

Check Your Food (CYF)

Check Your Food also abbreviated as CYF is the name of the app that I am proposing. This app takes real time images of food and compares it to a database of spoiled food and food that is still good to eat. The app aims to show what food is still good to eat and what food might not be the safest to ingest. The app uses convolutional neural networking in order to predict whether a certain item of food will be labeled as 'good to eat' or 'bad to eat'.

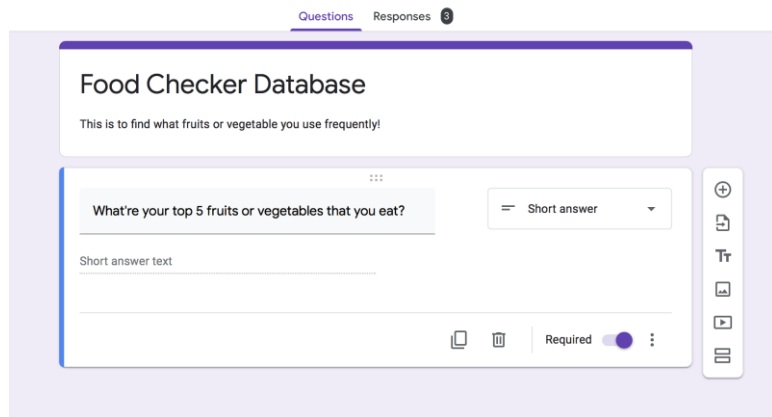
In times like these, where Stay at Home orders are in effect and having minimal exposure to the outside world is encouraged most families tend to stock up on their food supplies. Although it is integral for a family to have sufficient food supplies so that they do not need to make multiple food trips and extend their exposure to the outside world, it is also important to make sure that the food they have stored and are ingesting is actually safe to eat. Since many people may not know what the spoilt version of a certain food may look like, this app gives them a way to be certain that the food they are ingesting is indeed safe. In order to check the quality of this food, all you would have to do is take a picture of the food and then upload it to the app and the app will tell you whether or not it is safe to eat! This is useful to make sure families aren't ingesting expired items due to their bulk buying choices.

Firstly, in order to proceed with this app, I must obtain a dataset which is not already present. I can obtain the dataset by sending out a google form to friends, family, and fellow peers asking them to write down the food items that they use most often. Based on this I can conduct research on-line and obtain multiple images of both spoiled and good-quality versions of these food items. Once I obtain my images I must preprocess them and make sure they are all of a certain dimension and then label each image with either 'good-quality <insert food item name here>' or 'spoilt <insert food item name here>'. Obtaining a dataset of the best quality is definitely the most time rigorous part of this project. Next, I will use Tensorflow which is an open source deep learning library developed by Google and the programming language of Python to implement the food quality detection or fqd model. First, I will build an untrained model using different filters which result in different feature maps, these feature maps are sent to a pooling layer, the pooling and fully-connected layer are then implemented in the model. The model is then trained on 3 epochs (using 3 so the user does not have to wait for insanely long times on the presumably large database). Next, when the user interface, which I created using tkinter in Python, prompts the user for an image, the trained model will take your image of a certain pre-specified dimension and accurately tell you whether or not your version of the food is spoilt.

