Object Detection

Halil Eralp Koçaş

January 21, 2020

1 Object Detection

Object detection

1.1 Motivation

Various models are developed through years in the field of object detection. All these models are developed using different methods. Although all of these models are based on convolutional neural networks, their way of identifying objects in given images or frames are varies. Also, their backbone networks, scanning methods, multi-scale object detection, error functions are varies. These varieties affect the performance of models in different aspects such as small and large object detections, speed of the model, etc.

The aim in this article is to investigate, analyze, and compare the performances of various state-of-the-art object detectors through the time trained on video data. Then, promising detectors will be selected and these selected detectors will be analyzed on Video Object Detection dataset on both mean average precision and frame rate per second basis.

1.2 Literature in Static Object Detection

Static object detection refers to object detection in single image. In this context, object detectors do not use temporal information about objects in given image or frame sequences. Each given images are considered separately. The aim of investigating static object detection is to make a generalization from single frame to multiple frames. Understanding the pros and cons of each object detectors in various static object detection cases are important to make a generalization and creating a baseline for video object detection which requires more features to be considered for a well- performing detector.

TBC!!

1.3 Literature in Video Object Detection

Literature in Video Object Detection

2 Detector Features

As mentioned in Motivation section, in this article, the aim is to analyze object detectors based on various detector features. In the following five subsections, the following features will be examined:

- Backbone Networks
- Scanning Methods
- Multi-scale Handling
- Loss Functions
- Bells-and-whistles

These listed features are the essential features for detectors. When one studies the object detectors, one can see that major increases on the performances of object detectors are caused by changes on these features. These features will be analyzed and shown in the following subsections and section.

2.1 Backbone Networks

Backbone networks are the initial part of a detector's architecture. The given frame is processed for the first time in backbone networks. These backbone networks are implemented as Fully Convolutional Neural Networks. The used networks are the ones that are already proven to be perform well in images. The most frequent used ones are as following:

- VGG
- ResNet
- DarkNet
- Feature Pyramid Network
- Hourglass

The main function of these networks is to compute convolutional feature map over given frame, so that, these extracted feature maps are used to detect and localize objects in given frames.

2.2 Scanning Methods

There are many methods developed to scan given images through years. The following is a list for most frequent used scanning methods in object detectors:

- Sliding windows
- Region Proposals
- Grid Cells
- Anchor Boxes

All these methods have different advantages against each other. Also, Anchor Boxes are used together with other methods.

2.3 Multi-scale Handling

Multi-scale handling refers to detect objects that have different sizes in given frame. Multi-scale handling is a crucial feature for having a well performing detector since most of the objects in images have different sizes. There are various ways to handle this problem:

One of the solution is to run detection multiple times that in each run, the resolution of input frame has to be changed. Then, all the detected objects has to be combined after all iterations are completed. In case of multiple detection for same object, a suppression has to be performed to reduce single detection. Although this method works well, its runtime is slow.

Another solution is to use single feature map. In this method, single feature map is extracted from given frame and this feature map is passed through multiple convolutional layers to obtain a final feature map with more fine-grained features. Then, this feature map is used to predict the objects in given frame. This method is used to obtain fast detection but its performance is relatively worse than other methods.

Another solution is to use pyramidal feature hierarchy in which multiple feature maps are used to make prediction. In this method, a feature map is extracted from given frame and as it is in second solution, this feature map is passed through multiple convolutional layers. However, in each layer, the extracted feature map is used to make prediction. This method is relatively works slower than second solution but it performs better.

Another solution is to use feature pyramid network but this method will be examined detaily in section 3.2.8.

2.4 Loss Functions

Loss functions are used to measure the difference between ground-truth values and the predicted values in machine learning problems. Loss functions can also be called error functions. In object detection problems, loss functions are composed of two parts: regression difference and classification difference. Loss functions are the measure of difference between predicted regression box and ground-truth regression box and also, the difference between the correct class for object and the predicted class for object. These functions can vary based on design choices and different loss functions have different advantages and disadvantages. The most frequent used loss functions as following:

- L1 Loss
- L2 Loss
- Smooth L1 Loss
- Cross-Entropy Loss

2.5 Bells-And-Whistles

3 Static Detector Types

In this section, state-of-the-art object detectors will be examined. The following detectors are the best-performing detectors in their publication time. These detectors are divided into two subsection that are two-stage detectors and one-stage detectors. The difference between these one-stage and two-stage detectors is their way to process given frame to detect objects.

In two-stage detectors, detections process takes two stages that are region proposal and object detection. In first phase, model generates region of interest that are the regions in which objects can be found most likely. Then, in second phase, a classifier processes these regions to detect objects.

In one-stage detectors, detection process takes only one stage. Detector runs over a dense sampling of possible locations in given frame. This approach converges faster but its performance might be worse than two-stage detectors.

3.1 Two-Stage Detectors

3.1.1 Faster R-CNN

3.1.2 Mask R-CNN

3.2 One-Stage Detectors

3.2.1 You Only Look Once: Unified, Real-Time Object Detection

You Only Look Once (YOLO) refers to a unified, real-time object detection. YOLO is called unified since it performs simultaneous prediction over all image using its fixed number of bounding boxes in each grid cell. Also, whole detection pipeline is a single network, thus YOLO can be optimized end-to-end directly from given image to its detection performance.

