YOLO Summary

Roger Marí [1], Joan Sintes [2], Àlex Palomo [3], Àlex Vicente [4]
Universitat Pompeu Fabra, Image Processing and Computer Vision Group
{ [1]roger.mari01, [2]joan.sintes01, [3]alex.palomo01, [4]alex.vicente01 }@estudiant.upf.edu

***Abstract*—This is a summary of the original paper by Joseph Redmon, Santosh Divvala, Ross Girshick and Ali Farhadi [1].**

I. Introduction

The authors of You Only Look Once (YOLO), reframe the problem of object detection as a single regression problem, straight from image pixels to bounding box coordinates along with class probabilities.Therefore, the system is able to predict what objects are present on the image and where they are located within it. The authors label the architecture as refreshingly simple, since it is a single convolutional network that simultaneously makes the predictions. They present it as an unified model that trains on full images and directly optimizes the detection performance, which adds some benefits over traditional methods.

II. Unified Detection

They unify the separate objects of object detection into a single neural network: The network uses features from the entire image to predict each bounding box. It also predicts all bounding boxes for an image simultaneously. The design allows end-to-end training and real-time speed computations while maintaining high average precision. This unified approach presents three main advantages over the traditional methods:

- *Fast computation*. Since the detection problem is raised as a regression a complex pipeline is not needed. It is as simple as running the neural network on a new test image to predict the detections. It runs at 45 frames per second with no batch processing. Moreover, they implemented a fast version of YOLO that runs at more than 150 fps. Furthermore, YOLO achieves more than twice the mean average precision of other real-time systems.
- *Global reasoning*. YOLO reasons globally about the image when making predictions. It sees the entire image during training and test time allowing to encode contextual information about classes and their appearance. Compared to Fast R-CNN, it makes less than half the number of errors in the background of images.
- *Objects general representation*. YOLO learns generalizable representations of objects. They prove it by training the network by natural images and testing it on artwork ones, outperforming by a wide margin, top detection methods such as DPM and R-CNN. This way, it is less likely to fail when applied to new domains or unexpected inputs.

The procedure that applies the system is the following one: First, it divides the input image into a grid of S x S. If the center of an object falls within a grid cell, that cell is responsible for the detection of that object. Each grid predicts B bounding boxes and its confidence scores measuring how confident is the model that the box contains an object. Each bounding box consists of 5 predicitons: x, y, w and h, being (x,y) the center of the bounding box and w and h, the width and height respectively. The fifth prediction is the confidence. Finally, the confidence prediction represents the Intersection Over Union (IOU) between the predicted box and any ground truth box. Moreover, each grid cell, predicts *C* conditional class probabilities.

The design (Fig. 1) of the network is follows a convolutional neural network inspired by the GoogLeNet. The initial convolutional layers extract features from the image, while the fully connected layers predict the output probabilities and coordinates. It has 24 convolutional layers and 2 fully connected layers. Furthermore, they use 1 x 1 reduction layers followed by 3 x 3 convolutional layers. The final output of the network is a 7 x 7 x 30 tensor of predictions.

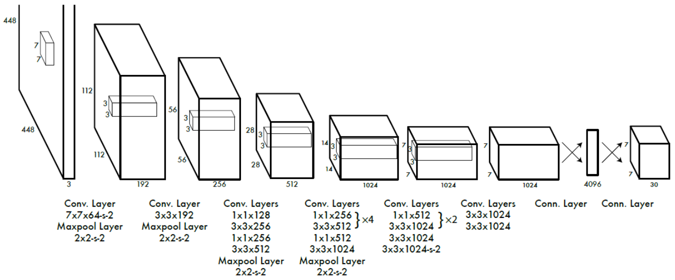

Fig. 1: YOLO convolutional neural network architecture.

A. Training process

They pretrain the 20 first convolutional layers on the ImageNet 1000-class competition dataset, followed by an average-pooling layer and a fully connected layer. Training the network for a week, they got a 88% of accuracy in the ImageNet. Then, they convert the model to perform detection. The fact of adding both convolutional and connected layers to pretrained networks can improve performance.Following this reasoning, they add four convolutional layers and two fully connected layers and two fully connected layers with randomly initialized weights. The input resolution of the network is finally increased from 224 x 224 to 448 x 448 to ensure fine-grained inputs. The final predicts both class probabilities and bounding box coordinates. Then the bounding box width and height are normalized by the image width and height so that they fall between 0 and 1. They use a linear activation function for the final layer and all other layers use the leaky rectified linear activation.

B. Limitations

There is a strong spatial constraint on bounding box predictions due to the fact that each grid cell only predicts two boxes and can have only one class. This constraint limits the number of nearby objects that the model can predict. Furthermore, the model struggles to generalize to objects with new aspect ratios or configurations since the model learns to predict bounding boxes from the data. Also, the features that the model learns are coarse due to the number of downsampling layers that the data meets. The main sources of errors are due to incorrect localizations. This error is increased by the fact that the loss functions treats errors the same in small bounding boxes versus large bounding boxes.

III. Experiments: Combining Fast RCNN and YOLO

The authors have proved that YOLO can be used to improve Fast R-CNN detections and reduce the errors from background false positives, giving a significant performance boost. The idea is that for every bounding box that R-CNN predicts, they test if YOLO predicts a similar box. If it is the case, a boost based on the probability predicted by YOLO and the overlap between the two boxes is given to that prediction. They get increases of mAP of approximately 3%.

IV. Conclusion

The authors of YOLO demonstrated that their model is simple to construct and can be trained directly on full images. It is trained on a loss function that directly corresponds to detection performance and the entire model is trained jointly.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.