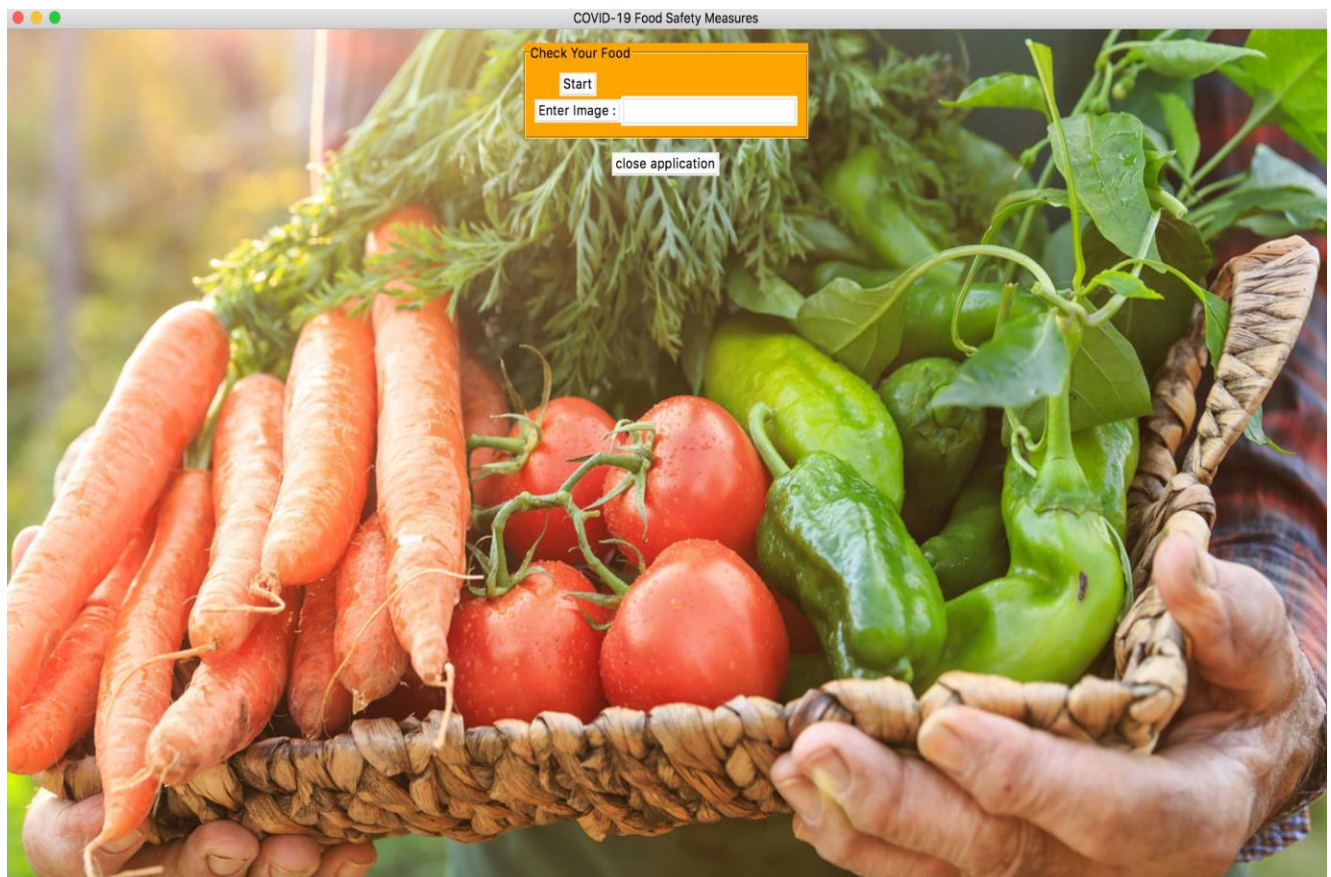
For the first part, I presume I will spend a good 2-3 weeks obtaining the data. Waiting 2-3 weeks will allow me to gather a database of the most frequently used foods. I will simultaneously be pre-processing and labeling each food item that is entered into the google form that I will send out. Therefore, the first step should take me a total of 3 weeks – one month depending on how long it takes me to clean the dataset by getting rid images that are not in the required dimensions and labeling each image. Since this is done mostly manually I will be evaluating each image as it enters the data pool. The next part consists of building the

model and training it to predict effectively, which means I must find the right layers in order to have the best output, this will presumably take around a week. I can ensure this by printing out the result and seeing that percentage match with a certain layer is significantly higher (the higher the better). The next part, I must build a working GUI for the user to interact with, this should take 2 days, I already have begun implementing the GUI for this project and am close to finishing. I can evaluate this by sending my GUI to my friends and re-implement it based on suggestions that they may have and add-ons that they would like to see while using the GUI. Allocating a couple more days to fix any errors and debug the code effectively, I believe this could be effectively implemented in around 1.25 – 1.5 months, right in time for summer!

For my preliminary work I have made a google form and a basic version of my GUI.



The screenshot shows a Google Form titled "Food Checker Database" with a subtitle "This is to find what fruits or vegetable you use frequently!". The form has a question "What're your top 5 fruits or vegetables that you eat?" with a "Short answer" dropdown menu. Below the question is a "Short answer text" input field. The form is set to "Required" and has a "Responses" tab at the top right.



