UPEC
UNIVERSITÉ PARIS-EST CRÉTEIL

International Master of Biometrics and Intelligent Vision

Report

Object Detection Using RCNN on Pears

Realised by:

Nour Benzeid

Professor in charge:

Delphine Maugars

2022-2023

BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

I. What is Mask RCNN?

Mask RCNN is variant of a Deep Neural Network; it detects objects in an image and generates a high-quality segmentation mask for each instance

To understand clearly we must first take a look at RCNN and Fast RCNN

1. Region-Based Convolutional Neural Networks (R-CNN)

R-CNN for Region-Based Convolutional Neural Network, is used for classification as well as objection detection with bounding boxes for multiple objects present in an image.

RCNN devide the image into number of boxes instead of grids, limited to 2000 regions, this regions are from diffrend sizes.

RCNN performe selective search to extract the 2000 regions and it's called regions proposals.

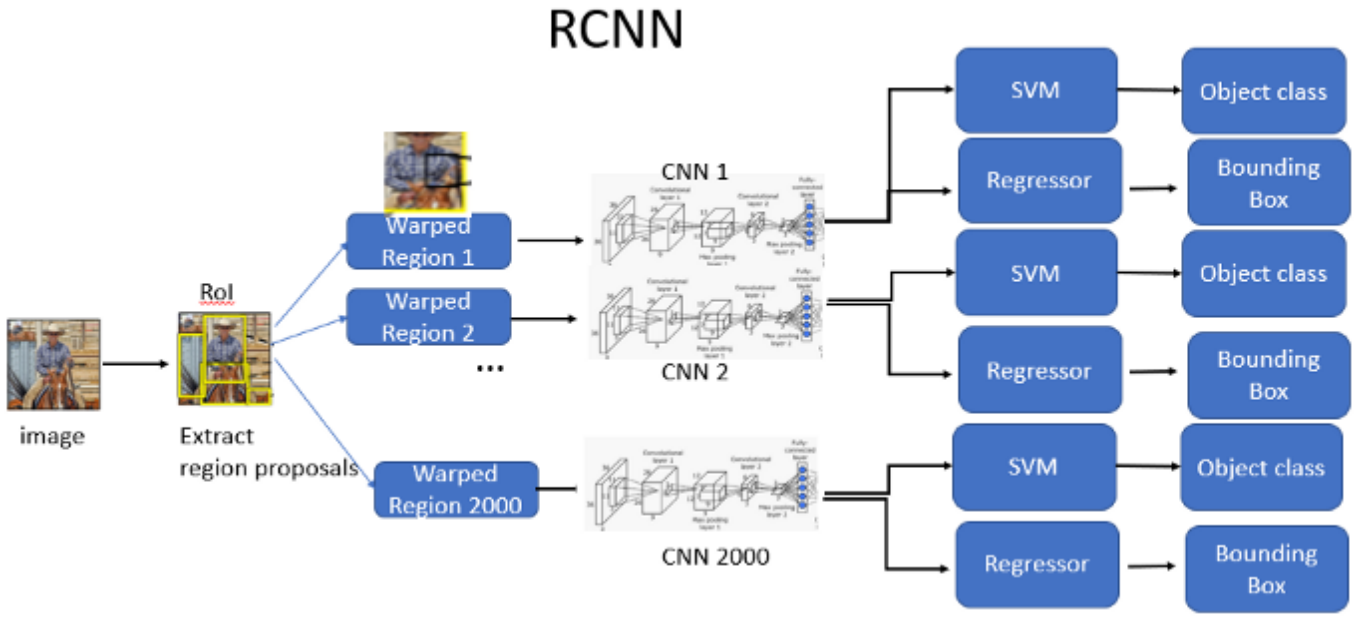


Figure 1 Steps of R-CNN

After generating many candidate regions via initial sub-segmentation to produce many little regions, RCNN use a greedy non-maximum suppression algorithm [1] to recursively combine similar regions into a large one, and then use the result to make the final region proposal.

[1] Non-Max suppression ensures detection of an object only once

BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

However, this algorithm is a slow algorithm, we repeat this steps for 2k regions to get 2k features extractions and classify them.

