

Hedia food recognition with faster RCNN

Authors: Jia Yi Johnny Ye s154074, Kia Kafaei s153811, Aslak Rønnow Franzen s163047, Jiayi Han s164229, Hjalte Reuss Schmidt s163471
Supervisor: Ole Winther DTU Compute, Technical University of Denmark, Jonas Meinertz Hansen, Hedia

Introduction



Figure 1: Object recognition [4]

Object Detection

This project is done in collaboration with Hedia who develops digital solution for diabetes patients. The goal of this project is to create an object detection algorithm.

Faster RCNN

- Faster than RCNN



Figure 2: Structure of VGG – 16 network, consist of 16 layers of convolutional network. [5]

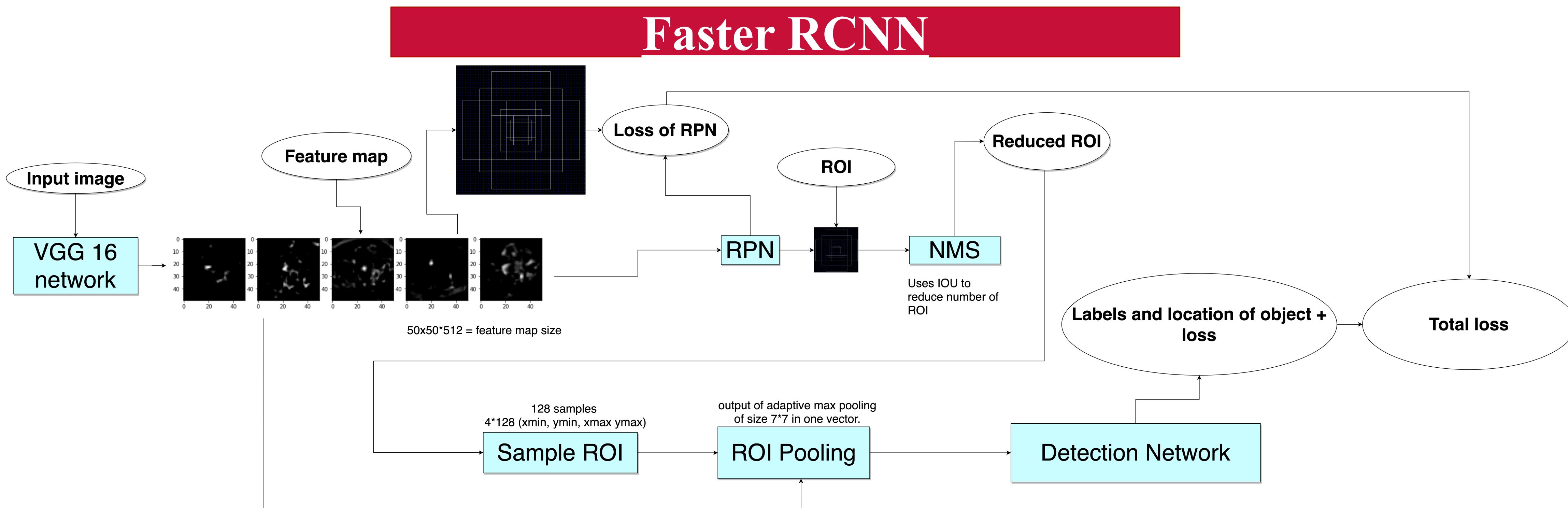


Figure 3: Faster RCNN:

For each image, the feature map is obtained using VGG-16 which is shown in figure 2, which is a predefined convolutional network that consists of 16 CNN layers. The feature maps are fed into the RPN (region proposal network) where it finds object bounds and scores for each position. Object bounds also called ROI (region of interest / region proposals). The number of ROI is then reduced using NMS (non-maximum suppression). After pooling there is the detection network which is also known as the classifier. [1][2]