I have made a google form asking potential users about their top 5 fruits or vegetables that they consume, and I will be using this information to create my database.

Additionally, above in my GUI if you click on the start button, a popup box will show asking the user whether they want to continue, with the options of yes or no. If the user decides to continue, then the GUI will continue and additionally will print

out a random fact or tip related to COVID-19 which is chosen randomly from a fixed number of hardcoded facts from the CDC. If the user does not want to continue and clicks no, the console shows a “come back later” message in the frame. Once, the user chooses yes, they must put an image name that exists in the same directory of this file in the entry button. I have not yet gotten to implement the rest of the GUI which would essentially be more user friendly and have more functionality, but all of the code compiles and works coherently. Feel free to write my code given above (sorry if the image is too small!) in your python console and run it to see how this works.

```

2 import tkinter as tk
3 import random
4 from PIL import ImageTk, Image
5
6
7 dataset = ["Diseases can make anyone sick regardless of their race or ethnicity.",
8           "The immediate risk of becoming seriously ill from COVID-19 is thought to be low.",
9           "Someone who has completed quarantine does not pose a risk of infection to other people.",
10          "Wash your hands often with soap and water for at least 20 seconds",
11          "In public, wear a cloth face covering that covers your mouth and nose.",
12          "Avoid touching your eyes, nose, and mouth with unwashed hands."
13          ]
14
15 window = tk.Tk()
16 window.title("COVID-19 Food Safety Measures")
17
18
19 window.geometry('1000x250')
20
21 my_img = ImageTk.PhotoImage(Image.open("L.jpg"))
22 background_label = tk.Label(window, image = my_img)
23 background_label.place(x=0, y=0, relwidth=1, relheight=1)
24 frame = tk.LabelFrame(window, text = "Check Your Food", padx = 10, pady = 10)
25 frame.pack(padx = 0, pady = 12.5)
26 frame.configure(background = "orange")
27 factLabel = tk.Label(frame)
28 label2 = tk.Label(frame)
29
30 def show():
31     check = e1.get()
32     print("Image: ", check)
33
34 e1 = tk.Entry(frame)
35 e1.grid(column = 1, row = 1)
36 btn1 = tk.Button(frame, text='Enter Image :', command=show)
37 btn1.grid(column = 0, row = 1)
38
39 def commence():
40     global factLabel
41     global label2
42     factLabel.destroy()
43     label2.destroy()
44     response = tk.messagebox.askyesno(title = "Lets Check that food", message = "Food checking program will begin, ready to proceed?")
45     if response == True:
46         tk.messagebox.showinfo("Food Check ", "Commencing Food Checking Sequence, Analyzing Images")
47         factLabel = tk.Label(frame)
48         factLabel.config(text = str(random.choice(dataset)))
49         factLabel.grid(column = 1, row = 0)
50         factLabel.configure(background = "green")
51     elif response == False:
52         tk.messagebox.showinfo("Food Check Halted", "Process terminated")
53         label2 = tk.Label(frame, text = "Come back later!")
54         label2.grid(column = 1, row = 0)
55         label2.configure(background = "green")
56     btn3 = tk.Button(frame, text="Start", bg = 'red', command = commence)
57     btn3.grid(column = 0, row = 0)
58
59 def close_window():
60     window.destroy()
61
62 buttonEnd = tk.Button(window, text = "close application", command = close_window)
63 buttonEnd.pack()
64 window.mainloop()

```

When it comes to similar projects, there do exist a few with a general idea that is the same. For example, there is a study that checks for spoilage of raspberries using convolutional neural networks done by Karthik Kuchangi and Jothi Prakash. There is also the company ImpactVision which does this slightly differently, but at an industry level using hyperspectral imaging for enhanced accuracy because hyperspectral images take into account what the human eye cannot see.

Link to raspberry spoilage study :-

<https://arrow.tudublin.ie/cgi/viewcontent.cgi?article=1170&context=scschcomdis>

Link to ImpactVision implementation description:-

<https://venturebeat.com/2018/12/12/how-impactvision-is-using-ai-to-detect-unripe-or-contaminated-food/>