Mattia Toffanin 2096045
Luca Vergolani 2089903
Giuseppe Labate 2095665

Pawns

Food recognition and leftover estimation

Project Overview

Food waste is a critical issue today, causing the loss of valuable food, depleting natural resources, and incurring organic waste management costs. Work and school canteens are particularly affected due to a lack of respect for food, over-ordering, and inattentive portion control. Implementing a tray scanning system to analyze leftovers can discourage waste and monitor habits, offering a potential solution.
The primary goal of this project is to create a computer vision system capable of scanning a canteen consumer's food tray before and after a meal to determine the quantity of leftover food for each type of food item.

General approach

The program receives an input of an image containing the tray before the meal and another one after the meal. For each of these images, a preliminary search for the plates is conducted using the Hough transform, followed by a search for food items outside of the plates. Once these searches are performed, a multilabel classification is conducted to identify the types of foods present both on the plates and outside of the plates. With the results of the multilabel classification, specific segmentation is performed for each type of food. After that, for each returned mask, a standard classification is carried out to determine the type of food it represents.

Histogram analysis and classification

One of the fist approaches we tried was to analyze the images with histograms. This idea was born by the understanding that one of the ways to recognise food is through the color distribution. First of all we tried with RGB space, but we haven’t got good results since all the histograms looked the same. At this purpose we tried to confront different reference food images and other food images to understand if there could be a sort of correlation. Specifically we tried covariance, intersection and correlation metrics. After several tries we understood that this approach was not robust and we tried to shift to another color space to completely discriminate against this approach. We decided to try HSV space since it has a better representation of illumination and color values. We observe that the distribution we get with histogram analysis was better than RGB color space and this leads us to explore new approaches for segmentation and food localization.

LBP analysis

Since we encountered a lot of difficulties using feature extraction methods and keypoints matching due to the high variation of food colors and shapes, we tried other methods such as LBP for texture analysis. We tried to classify reference food images with trial images comparing the resulting LBP histogram. But the texture analysis alone was not sufficient to classify food. For this reason we completely lost the idea of classifying food images without a classification ML or DL algorithm.

BOW classification

Before trying a strong classifier we tried to implement a classificator based on food features and keypointd mathing. To do so we created a small trial dataset using free distribution images. We noticed that the performance were obviously not that good since our dataset was small but also since the keypoints are not capable of representing the food changes and features. We tried to implement also a sliding window technique that compared every window with the bow reference to understand if reducing the window dimension could improve the results and also to try a first segmentation approach. This methods seemed to work but only with few classes and was not robust at all, even with the presence of a LBP histogram to make the BOW stronger we could’t improve the results. We then moved to other solutions.

Dishes detection by using hough transform

The first step towards the goal of this project was to identify the dishes wHere food is posed, this holds since the majority of the food in the trays is placed in dishes. At this purpose we tried to implement a hough circles robust enough to identify the dishes precisely but also that is capable of detecting circles with illumination, translation and scale changes in the image. To do so we firstly tried the main parameters such as minimum and maximum ray, and then we preprocessed the images enough to make them robust. Since not all the dishes are perfect circles and the scale is variable this method is not robust for big variations.

Outside the dish detection(bread)

After detecting the food dishes the other goal is to detect other foods outside the dishes. Since we observed that also the bread has a spherical shape we implement a similar approach as the one of the dishes. But to simplify the research we firstly masked all the dishes we detected to eliminate the most non probable regions. Then we threshold in the H channel of the HSV space to eliminate all the objects that have a high value like glasses and tissues. At this point we compute the hough transform looking at the S channel of the HSV. In this way after other pre-processing we were able to identify different regions where the bread could be placed. The next step was to expand these areas to make sure all the bread was detected. These areas are then going to be classified to get the hishes region score.

Multi-label classification

Approaches

The initial approach for food classification was based on creating a codebook using the bag-of-visual-words (BoW) technique, and a final classification was performed using a multilayer perceptron on the resulting BoW histogram. It appears that this method did not yield satisfactory results, leading to the decision to change the approach.

In response to the unsatisfactory results obtained with the previous method, the decision made was to adopt a deep learning approach. In particular, a preliminary training of an EfficientNetB1 network with a final multi-classifier was conducted using the Food101 dataset. Then, employing fine-tuning techniques, the weights of the previous architecture were saved and imported into another identical architecture, except for the last layer, which had a number of neurons equal to the number of final classes (13 classes + 1 class for the background). Finally, the new architecture was trained using a “handcrafted” dataset, consisting of approximately 1400 images of the 13+1 classes of interest.

The model trained with the Food101 dataset achieved an accuracy of approximately 80% on the validation set (the reason could be the large number of classes), whereas the final model on the 'handcrafted' dataset achieved a final validation accuracy of nearly 93%. The explanation could be that the number of classes is lower, and nonetheless, the model was already capable of extracting the right features from food images.