Because of that, RCNN was improved into what is known as Fast R-CNN.

2. Fast R-CNN

Fast R-CNN is a fast framework with deep ConvNets for object classification and detection.

Unlike R-CNN, Fast R-CNN uses a single deep ConvNet to extract features for the entire image once.

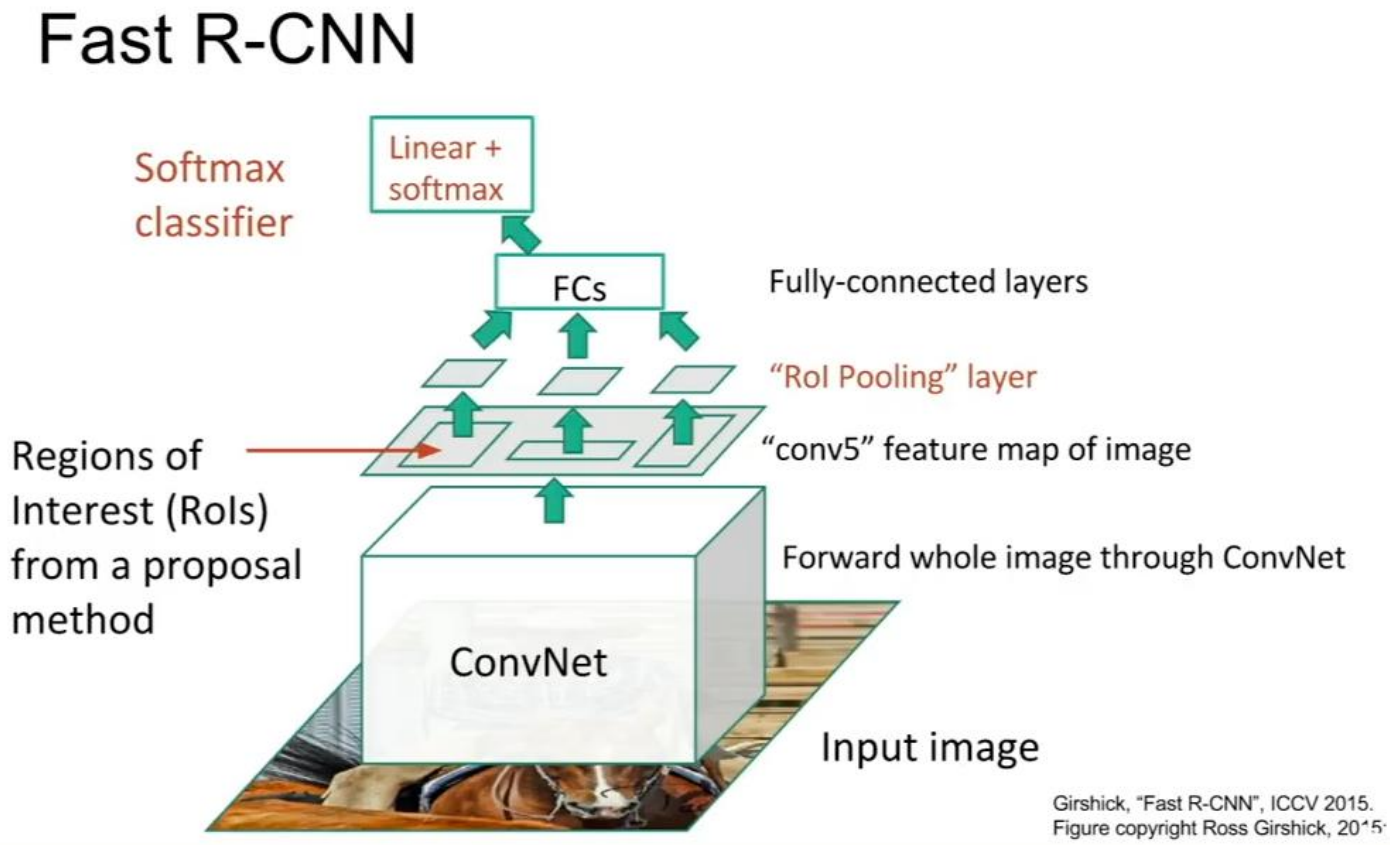


Figure 2 Fast R-CNN steps

So now the entire image goes through feature extraction, then in the features map we will have a region proposal method.

