DATE: __/__/____

201900 Muhammad Hassan Mukhtar

Assigment 02

Artifical Neural Network

1.

In You Only Look Once there are several anchor boxes also known defined as prior boxes or default boxes.

V1 2 anchor boxes, No grid formation (single scale)

V2 5 anchor boxes, 13x13 grid (at each scale)

V3 3 anchor boxes (at 3 different scales)
13x13, 26x26, 52x52 grids (at each scale)

V4 Same as V3 Mosaic Training
combine multiple grids

V5 same as V4 and improved anchor box initilization.

V6 Introduced "anchor free" approch
uses key point Detection and Non maximum Sepration

V7 Introduced "Auto Assign Labels"
"dual-teaching"
In dual-teacher model V7 learn from 2 teacher one is rigid and other is ease in such manner knowledge distilled and learning is consistance.

DATE : __/__/__

V8

Same as previous models

- Anchor free
- 16x16, 32x32, 64x64 grid (at each scale)
- Mosaic Training (combine multiple grid)
- Improved auto assign label (IOU)

Spatial attention

1. Apply Average Polling and Max polling along channel axis and concat them.
2. On concat feature descriptor you apply Conv layer. which encodes where to emphasize or supress.
3. Then aggregate chanel info into 2D map by using two polling operations.

$$Ms(F) = \sigma(f_{7\times7}([AvgPool(F), MaxPool(F)]))$$
$$Ms(F) = \sigma(f_{7\times7}([f_{avg}, f_{smax}])$$

σ => Sigmoid function $f_{7\times7}$ => Conv operation with filter 7x7

New use cases :- Instance Segmentation.
Keypoint Detection.

2.

Region Based Conv Neural Network family

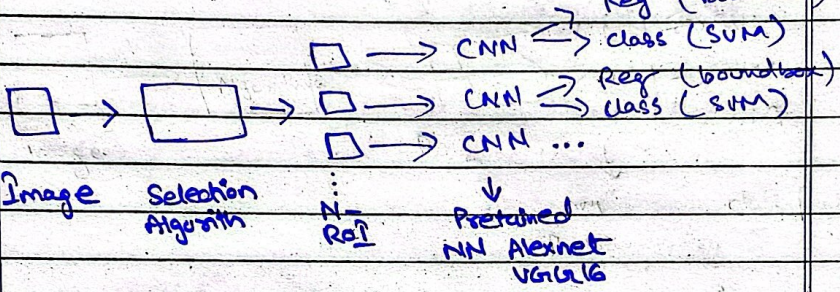
RCNN uses selective search algorithm to propose Region of Interest (RoI) and then uses CNN to classify each RoI.

DATE: __/__/____

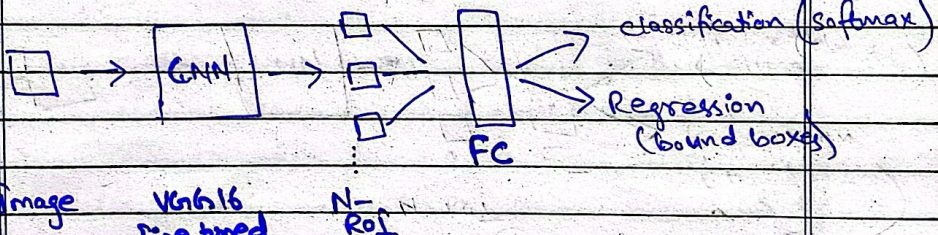
Fast RCNN uses shared convolution layer to whole image and all RoI rather then dealing with each seprately (Reduces computational time)

Faster RCNN uses Region Proposed Networks (RPN) to generate RoI which is faster than selective search algorithm. These networks also uses feature map to predict bounding boxes.

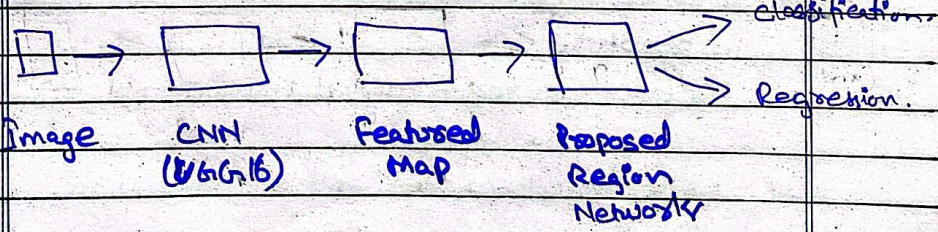
RCNN



Fast RCNN



Faster RCNN



Sliding Windows in CNN:

9 Anchors of different scales $\frac{1}{4}, \frac{1}{8}, \frac{1}{16}$ are applied to featured map. Then fed to proposed Region Networks.

DATE: __/__/____

Non-Maximum Suppression:

On ouput of RPN it is applied to remove duplication of detection.

Note:: Bounded Box with max confidence is choosen.

3.

