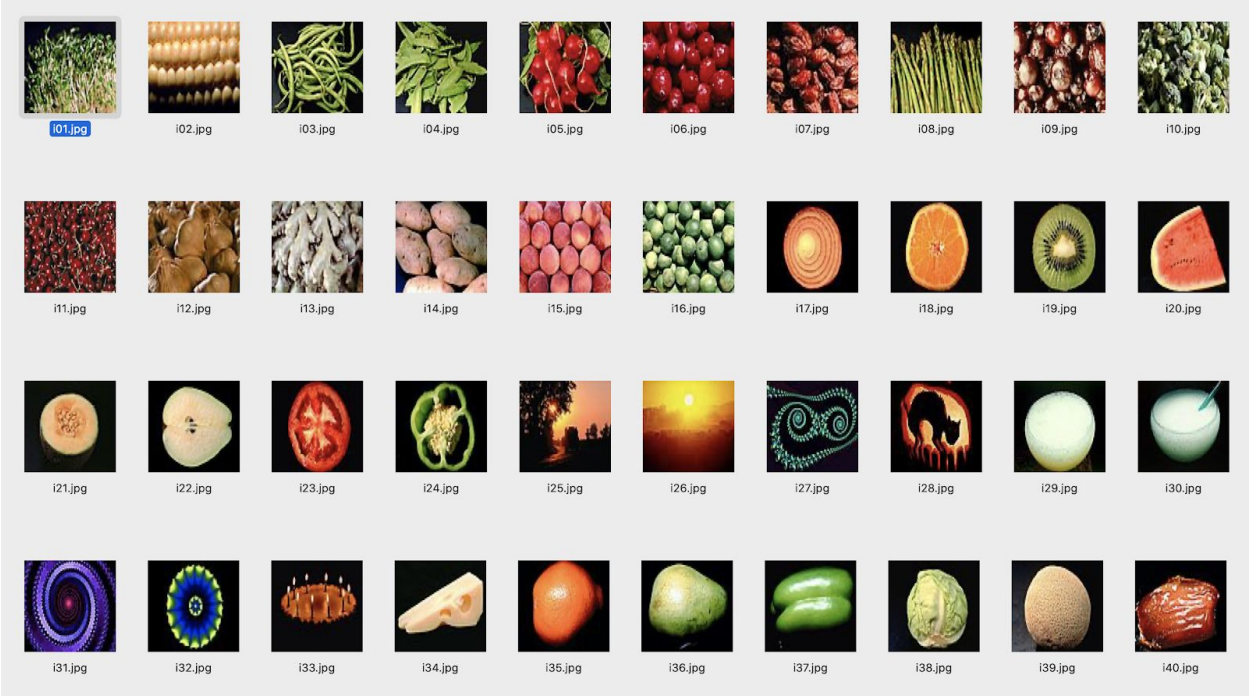
Image Retrieval Software

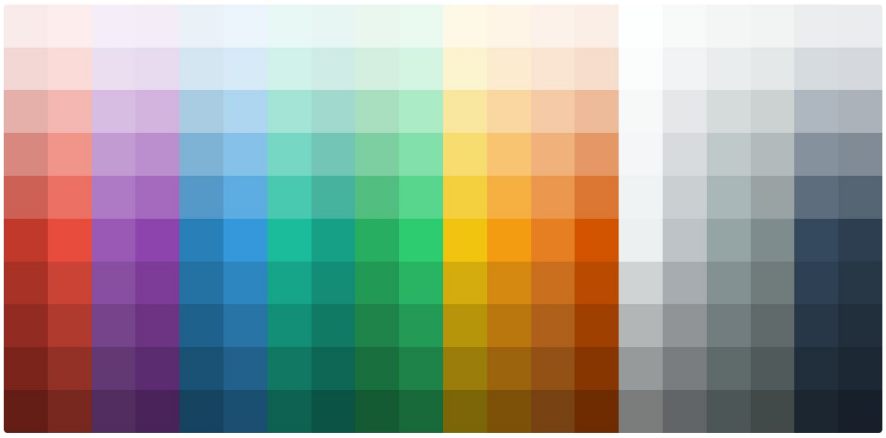
Author: Amee Assad

Using the following 40 images, I attempt to match images based on an analysis of their color, texture, and objects. Using crowdsourced data of image similarity, I then score my program's ability to match images, as judged by humans.