Even so, it still not fast enough specially for big dataset, because it uses selective search as a proposal method to find the Regions of Interest, which is slow and time consuming process.

BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

3. Faster R-CNN

Faster R-CNN does not use selective search as R-CNN and Fast R-CNN instead uses Region Proposal Network. It is a single, unified network for object detection

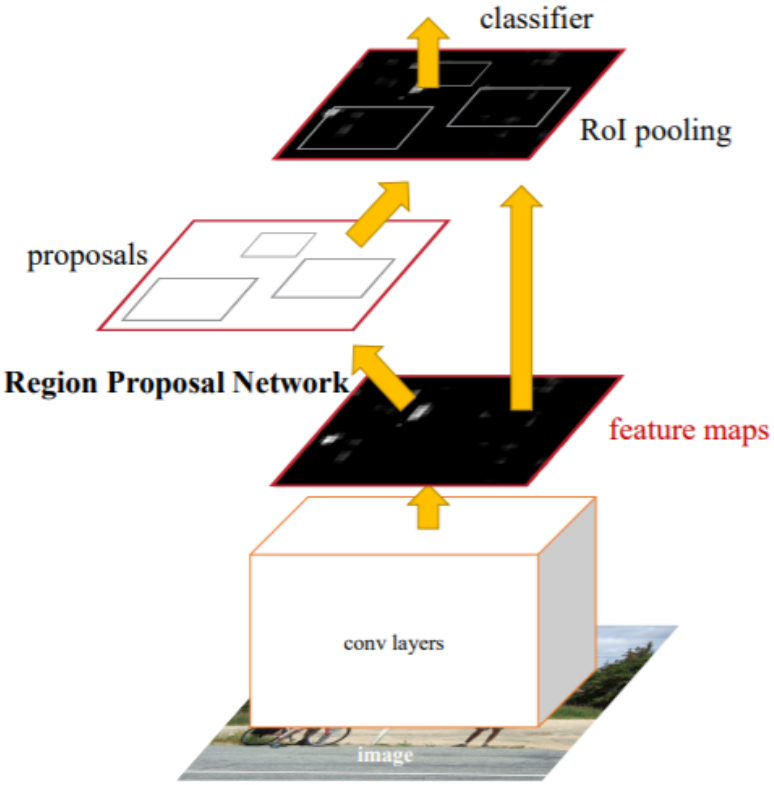


Figure 3 Faster R-CNN Architechter

Faster R-CNN takes an image as input and generates a feature map through a feature network.

RPN uses feature maps from feature networks as input to generate object proposal boxes and object scores.

The predicted region proposals from the RPN are then reshape using a RoI pooling layer, and fed into classification to predict the object classes and bounding box.

Only, one limitation of Faster R-CNN is that it is only able to classify and localize objects, but it is not able to segment them or generate masks for them.

BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

4. Mask R-CNN and how it works

Mask R-CNN was developed to address the limitation of Faster R-CNN by adding an additional branch to the network that is responsible for predicting a binary mask for each object in the image.

So now, plus label and Bbox we also have as output an object mask. The mask branch takes the region of interest (RoI) features from the Faster R-CNN network and processes them using additional layers to generate the mask predictions (use RoIAlign to locate relevant areas down to pixel level).

