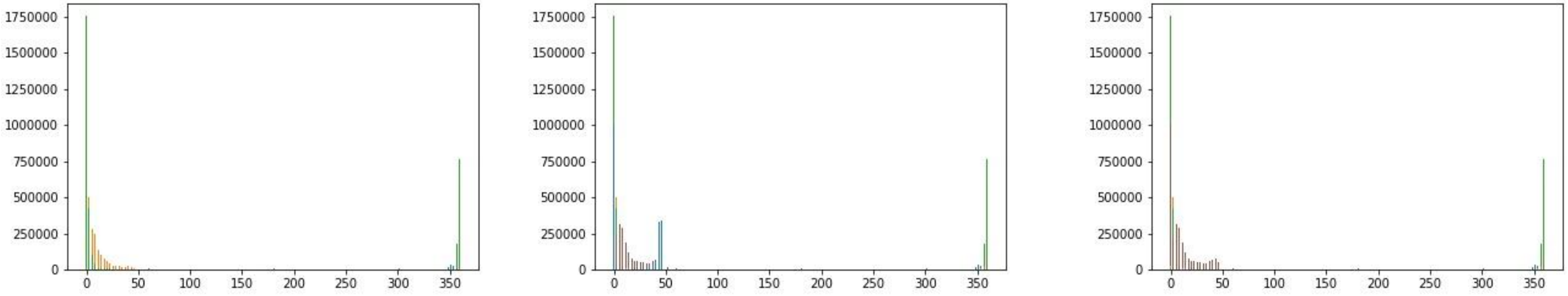


Figure 6.0 Color histograms under the HSV color space of the images. Shown above is a sample for each class of fruits where the histogram from left is for an apple, the middle is a histogram for a banana, and the histogram on the far right is that of an orange.

Another feature of interest was dominant color found in each image under the HSV color space. I thought it was a promising approach for determining the color space of each sample however - as one may observe - there is a certain bias to values less than 50 as the bins for HSV color space overlap at red, orange, and yellow spaces. Thus this specific feature was disregarded and deemed unlikely to contribute to differentiating the fruits.

