Final report (Option 1)

Course name: Multimedia Processing Technique (DD026_1594)
Project number: Option 1 - 1
Project name: Survey the paper: You Only Look Once: Unified, Real-Time Object Detection
Paper link: https://arxiv.org/abs/1506.02640
Student name: Ko Ji Woong. 201611772
Github of project: https://github.com/jiyko7/YOLO

Report:

1. Introduction

When humans glance at an image, They can instantly know what objects are in the image, where they are, and how they interact. Because human's visual system is fast and accurate, allowing us to perform complex tasks with little conscious thought. But computers can't know that. So they need Object Detection.

One of the famous Object Detections is R-CNN. R-CNN needs 2 stage for object detection. Region proposal and Region classification. These complex pipelines are slow and hard to optimize because each individual component must be trained separately.

YOLO reframes object detection as a single regression problem, So YOLO is simple. YOLO's single convolutional network predicts multiple bounding boxes and class probabilities at the same time.

YOLO has three advantages over other object detection.

First, YOLO is extremely fast. Because YOLO doesn’t need a complex pipeline.

Second, YOLO makes less than half the number of background errors. Because YOLO sees the entire image during training and test time.

Third, YOLO learns generalizable representations of objects. so it is less likely to break down when applied to new domains or unexpected inputs.

2. Unified Detection

YOLO unifies the separate components of object detection into a single neural network.

YOLO divide the input image into an S * S grid. If the center of an object falls into a grid cell, that grid cell is responsible for detecting that object.

Each grid cell predicts bounding boxes and confidence scores for those boxes.

Each bounding box consists of x, y, w, h and confidence.
x, y are the center of the box relative to the bounds of the grid cell.
w, h are width and height predicted relative to the whole image.
confidence is the IOU between the predicted box and any ground truth box. It reflects how confident the model is that the box contains an object and also how accurate it thinks the box is that it predicts. Each grid cell also predicts conditional class probabilities.

At test time, authors multiply the conditional class probabilities and the individual box confidence predictions, which gives us class-specific confidence scores for each box.

2.1. Network Design

YOLO's network architecture is inspired by the GoogLeNet model for image classification. It has 24 convolutional layers followed by 2 fully connected layers. It also uses 1 * 1 reduction layers followed by 3 * 3 convolutional layers instead of the inception modules used by GoogLeNet.

Fast YOLO uses a neural network with 9 convolutional layers and fewer filters in those layers.

2.2. Training

YOLO's final layer predicts both class probabilities and bounding box coordinates.
They use a linear activation function for the final layer and all other layers use the leaky rectified linear activation.

YOLO's loss function is based on sum-squared error by optimizing following three problems

First. It weights localization error equally with classification error.

Second. Posibility to lead model instability because of confidence score of no object cells.

Third. It equally weights errors in large boxes and small boxes.

Authors use the Darknet framework for all training and inference.
YOLO also use dropout and extensive data augmentation such as random scaling and translations, etc. to avoid overfitting.

2.3. Inference

YOLO is extremely fast at test time because it only requires a single network evaluation. It also uses Non-maximal suppression to fix multiple detections.

2.4. Limitations of YOLO

First. YOLO has spatial constraint that limits the number of nearby objects can predict. So YOLO struggles with small objects that appear in groups, such as flocks of birds.

Second. YOLO struggles to generalize to objects in new or unusual aspect ratios or configurations. Because YOLO has multiple downsampling layers from the input image.

Third. incorrect localizations. YOLO's loss function treats errors the same in small bounding boxes versus large bounding boxes.

3. Comparison to Other Detection Systems

Many research efforts focus on speeding up the DPM pipeline. But YOLO throws out the pipeline entirely. So YOLO is fast.

- Deformable parts models (DPM)
It uses a disjoint pipeline. But YOLO replaces that a single convolutional neural network. YOLO is faster, more accurate model than DPM.

- R-CNN
Each stage of complex pipeline must be precisely tuned independently and the resulting system is very slow. YOLO shares some similarities with R-CNN. But YOLO has spatial constraints on the grid cell proposals. So YOLO proposes far fewer bounding boxes, only 98 per image compared to about 2000 from R-CNN. Finally, YOLO combines individual components into a single, jointly optimized model.

- Fast and Faster R-CNN
While it offer speed and accuracy improvements over R-CNN, both still fall short of real-time performance.
- Deep MultiBox
Both YOLO and MultiBox use a convolutional network to predict bounding boxes in an image. But YOLO is a complete detection system.

- OverFeat
OverFeat efficiently performs but it is still a disjoint system. It cannot reason about global context and thus requires significant post-processing to produce coherent detections.

- MultiGrasp.
MultiGrasp simply needs to predict a single graspable region for an image containing one object. But YOLO predicts both bounding boxes and class probabilities for multiple objects of multiple classes in an image.

4. Experiments

YOLO can be used to rescore Fast R-CNN detections and reduce the errors from background false positives, giving a significant performance boost.

On VOC 2012, YOLO generalizes to new domains better than other detectors on two artwork datasets.

4.1. Comparison to Other Real-Time Systems

- Fast YOLO
It is the fastest object detection method on PASCAL. With 52.7% mAP, it is more than twice as accurate as prior work on real-time detection. YOLO pushes mAP to 63.4% while still maintaining real-time performance.

- Fastest DPM
It speeds up DPM without sacrificing much mAP. But it still misses real-time performance. It also is limited by DPM's relatively low accuracy on detection compared to neural network approaches.

- R-CNN minus R
It is much faster than R-CNN. But it still falls short of real-time and takes a significant accuracy hit from not having good proposals.

- Fast R-CNN
It has high mAP but at 0.5 fps. So it is still far from realtime.

- Faster R-CNN
The VGG-16 version of Faster R-CNN is 10 mAP higher but is also 6 times slower than YOLO. The Zeiler-Fergus Faster R-CNN is only 2.5 times slower than YOLO but is also less accurate.

4.2. VOC 2007 Error Analysis

Comparing YOLO and Fast R-CNN, YOLO struggles to localize objects correctly and Fast R-CNN makes much fewer localization errors but far more background errors. Fast R-CNN is almost 3x more likely to predict background detections than YOLO.

4.3. Combining Fast R-CNN and YOLO

By using YOLO to eliminate background detections from Fast R-CNN get a significant boost in performance. But this combination doesn't benefit from the speed. Because they run seperately and just combine the results. YOLO is so fast it doesn't add a lot of time.

4.4. VOC 2012 Results

On the VOC 2012 test set, YOLO scores lower than the current state of the art, because YOLO struggles with small objects.

Fast R-CNN gets a improvement from the combination with YOLO. Fast R-CNN + YOLO model is one of the highest performing detection methods.

4.5. Generalizability: Person Detection in Artwork

Artwork and natural images are very different on a pixel level. So R-CNN has high AP on VOC 2007 but drops off considerably when applied to artwork.
DPM maintains its AP well when applied to artwork. But it starts from a lower AP.
YOLO has good performance on VOC 2007 and its AP degrades less than other methods when applied to artwork.

5. Real-Time Detection In The Wild

YOLO is a fast, accurate object detector, making it ideal for computer vision applications. so when attached to a webcam it functions like a tracking system, detecting objects as they move around and change in appearance.

6. Conclusion

YOLO is simple to construct and can be trained directly on full images. Fast YOLO is the fastest general-purpose object detector. YOLO pushes the state-of-the-art in real-time object detection. YOLO also generalizes well to new domains.