

---

# COMPUTER VISION

---

## *Food recognition and leftover estimation*

### Introduction

Food waste is an important problem in today's society, with several negative impacts such as waste of valuable food, waste of natural resources used to produce it and the cost of organic waste management. Once prepared, food is often not fully consumed leading to a large amount of wasted resources very evident especially in work and school canteen settings. Some of the main causes are lack of culture of respect for food, over-ordering of food, and lack of attention to portions by the canteen staff. By introducing a system for scanning the consumer's tray at the end of the meal, food leftovers could be analyzed to discourage waste and monitor consumers' habits.

The goal of this project is to develop a computer vision system capable of scanning a canteen consumer's food tray at the end of a meal to estimate the amount of leftovers for each type of food. Such a system must be able to analyze pairs of images: one of the tray before the meal and one of the tray at the end of the meal. From the first image, the system will recognize the various types of food ordered, keeping track of the initial quantity of each food; at the end of the meal, the system must analyze a new image of the tray, recognizing which types of food are still present and in what quantity.

In more detail, the system to be developed should be able to 1) recognize and localize all the food items in the tray images, considering the food categories detailed in the dataset; 2) segment each food item in the tray image to compute the corresponding food quantity (i.e., amount of pixels); 3) compare the “before meal” and “after meal” images to find which food among the initial ones was eaten and which was not. The leftovers quantity is then estimated as the difference in the number of pixels of the food item in the pair of images.



*Figure 1: Example of the pair of images to be analyzed. On the left, the customer's tray before the meal, on the right the tray after the meal with some food leftovers.*

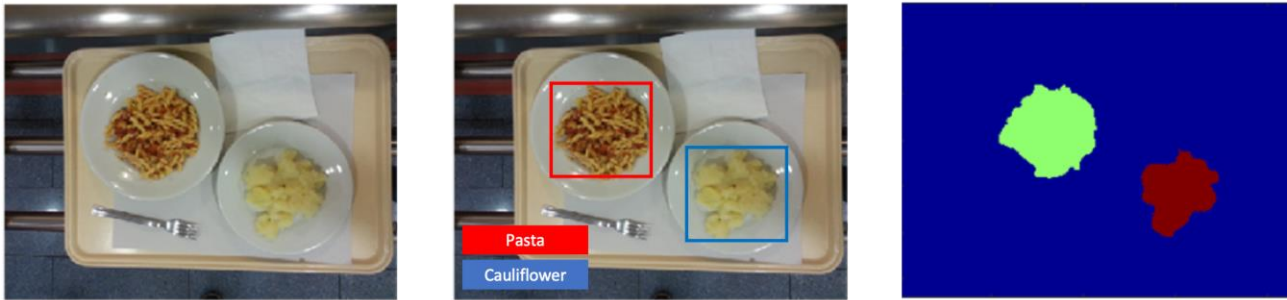


Figure 2: Example of the system outputs. From left to right: input image, food recognition and food segmentation.

As a case study for the development of the required system, consider a canteen scenario. Although this scenario includes some simplifications (e.g., controlled images acquisition, known weekly menu), the problem of food recognition is still a challenging problem due to the enormous variations in the tray and plate compositions. Moreover, the appearance of the same dish may change depending on how it is placed on the plate.

The system to be developed should be robust to all such conditions. In particular:

- it should recognize all the food on the tray, even food items that are not on a plate (e.g., fruit, bread);
- it should ignore any non-food object on the tray (e.g., smartphone, badge);
- it should segment different food placed on the same plate.

To assess the robustness of your system a benchmark dataset with annotations is provided at the following link:

[https://drive.google.com/drive/folders/1oTsvFwypHQ6PGV\\_VujG4LMvJA\\_Zrdt8Z?usp=share\\_link](https://drive.google.com/drive/folders/1oTsvFwypHQ6PGV_VujG4LMvJA_Zrdt8Z?usp=share_link)

The benchmark dataset consists of 8 different trays of food, each containing a first course, second course, side dish, and possibly salad and bread. For each tray, a “before” image is provided with the state of the food immediately after placing the order, and several “after” images with the presence of leftovers in the tray.