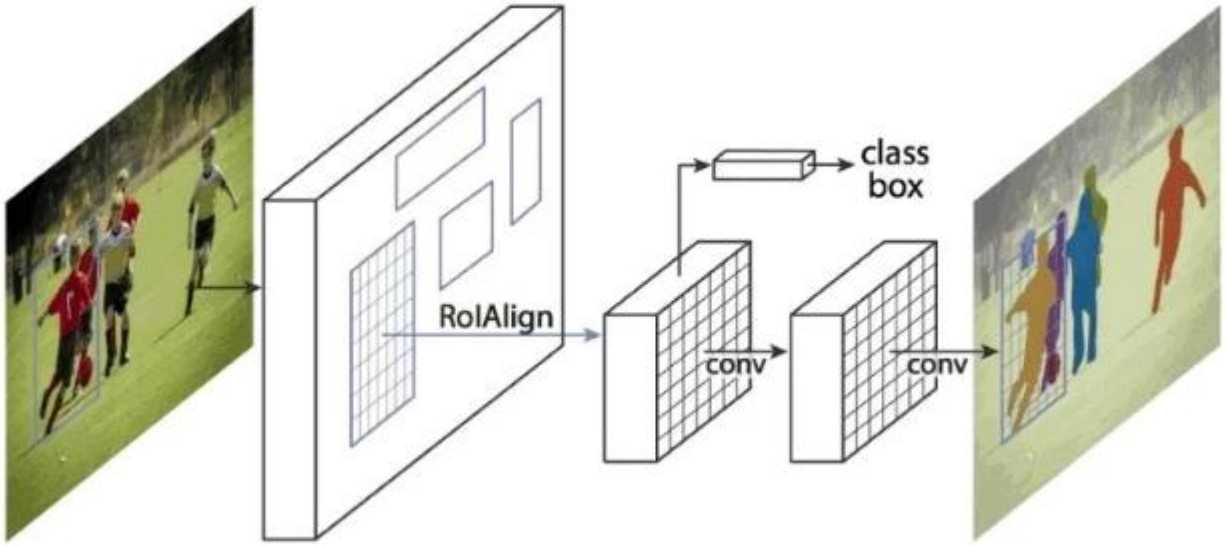


Figure 4 The Mask R-CNN Framework for Instance Segmentation

The key element of Mask R-CNN is pixel-to-pixel alignment, which is the main missing part of Fast/Faster R-CNN.

It uses the same two-stage process and the same first stage (RPN). In the second stage, MRCNN also outputs a binary mask for each RoI, predicting both category and box offset. This is in contrast to recent systems where classification depends on mask prediction.

BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

II. What is Coco?

COCO (Common Objects in Context) is a large-scale object detection, segmentation, and captioning dataset that was created to enable the development of advanced object detection and segmentation models. It contains a diverse set of images, including images of common objects, animals, scenes, and more, and is widely used as a benchmark dataset for evaluating the performance of object detection and segmentation algorithms.

Aimed at characterizing the relationship between objects and providing a semantic description of the scene.

COCO stores the annotation details for the bounding box in a JSON file, and has over 1.5 million object instances across 80 object categories.

III. Why Pear?

At the beginning, I wanted to work on rollers. I made the annotations of 80 images, 50 for the training and 30 for the validation, when i started the training it took me a lot of time and at the end my pc crashed.

I tried to redo it didn't work anymore and for the last attempt before deciding to change object, I decreased the number of images and as a result the model couldn't detect anything but a wheel ?

Therefore, I decided to change the object and to choose a simple object that does not exist in the coco bank.

Since I haven't a lot of time and the object is pretty simple, I create my own dataset of 40 images containing pears and use 32 to train a Mask R-CNN model specifically for detecting pears and 8 images for validation.

Annotator used: https://www.robots.ox.ac.uk/~vgg/software/via/via-1.0.6.html

I trained the upper layers of the network using coco as a base with parameters you can find on github.