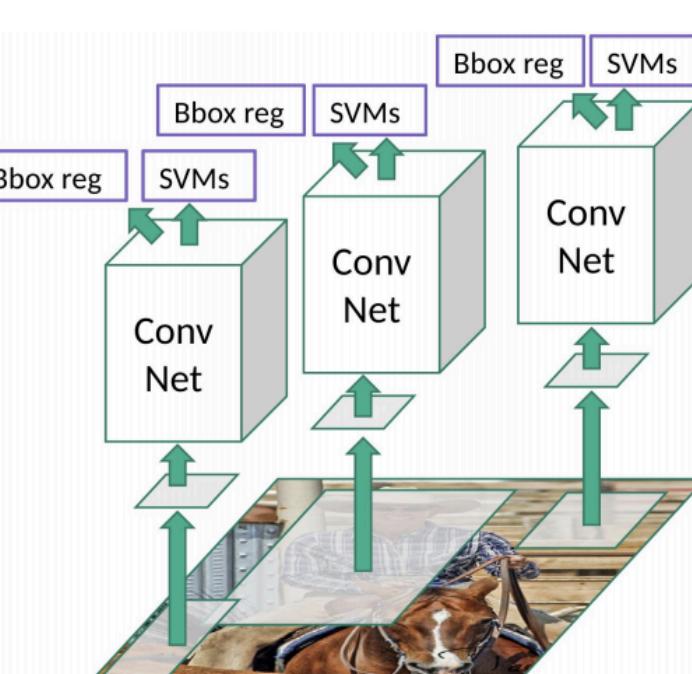


Figure 11: RCNN

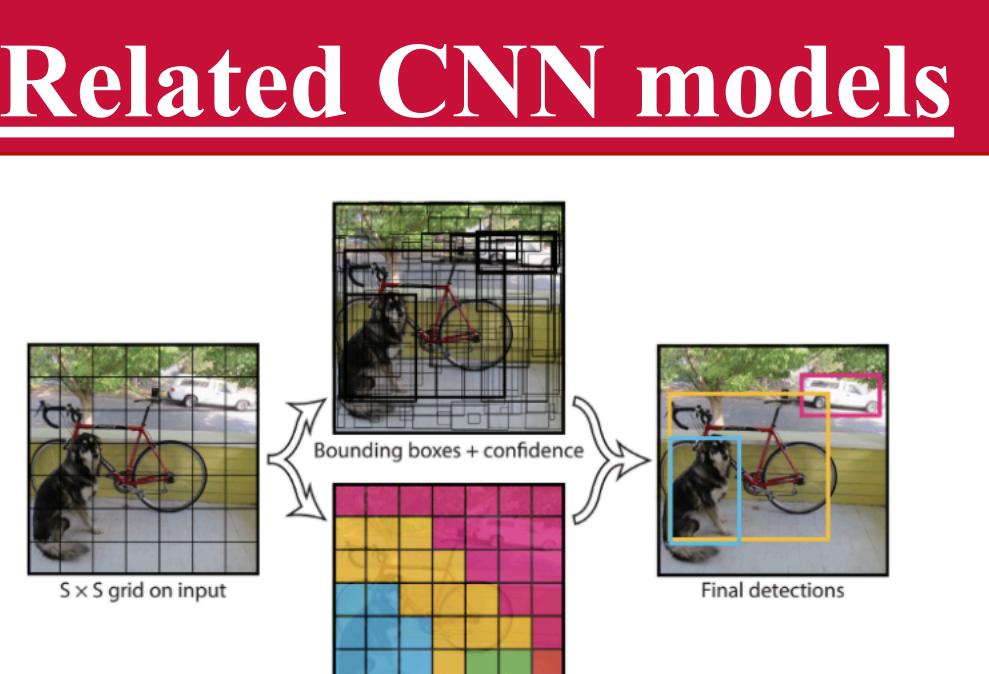


Figure 12: YOLO [3]