The process of prediction can be examined as follows:

1. The given image is split into SxS grid.
    - Each grid cell is responsible for prediction of one object. The selection of which cell is responsible is chosen by the center of the object. If an object's center falls into a cell, that cell is responsible to predict that object.
2. Each grid cell has B bounding boxes to make prediction.
    - One prediction for each grid cell, independent of number of bounding boxes, causes YOLO to predict limited number of objects when there are multiple object present in a grid cell.
3. Thus, YOLO predicts B bounding boxes and one confidence score for each grid cell. Now, let's get into more detail for these predictions:
    - For each of these bounding boxes, five values are predicted: x, y, w, h, and confidence score. The values x and y are corresponding object's center coordinates on given image. The values w and h are corresponding object's width and height. The confidence score refers to detector's confidence whether a object exists in the corresponding bounding box and how accurate the bounding box is.

- The values x and y are normalized relative to the cell they belong and the values w and h are normalized relative to the given image, so that, all these x, y, w, and h values are between 0 and 1. Also, a class conditional probability for each grid cell is predicted. Thus, each category in possible class set has one class conditional probability in each grid cell.
- Thus, for SxS grid and B bounding boxes for each grid cell, predictions are encoded as S x S x (B * 5 + C) tensor.

3.2.2 YOLO9000: Better, Faster, Stronger

Before introducing YOLO9000, one has to explain YOLOv2 since YOLOv2 is the improved version of YOLO in both detection performance and speed while keeping real-time speed of detection. YOLO9000 refers to a joint training method on object detection and classification. Using this joint training method, one can detect objects that do not have any labelled detection data. Now, YOLOv2 will be explained and later, YOLO9000 will be explained.

As the name of the article suggested, YOLOv2 will be explained in three different perspective: Better, faster, stronger. While better and faster refer to performance increase of YOLO which is YOLOv2, stronger refers to YOLO9000.

1. Better

   - Analyzing the sufferings of YOLO, one can see that YOLO suffers from localization and having a low recall rate. Thus, the aim is to improve recall and localization while maintaining classification accuracy.

   (a) Batch Normalization

   - Applying batch normalization leads to significant improvement in performance and also it reduces the need for other forms of regularizations. One can remove dropout from the model without overfitting.
   - Batch normalization leads to 2% increase in mAP.

   (b) High Resolution Classifier

   - Training of YOLO is occured in two phases: first, convolutional layers are trained on ImageNet classification task on image resolution of 224 x 224. Then, the convolutional layers are trained for detection on image resolution of 448 x 448. This causes detector to switch itself for learning detection for new resolution simultaneously.
   - In YOLOv2, before detector is switched for learning detection, classifier network is trained for 448 x 448 resolution images for 10 epochs on ImageNet. Thus, a time is provided for network to adjust itself to work better on high resolution.

- High Resolution Classifier leads to 4% increase in mAP.

(c) Convolutional with Anchor Boxes

- xd

2. Faster

3.2.3 YOLOv3: An Incremental Improvement

3.2.4 SSD: Single Shot MultiBox Detector

Single Shot MultiBox Detector is a single deep neural network which aims to detect objects in real-time. The main improvement in speed comes from eliminating region proposals as Faster R-CNN does. Although eliminating region proposals increases the speed of detection, it reduces the accuracy significantly. Thus, three main improvements are applied to increase accuracy:

1. Multi-Scale Feature Maps for Detection
   - As mentioned in section 2.3, handling multi-scale is important feature for object detectors. It can reduce the mAP for detector's performance since objects of different sizes in given image cannot be detected well. To handle multi-scale detection, SSD uses multi-scale feature maps instead of using different sizes of input images. The size of feature maps decreases through the network's architecture. For larger feature maps, SSD can detect smaller objects and for smaller feature maps, smaller objects can be detected.
2. Convolutional Predictors for Detection
   - Small-size convolutional filters are used to make prediction for object detection. These convolutional filters are applied on extracted feature maps to compute both localization and class scores. Each filter computes a bounding box and corresponding class scores for each category.
3. Default Bounding Boxes and Aspect Ratios
   - SSD divides its feature maps into a grid and each feature map cells are associated with a set of default bounding boxes. These bounding boxes have fixed position relative to its corresponding grid cell. The aim of using multiple bounding boxes in each grid cell is to detect different-shape objects such as cars and people.
   - For different layers of SSD, default bounding boxes are customized for different resolutions. There are target aspect ratios and corresponding to these aspect ratios, default bounding boxes are calculated.

3.2.5 DSSD: Deconvolutional Single Shot Detector

Deconvolutional Single Shot Detector aims to contribute a new approach for object detection. The major changes in DSSD are to change the base network from VGG16 to ResNet101, using prediction modules, and adding deconvolution layers after convolutional layers of SSD. So, these changes will be examined as follows:

1. ResNet101

   - The aim of changing base network from VGG16 to ResNet101 is to increase accuracy. However, it does not improve accuracy by itself. That's why, the prediction module is used to increase performance.

2. Prediction Module

   - Due to the principle of improving sub-network can improve accuracy, original SSD approach for prediction is replaced with a prediction module which consists of a residual block. Thus, using ResNet101 with prediction module performs better than VGG16.

3. Deconvolution Layers

   - The aim of using deconvolutional layers is to include more high-level context in detection, so that, deconvolution module integrates information from both earlier feature maps and earlier deconvolution layers.

3.2.6 CornerNet: Detecting Objects as Paired Keypoints

3.2.7 RetinaNet

3.2.8 FPN

3.2.9 EfficientDet: Scalable and Efficient Object Detection

3.2.10 FreeAnchor: Learning to Match Anchors for Visual Object Detection

4 Datasets and Metrics

5 Performance Comparison

5.1 Accuracy and Real-Time Applicable

5.2 On COCO

5.3 In Video

5.3.1 MOT17Det

5.4 Quantitative Analysis

5.4.1 Based on Size

5.4.2 Based on Location on Image

5.4.3 Based on Visibility Ratio

5.4.4 Based on Confidence Scores

5.4.5 Based on Combinations

5.5 Qualitative Analysis