Features Extracted

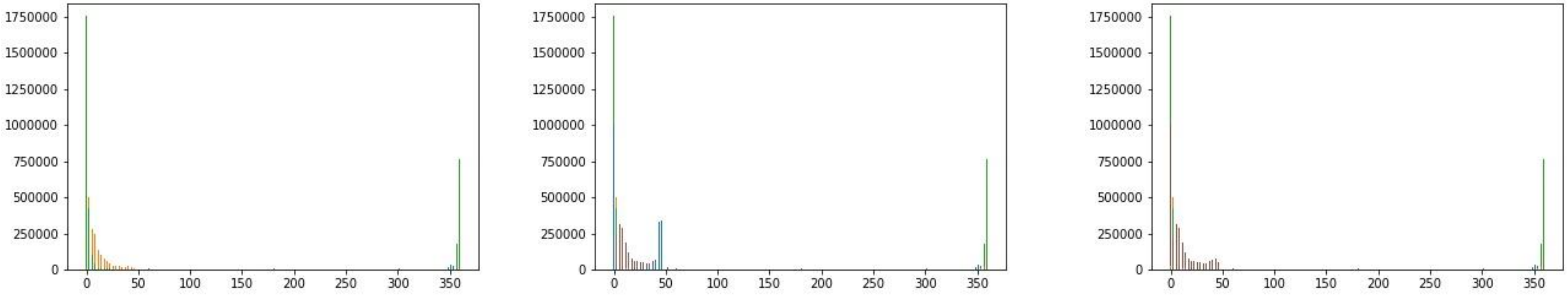
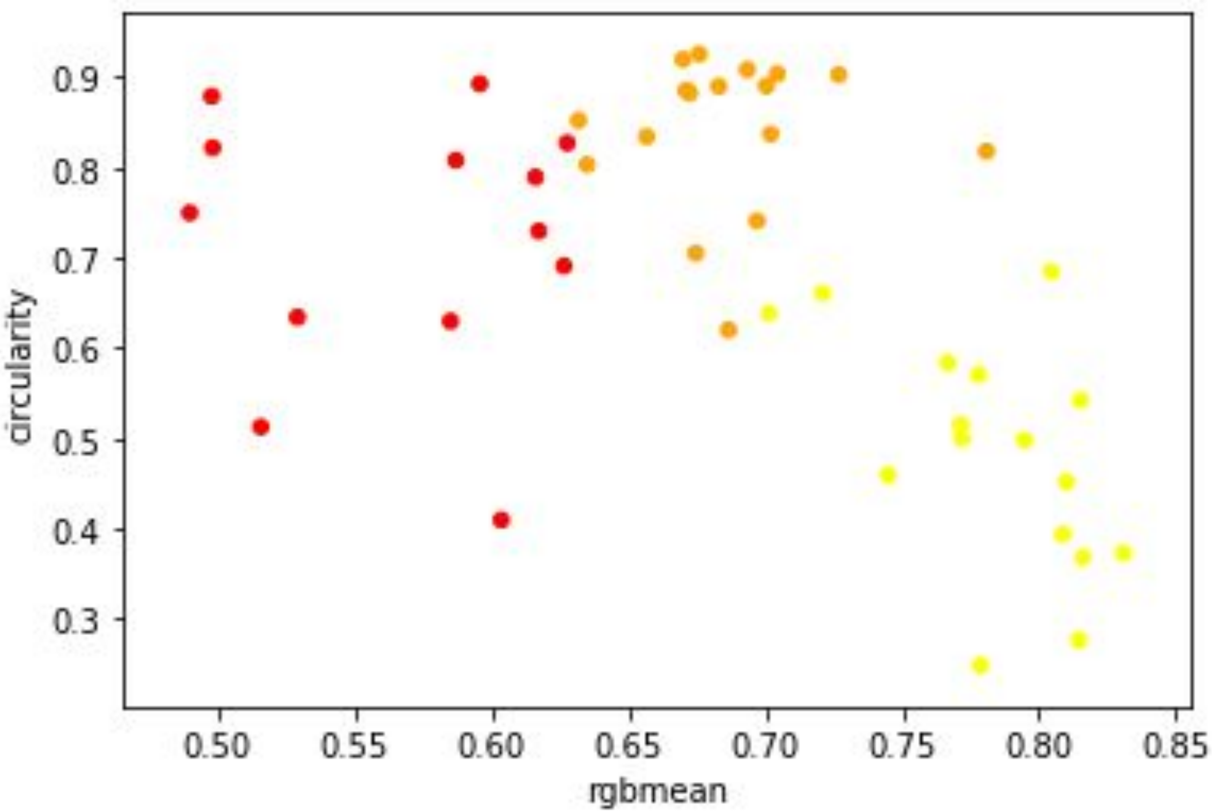


Figure 6.0 Color histograms under the HSV color space of the images. Shown above is a sample for each class of fruits where the histogram from left is for an apple, the middle is a histogram for a banana, and the histogram on the far right is that of an orange.

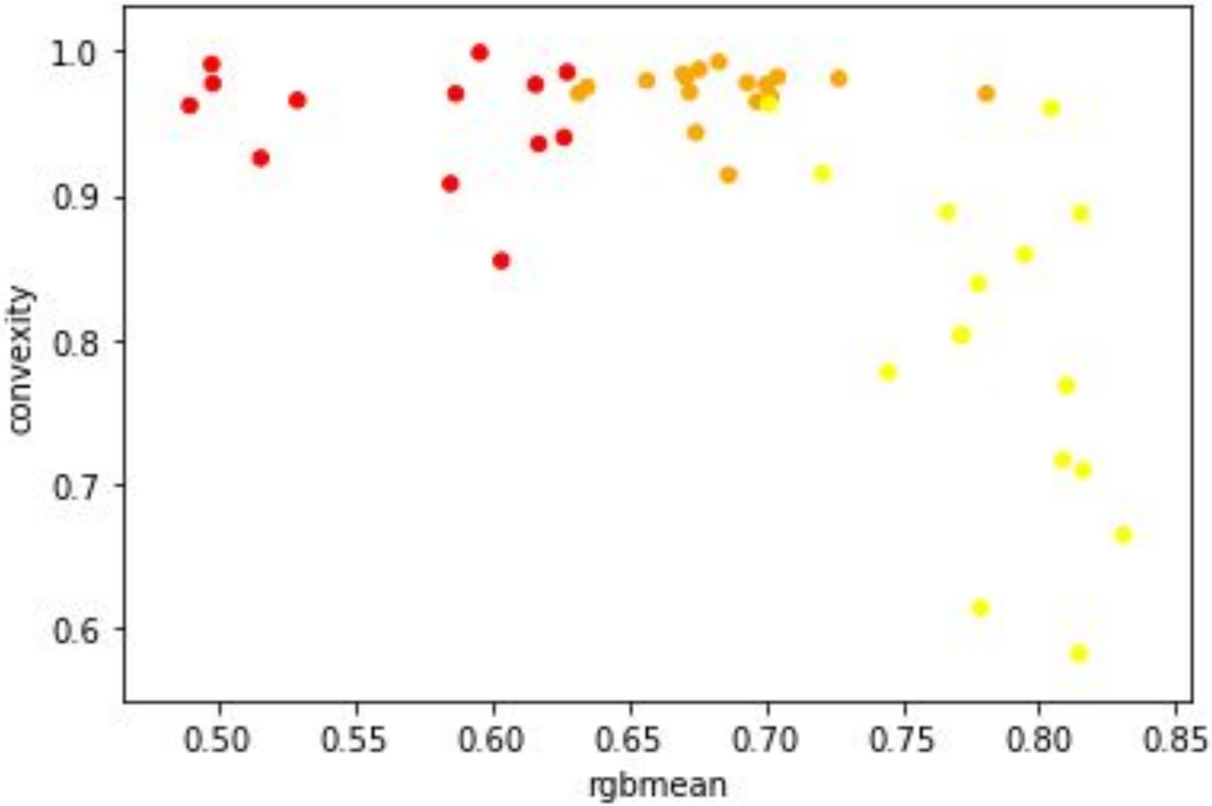
Another feature of interest was dominant color found in each image under the HSV color space. I thought it was a promising approach for determining the color space of each sample however - as one may observe - there is a certain bias to values less than 50 as the bins for HSV color space overlap at red, orange, and yellow spaces. Thus this specific feature was disregarded and deemed unlikely to contribute to differentiating the fruits.