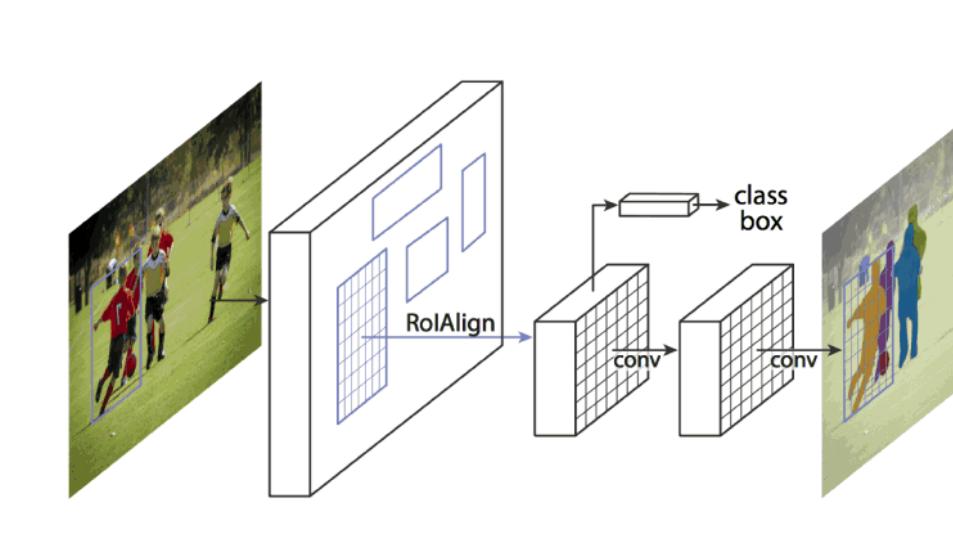


Figure 13: Mask RCNN

Performance

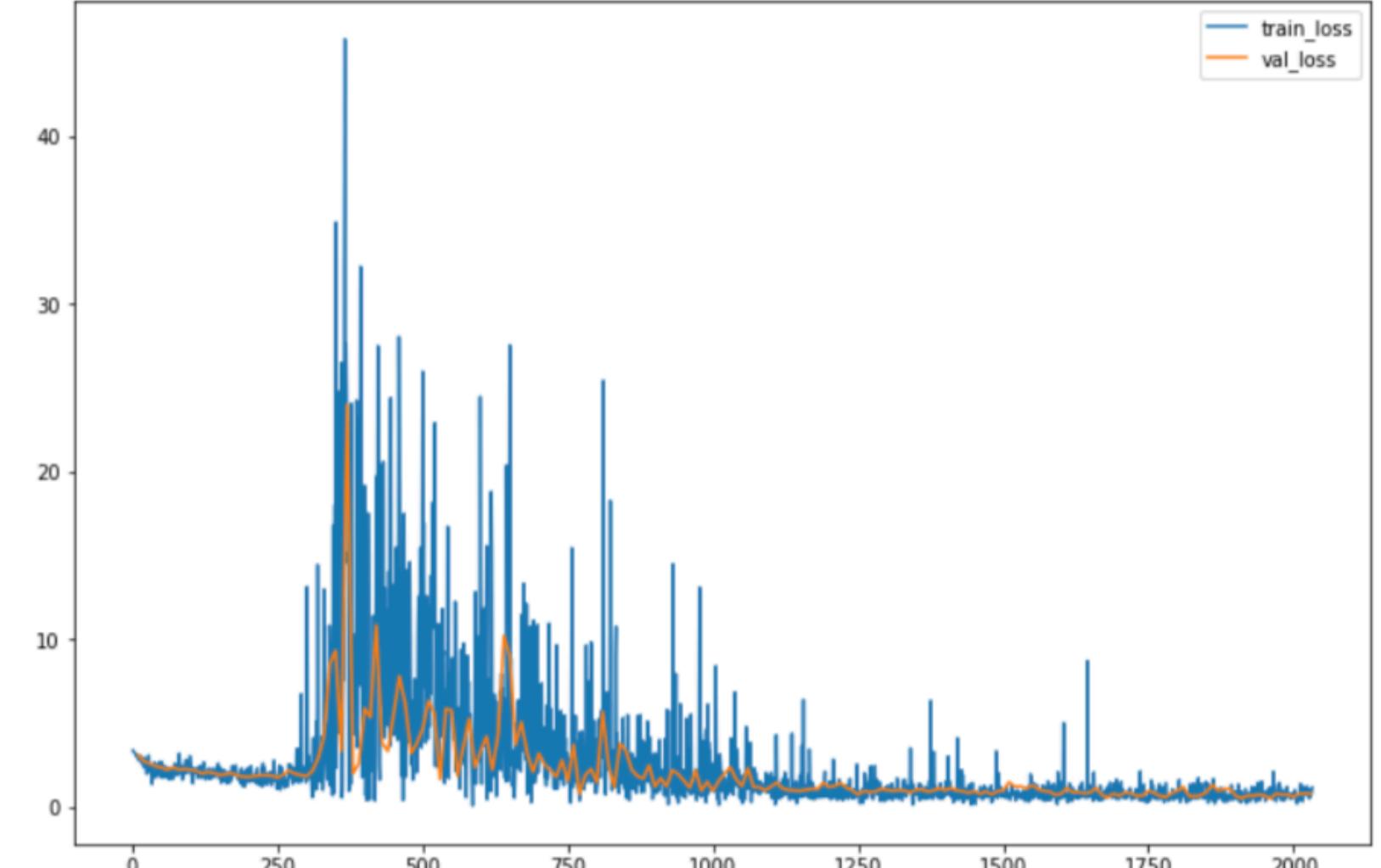


Figure 4 Total loss (training and validation)