The code is written in Python and uses methods from opencv (for image processing), numpy (to compare different images), and imutils (for displaying the images).



I Color Distance



Computation

I used opencv's calcHist() function in order to get 3-dimensional histograms for each of the images. I decided to use 12 bins for the red channel, 12 bins for the green channel, and 6 bins for the blue channel. I used less bins for blue than the rest of the colors because not many blues were generally used in the set. More specifically, whenever a blue was used, it tended to stand out, making the user identify any image with a noticeable amount of blue as similar to other blue images. Red and green channels have the same amount of bins, as they were the two main colors used in the images and tended to differentiate one image from another if the colors were mainly green or mainly red. 12 was the "sweet spot" of number of bins because any more and unnoticeably different colors would be marked as different, and any less and noticeably different colors would be marked as the same.

To calculate the L1 distance between histograms, I used the function

$$L1(Image1, Image2) = \sum_{(r,g,b)} |Image1(r,g,b) - Image2(r,g,b)| \,/\, (2 * rows * cols)$$

Visualization and performance evaluation

1. System versus crowd preferences, using scores

TOTAL SCORE: 5794

Each target score is shown in white over the respective target image, and each row score is shown in the first column in green over the query image. The final column of images display the farthest match from the query image (and the score is shown in white but is not added to the row score).

2. System versus personal preferences, using set intersection

The number of matches with my personal preferences for each image totals:
1 +2 +2 +1 +3 +2 +2 +1 +3 +2 +2 +2 +1 +2 +1 +1 +1 +1 +1 +2 +1 +1 +1 +1 +1 +0 +0 +2 +1 +1 +1 +1 +1 +2 +1 +1 +1 +1 +0 +0 = 51 (out of max 120)

3. Discussion

The crowd-based accuracy and user satisfaction are OK when only matching colors, perhaps because RGB colors are closely connected to HSV, which is what humans see and make connections with. The highest score were the cherries in image 6, which were the most red photo, followed by another image of cherries with green leaves (#5). Image 5 also matched all of my personal preferences. On the opposite end, the image of a scenic sunset got 1 point because it was matched with dominantly red photos of cherries and dates. Four images did not match with any of my preferences: the date (#40), the melon (#39), the cave (#27), and the blue abstract image (#26), because I didn't prioritize color but meanings for those images.

II Texture Distance

Computation

In order to measure the texture roughness of the images, I first made them all grayscale in order to negate color differences, and got the laplacian of each image by taking the sum of the bordering pixel values of each non-edge pixel multiplied by 8. I then used opencv's calcHist() function to create 1D histograms (with 510 bins) for the grayscale channels of all images. The resulting distances were done in the same way as the color L1 distances.

Visualization and performance evaluation

1. System versus crowd preferences, using scores

TOTAL SCORE: 4886

2. System versus personal preferences, using set intersection

2 +0 +1 +2 +3 +2 +1 +2 +1 +2 +3 +0 +1 +1 +0 +1 +1 +1 +1 +0 +1 +1 +1 +1 +1 +0 +1 +0 +1 +2 +0 +1 +0 +1 +0 +0 +1 +1 +0 +0 = 38

3. Discussion

- Highest scores: image #8 (scored 336pts), image #1 (scored 300pts)
- Lowest scores: image #28 (scored 7pts), image #35 (scored 4pts)
- Perfect match of personal preferences: image #5, image #11
- Perfect mismatch of personal preferences: image #2, image #12, image #15, image #20, image #26, image #28, image #31, image #33,image #35, image #36, image #39, image #40

The scored highest based on crowd preferences were the most textured images (the ones that had many of the same object rather than one instance of the object). On the opposite end, the images with just one centred object without much texture didn't do so well in comparison.

In regard to personal preferences, only two images got perfectly matched to my preferences -- and both were images of many cherries, which is interesting because the colors also match, which means that I prioritize color a lot more than texture. Many images didn't even get one of my personal preferences correct, including the corn, the watermelon, the scenic sunset, the cave image, the blue swirl, the cake, the orange, the pear, the melon, and the date.

