Summary: You Only Look Once: Unified, Real-Time Object Detection (CVPR 2016)

Sungchul Hong[1], Eungyeom Ha[2], Jooyoung Song[3], and Heemook Kim[4]

[1]R&D center, wavekorea co.
[2]Department of Industrial Engineering, Yonsei University
[3]Department of Electronics & Computer Engineering, Hongik University
[4]Department of Aerospace Engineering, INHA

1 Introduction

The authors propose a novel object detection algorithm called You Only Look Once (YOLO) [RDGF16]. Unlike traditional object detection methods, YOLO takes a different approach. It processes the entire image in one pass, simultaneously predicting bounding boxes and class probabilities for detected objects. Thus, the YOLO model has the three advantages such as fast, fewer errors for background, and generalization.

2 Related Work

2.1 Deformable parts models

Deformable Parts Models (DPM) are models that use the Sliding Window to detect objects. DPM divides the pipeline into feature extraction, region classification, and predicting bounding box. YOLO has made all these a single network.

2.2 R-CNN

R-CNN uses Region Proposal (Selective Search) for object localization, involving multiple complex steps, leading to slow performance. In contrast, YOLO uses a grid cell approach similar to R-CNN's Region Proposal, but with spatial construction, resulting in fewer bounding boxes than Selective Search, achieving optimization in a single unified model.

3 Proposed Methods

This approach allows for end-to-end training, real-time performance, and high precision. The method utilizes an S $\times$ S grid to detect objects by pinpointing the grid cell containing the object's center. The bounding boxes are defined by 5 predictions: (x, y) coordinates relative to the grid cell, width and height relative to the image, and a confidence value indicating intersection over union (IOU as below) with the ground truth boxes. Additionally, each grid cell predicts conditional class probabilities $Pr(Object) * IOU_{pred}^{truth}$, which are influenced by the presence of an object.

3.1 Network Design

We created a convolutional neural network based on GoogLeNet for object detection using the PASCAL VOC dataset. The initial layers extract image features, while later fully connected layers predict probabilities and coordinates. Our network consists of 24 convolutional and 2 fully connected layers.

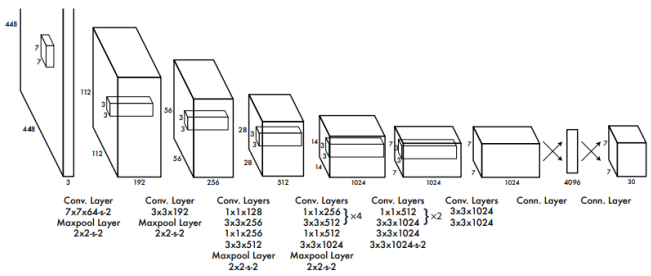

Figure 1: The Architecture.

3.2 Training

We pretrain our convolutional layers on ImageNet's 1000-class dataset using Darknet framework for both training and inference.

The last layer predicts class probabilities and bounding box coordinates with a linear activation, while the rest of the layers employ leaky rectified linear activation (i.e., Leaky ReLU function).

$$\phi(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.1x, & otherwise \end{cases}$$

We optimize our model for sum-squared error in the output. This choice simplifies optimization but doesn't perfectly match our goal of maximizing average precision. To address this, we adjust the loss: increasing loss for bounding box coordinates and decreasing it for confidence predictions when boxes lack objects. We use parameters λ_{coord} (set to 5) and λ_{noobj} (set to 0.5) to achieve this.

To partly address this, we predict the square root of the bounding box width and height instead of their direct values. The proposed loss functions are as follows:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} [(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2]$$

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{noobj} (C_i - \hat{C}_i)^2 + \sum_{i=0}^{S^2} I_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2$$

In YOLO, each grid cell predicts multiple bounding boxes. During training, we assign one predictor responsibility for each object. The predictor is chosen based on the highest current IOU with the ground truth, fostering specialization among bounding box predictors.

We train the network for around 135 epochs on PASCAL VOC 2007 and 2012 training and validation datasets. During 2012 testing, VOC 2007 test data is included for additional training. Our training employs a batch size of 64, with a momentum of 0.9 and a decay rate of 0.0005.

