**Machine Learning Hamburger Ingredients Project**

In this project, we have a stationary camera that continuously captures photos of hamburger ingredients as they pass through a factory conveyor belt. These photos are subsequently divided into 10 equal parts and labeled using software.

Names Of Ingredients:

BBQ Souce: Chicken: Big Nugget: Sausage: Meatball:

çerez, atıştırmalık, kabuklu yemiş ve çekirdek, yemek, gıda, kabuklu yemiş içeren bir resim

Açıklama otomatik olarak oluşturuldu

Cheese: Nugget: Ham. Patties: Meat: Mushroom:

yemek, gıda, çerez, atıştırmalık içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Creating Dataset**

Given the fixed camera position and the consistent factory environment, we took the initiative to create our own dataset. Since the background remains unchanged and the photos are consistently taken from a top-down perspective, we focused solely on capturing the RGB values of the ingredients rather than including the RGB values of the conveyor belt.

Dataset Images:

sarı içeren bir resim

Açıklama otomatik olarak oluşturuldu

To create the dataset, we calculated the sum of RGB values for each pixel within an ingredient and divided it by the total number of pixels in that ingredient. This process yielded the average RGB value for each individual ingredient. We compiled these average RGB values into a list and then determined the total number of samples collected for each ingredient, which we divided by the number of ingredients. This calculation provided us with the average value across all the images.

Dataset:

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Creating Dataset Code Explanation:**

**metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Imported PIL and numpy.

Clears dataset (mean\_list.txt) at opening

Min\_rate and max\_rate will be used later for deleting background color

İmg\_path\_list is our pictures location. We can add or remove pictures.

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

This function takes inside a piece of photo (1/10).

Then takes RGB values of each pixel of that piece.

If R or G or B values = 0, it passes.

If RGB values are on grayscale with %10 miss rate, it passes.

Else RGB values add to a list and the function returns that list.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Here we take all img paths from img\_path\_list

Then we splitted the img to 10 pieces and each piece get (width/5 height/2) size

Then this pieces add to a new variable.

We used get\_all\_rgb function here and from that list we add R values to R\_list G values to G\_list B values to B\_list.

Took their mean and append them to the\_list.

**Now the code is being hard to explain, our brains are shutted down here:**

**metin, ekran görüntüsü, yazı tipi, tipografi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Created these values.

(r0,g0,b0) = “BBQ souce /// (r1,g1,b1) = “Chicken” … (r9,g9,b9) = “Mushroom”

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Then we add RGB values to r0,g0,b0 etc. from the\_list.

metin, ekran görüntüsü, yazı tipi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu metin, yazı tipi, ekran görüntüsü, tipografi içeren bir resim

Açıklama otomatik olarak oluşturuldu ekran görüntüsü, yazı tipi, metin, tipografi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Then all variables are divided by length of img\_path\_list. The reason of divide is we took mean of all pictures here. We could use for loop here but we just want to see this without for loop and didn’t change it.

metin, yazı tipi, yazılım, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

At least we write all these values to txt file.

**Ingredient Detector**

In this part of the project, we compared the images from the camera with the dataset we created. Based on the comparison results, we wrote code that attempts to predict the name of the ingredient. We believe it would be more effective to explain this part of the project through the code. So, let's take a look at the code without further ado.

**Ingredient Detector Code Explanation**

**metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

We talked about this code on dataset code explanation. But in here we see just an image path, not a list. Because we took camera’s view here. TextList is the list of ingredients’ names.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

In here we took data from txt file and append that data to mean\_list

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

The same function that we explained in dataset code explanation.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Takes image’s width and height then gives pieces’ width and height values.

Then creates empty lists. For normal pieces and bordered-subtitled pieces.

From line 65 to 180 there is a for loop. Let’s look at it closer.

metin, ekran görüntüsü, yazı tipi, yazılım içeren bir resim

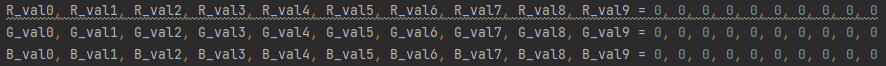
Açıklama otomatik olarak oluşturuldu

x,y are pieces’ position on big image (5x2). Then crops piece to it’s place. Current\_data is one of ten piece’s data. Data\_diff\_ …. lists are takes difference between current picture’s each pixel (without background) and dataset’s RGB values. This algorithm is using ML algorithm here.

ekran görüntüsü, çizgi, diyagram, öykü gelişim çizgisi; kumpas; grafiğini çıkarma içeren bir resim

Açıklama otomatik olarak oluşturuldu

It tooks all red,green,blue difference from dataset with abs() for absolute result. Each line on txt file represents a ingredient. For example line1: BBQ souce… line10:Mushroom. After taking difference with all lines on data we will look for the values.



Again 10 RGB values for comparising 10 ingredients.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu metin, ekran görüntüsü, yazı tipi, menü içeren bir resim

Açıklama otomatik olarak oluşturuldu

From data\_diff\_list we took RGB difference values and add them to these variables. These variables are sums of abs(piece pixel RGB-data RGB).

For example

Piece1 RGB:

pixel1:10,20,30

pixel2: 30,20,50

……

Pixel2650 : 120,115,150

Dataset line1 = 115/83/63

R\_val0 gets all pixels difference’s sum with line1

Dataset line2 = 177/139/92

R\_val1 gets all pixels difference’s sum with line2

….

ekran görüntüsü, metin, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Then we divide these values with length of data\_diff\_list. This is probably different from eachother. Because a piece’s ingredient is might 30% of the piece, 70% is the background. Or maybe 60% ingredient and 40% background. To make a better desicion, all values must be divided by their piece’s pixel value.

metin, yazı tipi, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

This is for output that we are going to explain in presentation.

metin, yazı tipi, ekran görüntüsü, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Then we took the least difference value from there and we took index of it, Index is for getting the ingredient name from the list that we created at start of project. And least difference means more similar pic from others.

**Now we are going to add border and text to images:**

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

These are for text color – font stye – font size.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

All pieces gets borders and texts. Also their locations are declared here. Then append to pieces list. And this is 180. line. So the for loop is end.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

These codes are spread and connect pieces then makes ouptpu 3x larger. At least show’s the image.

**Output**

**çerez, atıştırmalık, yemek, gıda, kurabiye, pastane, şekerci içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Fourth and seventh piece’s titles are wrong. That’s because their RGB values are very similar with sixth and third pieces.

metin, ekran görüntüsü, yazı tipi, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

These lists are differences of RGB values with dataset. And the lowest one selected from these lists.