III Shape Distance

Calculation

In order to determine what is "black", I used a color picker tool to find out the RGB values of various of the images' backgrounds of what I thought was "black" in my opinion. This allowed me to quantify my concept of black. Some RGB values found by the color-picker tool include (8,7,10), (0,0,12), (8,8,8), (19,17,16) - asparagus, (19,28,3) - broccoli, and (24,15,18) - cherries. I decided to round up for each color channel, and quantified as black as any RGB value less than (30,30,30). If any one color channel went above 30, then the program would not see that pixel as black. Doing this allowed me to take the binary of each image, which I then used to find the normalized overlap between every pair of images.

Visualization and performance evaluation

1. System versus crowd preferences, using scores

TOTAL SCORE: 2909

2. System versus personal preferences, using set intersection

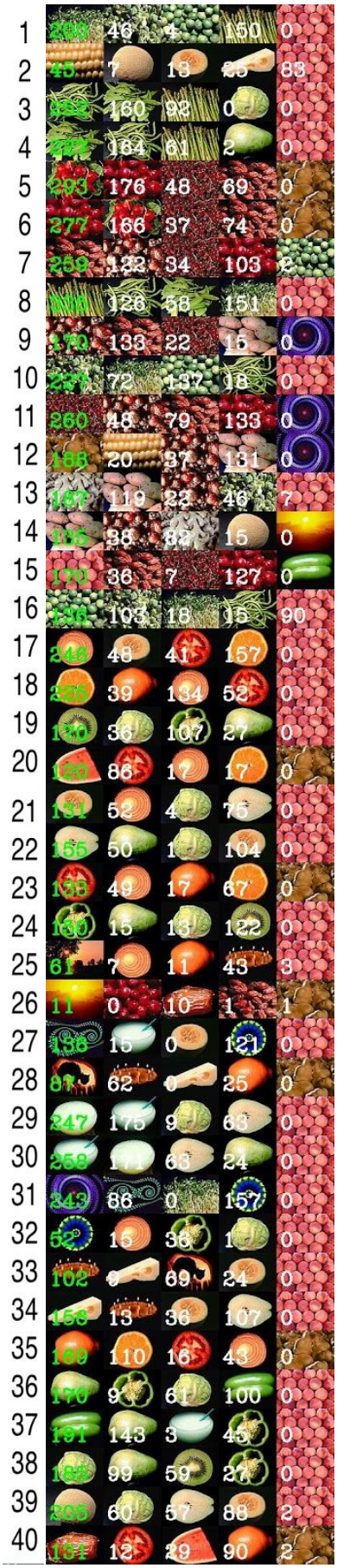
0 +1 +1 +0 +0 +0 +0 +1 +0 +0 +0 +0 +1 +1 +0 +1 +1 +1 +0 +1 +1 +0 +0 +0 +0 + 0 +0 +0 +1 +1 +0 +0 +0 +0 +1 +1 +0 +0 +1 +0 = 15

3. Discussion

- Highest scores: pear - #36 (scored 205pts), onion - #17 (scored 204pts)
- Lowest scores: plants - #1 (scored 1pt), sunset - #26 (scored 1pt)
- Perfect match of personal preferences: *none*
- Perfect mismatch of personal preferences: *25 images*

The purely shape-based image retrieval was not accurate when scored by both the crowd and my own preferences. In fact, it did not get any perfect matches with my personal preferences, and for 25 out of 40 of the images it couldn't even get one of my preferences correct. However, it did OK with images of an individual object centred in the middle, but anything else more complicated and the system did a poor job at matching the images. The lowest scores were images without a single centred object.

IV Overall Distance



Computation

In order to find a good simplex vector, I tried to prioritize either shape, color, or texture. Since color got the highest score by a lot, I decided it was the most important factor when determining whether images matched one another.

For color distances of less than 0.3, the we should pay attention to the other image factors: 20% shape and 10% texture.
If the color distance and texture distance are less than 0.6, then the color distance isn't as important, and drops from 70% of the total distance to 40%. The other 60% are made up of 10% texture distance and 50% overlap.
Otherwise, when the color and texture distances are both above 0.6, the program will only pay attention to color.

```
#CROWD:
if color_dist < 0.3:
  tot = 0.7*color_dist +  0.2 * overlap + 0.1*text_dist
elif color_dist < 0.6 and text_dist < 0.6:
  tot = 0.4 * color_dist + 0.1 * text_dist + 0.5 * overlap
else:
  tot = color_dist
```