The benchmark dataset contains 3 “after” images for each tray, categorized by level of difficulty:

- 1) Dishes and objects in the same position on the tray for the “before” and “after” images;
- 2) Dishes and objects in a different order on the tray between the “before” and “after” images, but food only partially eaten (i.e., looking very similar between the “before” and “after” images);
- 3) Dishes and objects in a different order on the tray between the “before” and “after” pictures, and minimal leftover food.

## Performance measurement

For measuring the system performance, you should have a look and understand the following metrics:

- The mean Average Precision (mAP) - <https://learnopencv.com/mean-average-precision-map-object-detection-model-evaluation-metric/>
- The mean Intersection over Union (mIoU) - <https://towardsdatascience.com/metrics-to-evaluate-your-semantic-segmentation-model-6bcb99639aa2>

Such metrics shall be used to evaluate the food recognition system as follows:

- For food localization, the mean Average Precision (mAP) calculated at IoU threshold 0.5;

- For food segmentation, the mean Intersection over Union (mIoU) metric, that is the average of the IoU computed for each food item;
- For food leftover estimation, the quantity of food leftover estimated by your algorithm and the difference between your estimated quantity of food leftover and the quantity computed from the annotations; the quantity of food leftover is defined as the ratio  $R_i$  (in terms of pixels) between the segmentation mask in the “after” image and the segmentation mask in the “before” image:

$$R_i = \frac{\#pixels \text{ for food } i \text{ in the "after" image}}{\#pixels \text{ for food } i \text{ in the "before" image}}$$

For food localization and food segmentation you need to evaluate your system on the “before” images and the images for difficulties 1) and 2) of each provided tray in the dataset. For food leftover estimation, you need to evaluate your system on each pair of “before” and “after” images considering all difficulties levels.

The metrics mentioned above need to compare the output of your system against the ground truth, namely what is considered “the truth” for each output image. Ground truth annotations are already included in the dataset provided for the final evaluation of your system.

The ground truth is usually stored in a set of files, one file for each input image. The information representing the ground truth (e.g., the rectangle enclosing the food item or the pixels belonging to each food item in the image) is expressed in such a file based on a standard that you can define (and you should describe in the report). Such standard might be extremely simple, for example the provided benchmark dataset uses the following conventions:

- For food detection: every food item is found inside a rectangle defined by 4 parameters [x, y, width, height], where (x,y) are the top-left corner coordinates and width and height are the bounding box main dimensions; such parameters are listed in a row, one food item per row; each row contains also an additional parameter which describes the food category ID;
- For food segmentation: a grayscale mask where each pixel is assigned with a food category ID;

As references, the images in the benchmark dataset provided have been acquired at the Piovego canteen in Padova. Feel free to add more test images, taken from the internet or acquired using your camera. In case you use additional images for the development of your system, you must include those images and related annotations (or a link to them) in your final delivery.

If you use additional images, you are free to define your own standard. If you agree with other groups to share the ground truth collection, be sure to share the standard. The organization of the ground truth collection is completely free; if you wish, you can use the dedicated section in the moodle forum.

## Project delivery

The project must be developed in C++ with the OpenCV library. The only allowed exception is the usage of Python code for the deep learning part if you decide to exploit this family of techniques. **The project cannot be developed using machine learning / deep learning only.**

You need to deliver your project including:

- All the source code (both C++ and Python);
- CMake configuration files (the use of CMake is mandatory);

- A report (no page limit) presenting your approach and **the performance measurement on the dataset provided and linked above**. You shall report **the metrics and the output images** for every element in the dataset.

When delivering your project, you should **clearly identify** the contribution of each member in terms of ideas, implementation, tests and performance measurement. You can organize the work as you prefer: you are not forced to assign one specific step to each group member. Please also **include the number of working hours** per person in the report. This is needed for a monitoring on our side on the effort requested – the evaluation will not depend at all on the number of working hours, but on the quality of the result.

You should include in your report the results of the food detection, food segmentation and food leftover estimation (both output images and metrics values) for the test images in the provided benchmark dataset.

**It is not allowed to include code that was not written by any of the group members.**