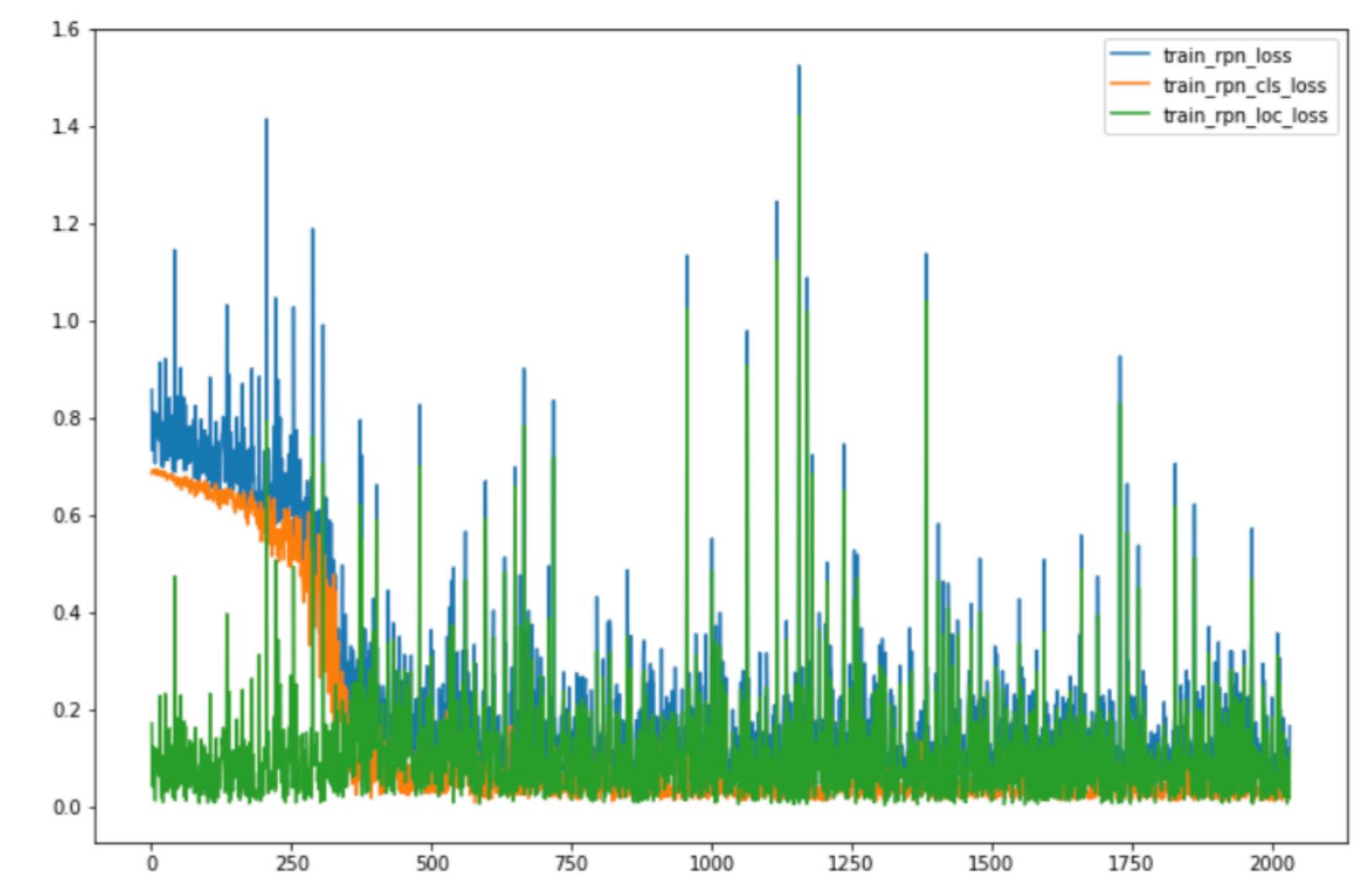


Figure 5 RPN location and class loss

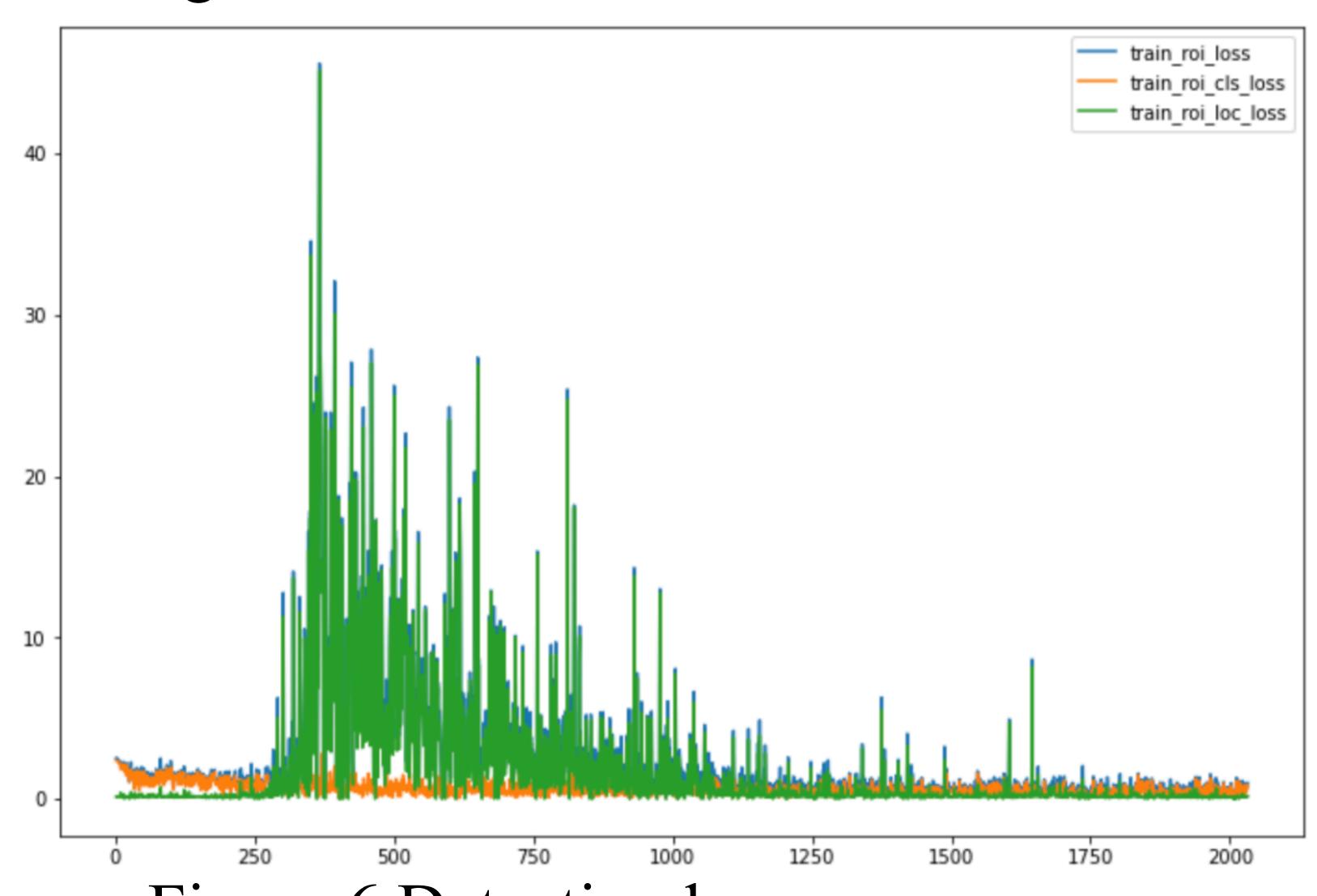


Figure 6 Detection loss

Distribution of data

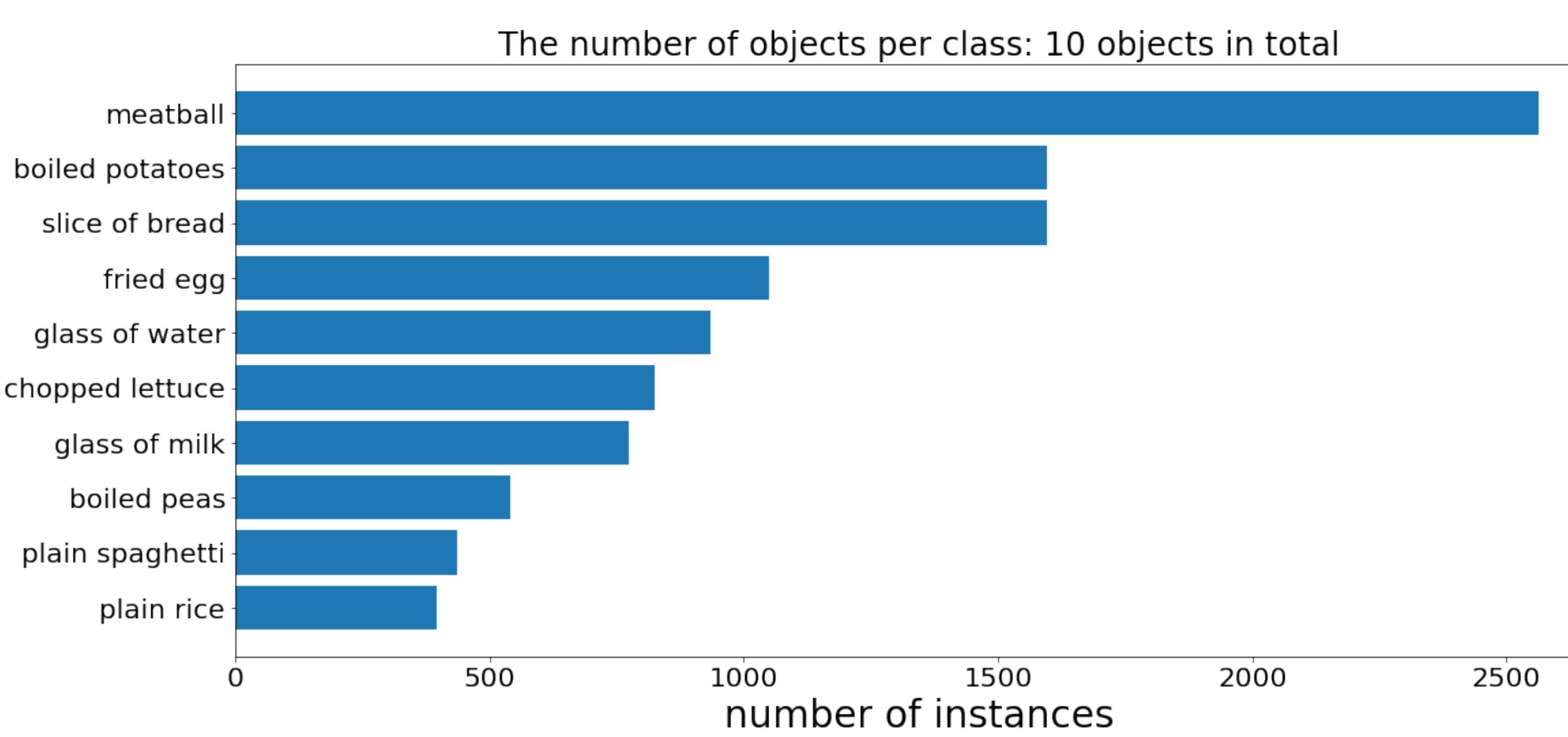


Figure 10: Data distribution, this project builds upon food images with pre-assigned region of food and labels in VASCO data format. The distribution of each food item within the dataset is shown in this figure. In total we have been provided with 5124 images with 10 different types of food.