Visualization and performance evaluation

1. System versus crowd preferences, using scores

TOTAL SCORE: 7095

2. System versus personal preferences, using set intersection

2 +2 +2 +1 +3 +2 +2 +2 +2 +2 +2 +2 +1 +2 +1 +1 +2 +2 +1 +1 +2 +1 +0 +1 +1 +0 +1 +1 +2 +3 +1 +1 +1 +1 +2 +0 +1 +1 +2 +1 = 58

3. Discussion

- Highest score: image #8 (scored 335pts)
- Lowest score: image #26 - sunset (scored 11pts)
- Perfect match of personal preferences: *none*
- Perfect mismatch of personal preferences: img #36, img #26, img #23

The system scored highest against the crowd for images with similar colors and shapes, and with lots of texture filling up the entire image. The lowest scores were those that weren't images of fruits (their similarity is more abstract). While the system overall did well, it did not match perfectly with my personal preferences on any one image. It did completely mismatch with my preferences on the pear, the sunset, and the tomato.

V Crowd Versus User

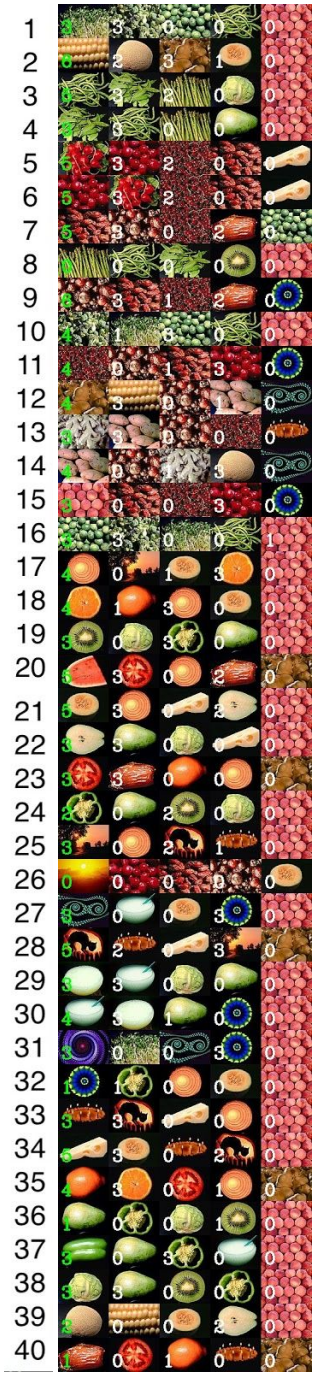
Crowd-based performance

a. The actual upper bound on the very best grand total score that is possible with these images, given all the differences of opinion in Crowd.txt, is **9853.**

b. The final system grand total score came **72%** close to this actual upper bound.

c. The personal preference grand intersection value came **48.3%** to the maximum of 120.

User-based performance



After replacing the crowd preferences matrix with my own personal preferences matrix, I was able to improve the system slightly. For my own preferences, I increase the importance of color but never only rely on color. The new weighting vector is:

```
#AMEE:
if color_dist < 0.3:
  tot = 0.7*color_dist +  0.2 * overlap + 0.1*text_dist
elif color_dist < 0.6 and text_dist < 0.6:
  tot = 0.7 * color_dist + 0.15 * text_dist + 0.25 * overlap
else:
  tot = 0.8*color_dist +  0.2 * overlap
```

1. System versus crowd preferences, using scores

TOTAL SCORE: 136

2. System versus personal preferences, using set intersection

1 +3 +2 +1 +3 +2 +2 +1 +3 +2 +2 +2 +1 +2 +1 +1 +2 +2 +1 +2 +2 +1 +1 +1 +2 +0 +1 +2 +1 +2 +1 +1 +1 +2 +2 +1 +1 +1 +1 +1 = 61

3. Discussion

- Highest score: image #2, image #9
- Lowest score: image #8, image #26
- Perfect match of personal preferences: img #2, img #5, img #9
- Perfect mismatch of personal preferences: img #26

My personal preference grand intersection value now comes to 50.8% of the maximum score of 120. I was unable to get it higher than this because when it comes to matching the preferences of a single person, while it is easier to figure out his or her thought process, there are still some choices that are unpredictable. In terms of myself, I know that I selected some images in a more abstract manner than in a way that can be mapped out logically. Especially for images that did not fall under the fruit grouping, the choices were a lot more unpredictable - more so than a software can understand.