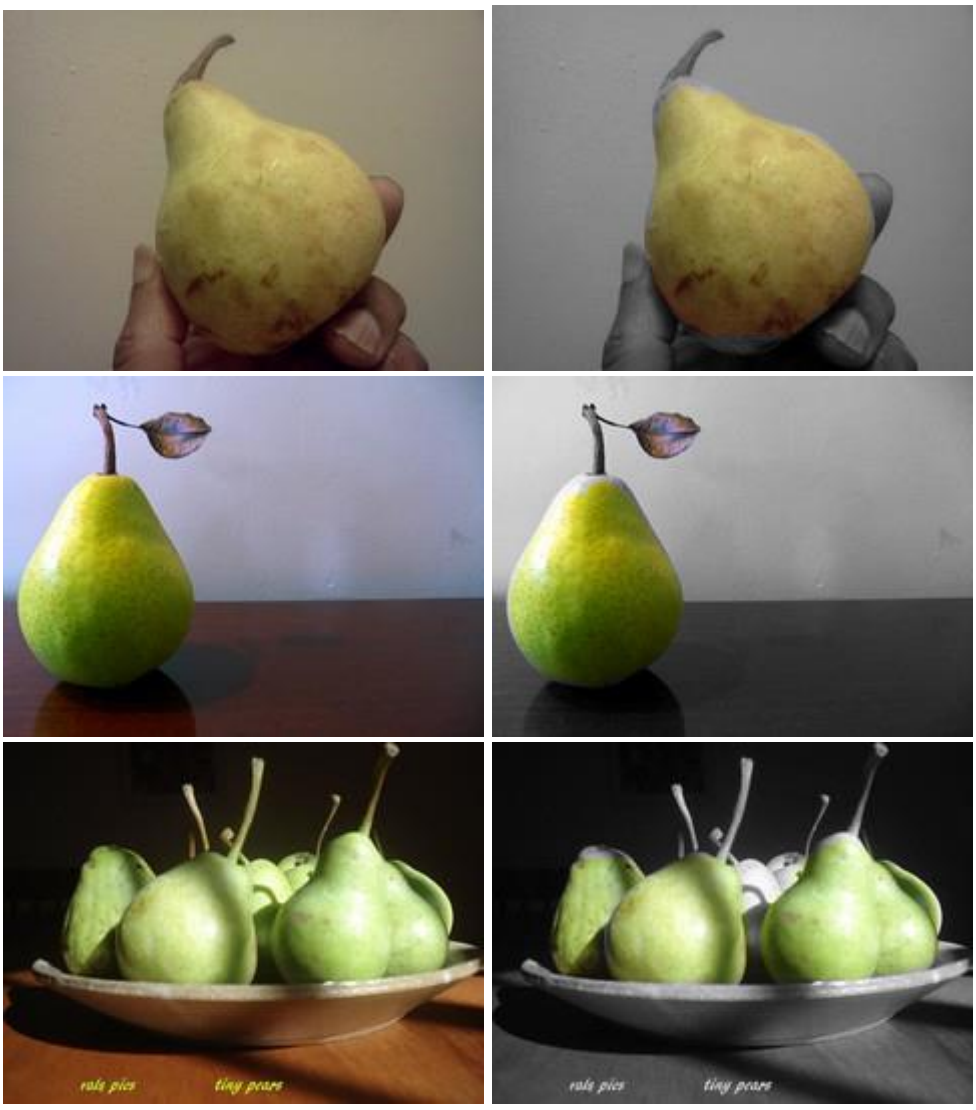
BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

To run the code I did make an instruction on git with my model (because of size I couldn't put it on github, instead I put it in drive this is the link https://drive.google.com/drive/folders/1_FvCDQX8HJR1cJwvcomf0CrIk4Ff8t4i?usp=share_link).

And it works, the model was able to learn to recognize pears in a variety of different contexts and environments.

IV. Results

I implements the code found on the project "Mask R-CNN for Object Detection and Segmentation". I used 7 images and one of them was generated by AI (the one with dogs).



BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr






BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

V. Conclusion

We can notice from the result, that Mask R-CNN is able to achieve both high object detection accuracy and good mask generation performance when the object is not so complicated, because as i said for rollers it didn't works very well (I know because I couldn't train properly my model but stil with a small data base it couldn't detect it).

BENZEID Nour
32221063
nbenzeidh@gmail.com
nour-el-houda.benzeid@etu.u-pec.fr

This project was an opportunity for me to deepen my knowledge of computer vision through the research I had to do for this report, especially through the two objects I chose and the study of the article.
The project is very interesting as the steps are clear and have all been covered in class

Note:

I am aware that the work i presented in this report isn't perfect since the direction i took is different from the initial plan, but these changes were more than necessary to get the work done (as i explained, my computer was unable to handle the training of the model with the object originally planned). I understand this deviation from the original proposal may affects my grade (I hope not), but i have put in a lot of effort and have done my very best to complete the project to the best of my abilities and my computer's. I hope that these changes will not significantly impact my overall grade for this course.

VI. References

1. Machine Learning Courses – Delphine Maugars
2. Mask R-CNN https://arxiv.org/pdf/1703.06870.pdf
3. R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e