Results / conclusion

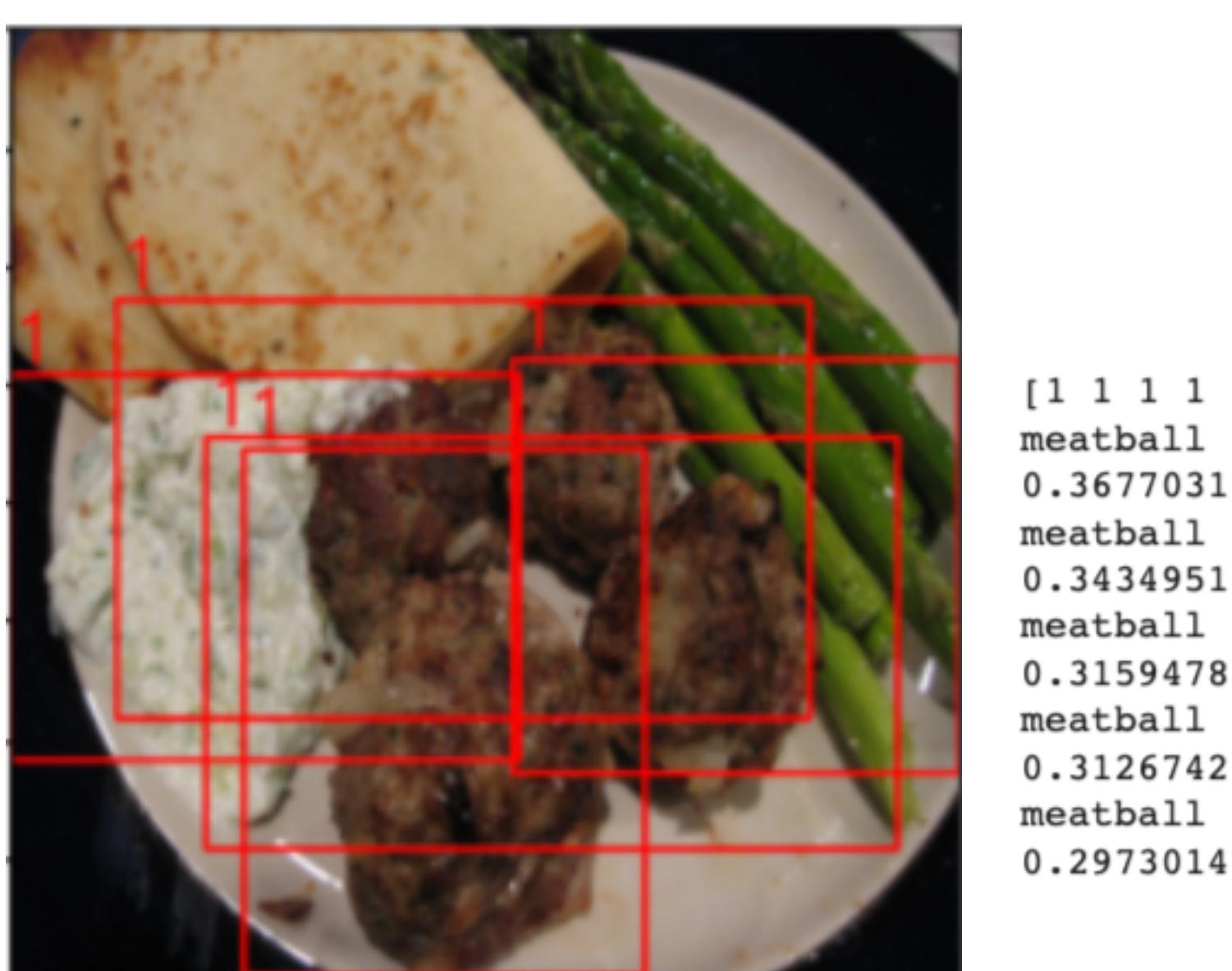


Figure 7: Rice

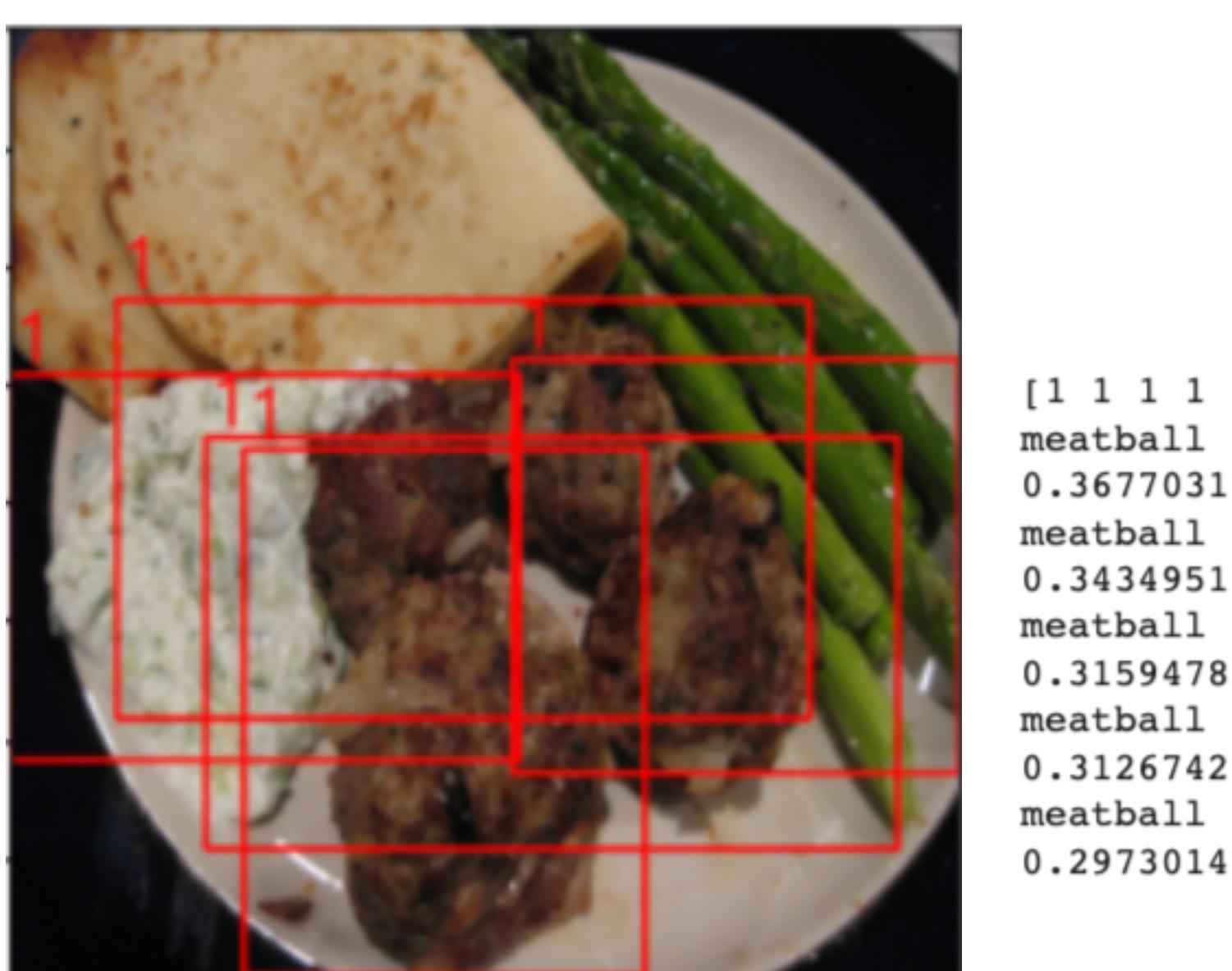


Figure 8: Meatballs

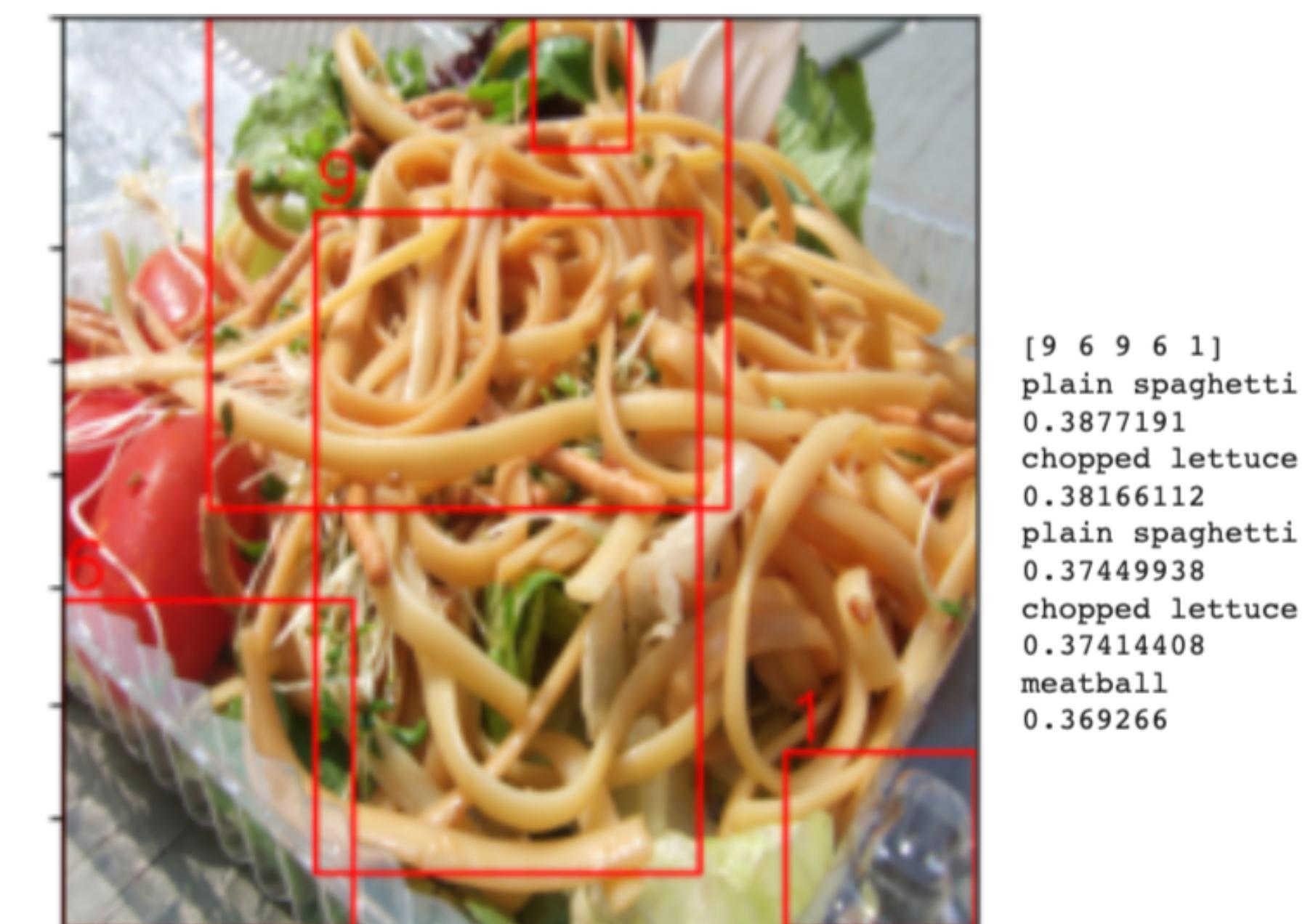


Figure 9: Meal with spaghetti

References

- [1] S. Ren, K. He, R. Girshick and J. Sun: Faster R-CNN: Towards Real- Time Object Detection with Region Proposal Networks
- [2] <https://medium.com/@fractaldile/guide-to-build-faster-rcnn-in-pytorch-95b10c273439>
- [3] <https://towardsdatascience.com/yolo-you-only-look-once-real-time-object-detection-explained-492dc9230006>
- [4] <https://medium.com/carat-global/evolving-influence-3c76c0386f7e>
- [5] <https://neurohive.io/en/popular-networks/vgg16/>