RGB Mean Value vs Circularity



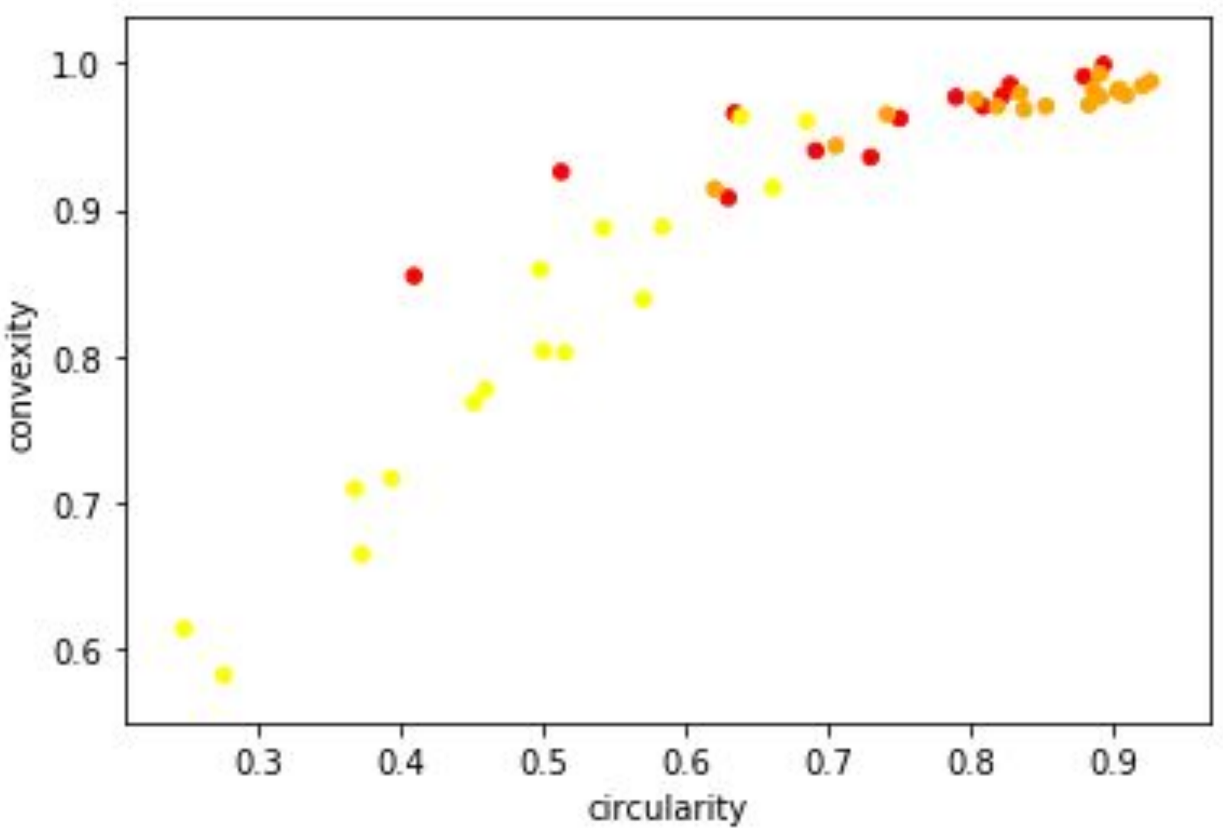
Upon plotting the features, we observed that using RGB mean values and circularity score highly differentiate apples from bananas. However, a minor overlap is observed for oranges and apples, as well as bananas and oranges. As for the circularity of the fruits, a vague difference between bananas and oranges can be observed - although as apples circularity against orange circularity has a vague difference.