4 Experiments

First we compare YOLO with other real-time detection systems on PASCAL VOC 2007. Second we show that YOLO can be used to rescore Fast R-CNN and reduce the errors from background false positives. Third we also present VOC 2012 results and compare mAP to state-of-the-art methods. Finally, we show that YOLO generalizes to new domains better than others on two artwork datasets.

5 Conclusion

We introduced YOLO, a model that is well generalized to the new domain, fast, and simple to construct. Instead of the two-stage object detection method, we propose an integrated method in which one single neural network for the entire image predicts bounding box and class probability with only one calculation [**?**].

References

[RDGF16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

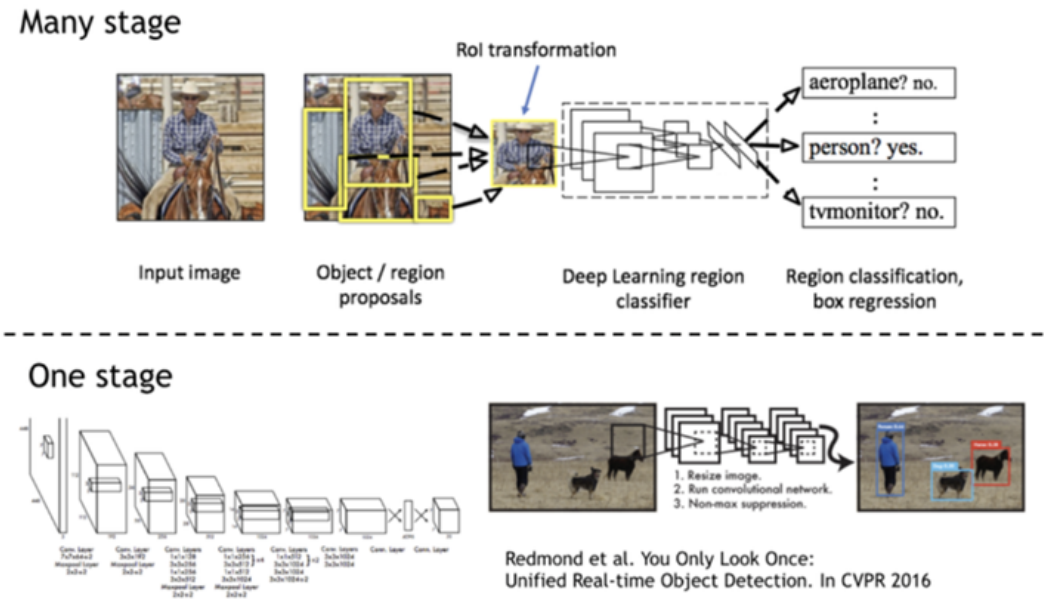

Figure 2: YOLO Detection System

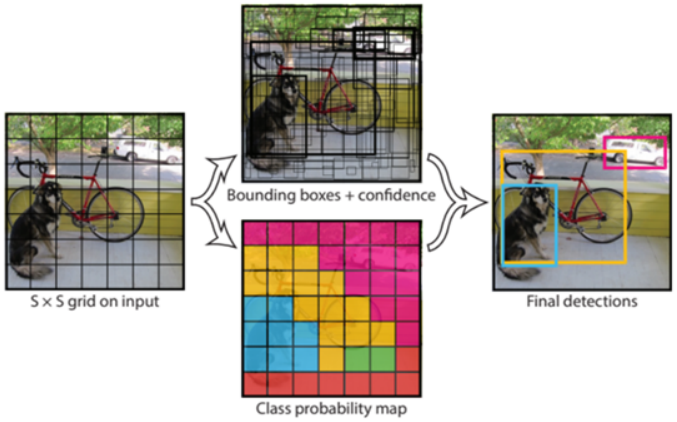

Figure 3: The Model

The Model - Unified detection :

Our system models detection as a regression problem. It divides the image into an S x S grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. These predictions are encoded as an