EfficientNetB1

EfficientNetB1 is a powerful deep neural network designed for computer vision tasks. It is renowned for its efficiency and top-notch performance. The architecture, based on convolutional neural networks, employs "compound scaling" to strike a perfect balance among network width, depth, and resolution. This balance ensures that the model is both lightweight and accurate.

EfficientNetB1 has been extensively trained on large image datasets, allowing it to excel in tasks such as image classification and object detection. Its efficient design makes it ideal for real-time applications and resource-constrained environments. Additionally, thanks to its transfer learning capabilities, it can be fine-tuned for specific tasks on smaller datasets, providing impressive results even with limited training data.

Food101 dataset

The 101food dataset is a collection of food images containing 101 different food categories. It serves as a benchmark for image recognition tasks in the field of computer vision. Each category consists of a varying number of images, providing diversity in food types and presentations. Researchers and developers often use this dataset to train and evaluate machine learning models for food classification and object detection applications. The 101food dataset is a valuable resource for advancing research in food-related image recognition and understanding.

Handcrafted dataset

Since the most famous food datasets did not contain all the classes required for our case study, it was decided to manually create a dataset for the final training of the model. This dataset is composed of 14 directories (13 classes + 1 class for the background), each with a name corresponding to the label to be associated, containing approximately a hundred images per class.

To ensure that the model predicts dishes with multiple food items, images containing multiple categories of food were added to each food folder where dishes typically include multiple items. For example, images containing both fish cutlet and potatoes were added to both the fish cutlet and potato folders. This approach helps the model learn to recognize and predict dishes with various combinations of food items.

Final multi-label classification

The final model is capable of providing predicted classes along with a confidence score. By examining the label with the highest score, the model identifies the primary food item in the image. Then, based on the type of food predicted, it checks if there are any other food items present in the image. The confidence score helps in assessing the model's certainty about its predictions, allowing for better analysis and decision-making.

Segmentation

To segment the food and separate it from the dish, we tried plenty of methods.
We tried without a good result to use histograms, to see, by checking the distribution of the colors, if we could apply a threshold that divides dish and food.
Then we tried again this method with hsv color space and different values for the lighting and the contrast, but again no major result.
We then tried to use kmeans which had a fair response from which we started developing our code.
By using hsv color space and particular lighting and contrast values (which were different and depended on the food label) and then applying kmeans method on this new image we reached a good food clusterization at the expense of dish and background clusterization.

Kmeans algorithm used color distances from reference colors as data features, where the reference colors were the general values of dish color and background color in that specific lighting and contrast.

So we took then the food cluster by finding its color in the clusterized image and masking it to pick the food segmented from the dish.

To achieve an optimal result, we applied to the food mask a series of morphological operations, such as closing and opening, to fill food's holes and eliminate useless areas.

Food separation

To distinguish one food from another, we picked the segmented food and applied a canny edge detection to a scale space of it.
We then noticed that, by applying a dilatation, we would achieve an image, where one food was black and the other white.
To obtain this result we found optimal parameters for each class of food which were worth for all their appearances.

So by exploiting this result, we masked one food and then use its negative mask to find the other one.

While trying different possible solutions to this purpose, we tried kmeans and again histograms thresholding, but also otsu thresholding on the s channel of the hsv image.

This channel was quite capable of distinguish the foods in the plate by itself, but we didn't continued with this path because we would had to place arbitrary thresholds for each food category, which isn't so good.

Single-label classification

This classification step is very similar to the process of multi-label classification. The only differences are that it no longer classifies the entire dishes but only the bounding boxes returned from segmentation, and it returns the single label with the highest score.

Metrics

Results

The majority of our trials didn’t have either good and strong results.
We couldn’t finish our project since we encountered several difficulties in finding a good approach, which took us most of our working time.
For this reason, our project is capable of detecting food in dishes and outside dishes, classify it, and label multiple foods in the same dish. It is then capable of segmenting food regions but not to split them with respect to the labels. Also we couldn’t finish the estimation of the metrics because we hadn’t finished the algorithm, the segmentation and the comparison between trays and leftovers. The only metrics we estimated are Map.

Hours of work and member contributions

Mattia Toffanin was in charge of the neural network manufacture.
Giuseppe Labate and Luca Vergolani were encharged to find the dishes masks to give to the network, to segment the food, and to distinguish one from the other.

Giuseppe Labate worked approximately 150 hours on this project
Luca Vergolani worked approximately 200 hours for this project
Mattia Toffanin worked approximately 150 hours for this project

Note

To check for the metrics, please run the code by following README file.
To view the food detection and food segmentation images, check “outputs/masks” directory.
We avoided to upload them into report file, because if so it would be heavy.