RGB Mean Value vs Convexity



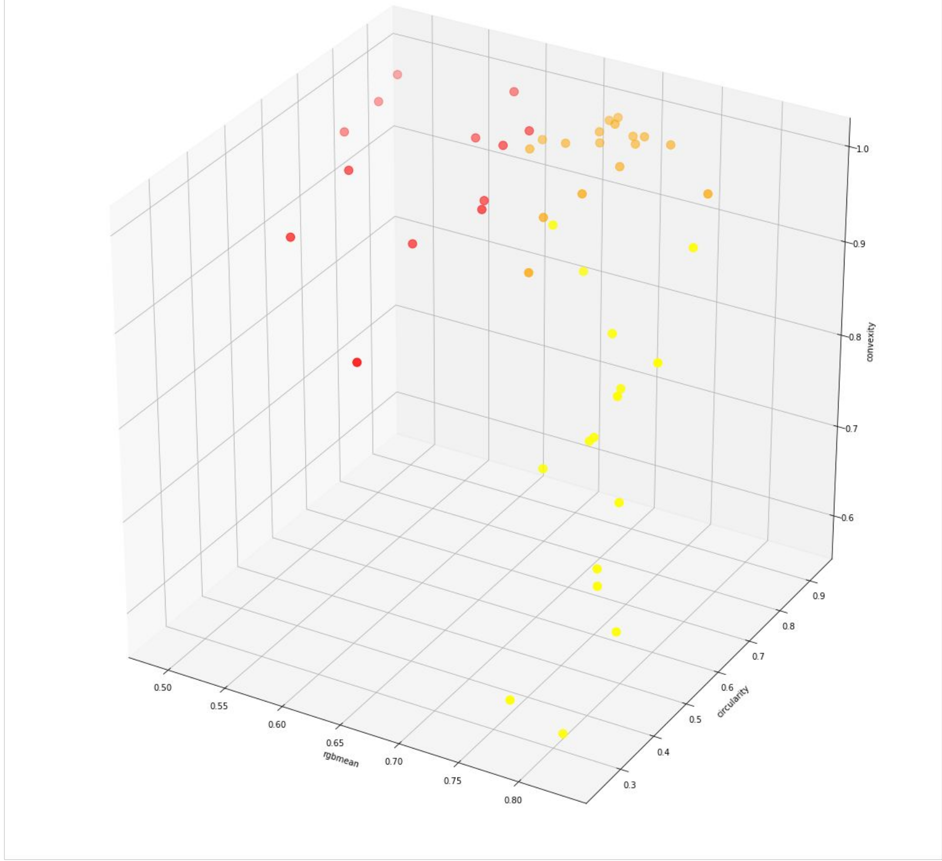
As for the convexity of the fruits, apples and oranges had no observed difference for this specific feature. However, aside from some stray samples, bananas were distinctly observed to be inclined towards a more concave shape.

Circularity vs Convexity



As for circularity and convexity of the fruits. I observed that bananas were highly distinguishable within this specific feature space. However, as the circularity and convexity of apples and oranges are highly similar - these two features are not suited for separating them.

3D Representation of the Image Features



Representing the features in 3D space shows that it is highly possible to differentiate the fruits using a combination of these three features. Using machine learning models - such as perceptrons, linear regression, and convolutional networks. We may be able to determine a model for separating these images.

This classification will be further explored in the Activity 13 were we apply a perceptron to the data we uncovered using Feature Extraction.

SCORE

QOP: 5/5

TC: 5/5

Initiative: 2/2