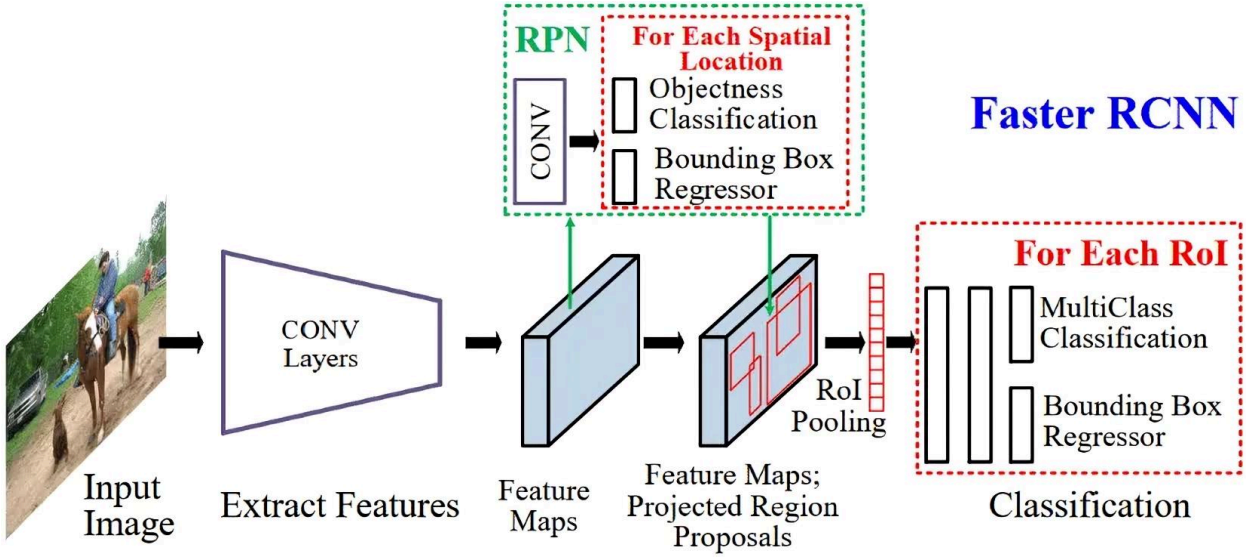
Key Differences:

1. RCNN uses selective search Algorithm while Yolo uses single neural network to predict bounding boxes.
2. RCNN has two stage approach (Region proposal and classification). Yolo is single stage.
3. RCNN prioritize accuracy over speed while Yolo prioritized speed over accuracy.
4. Yolo is used on live video stream while RCNN benefits for high accurate decision making on regions.

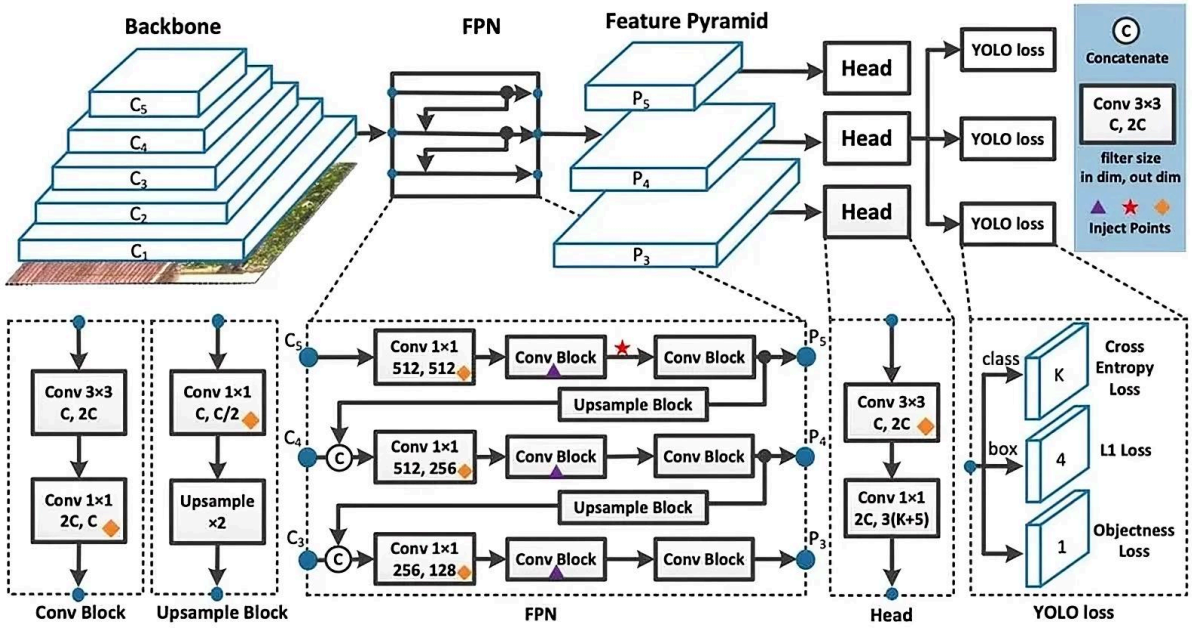
In structure

5. RCNN has Region Proposal Network and RoI pooling extra layer.
   while Yolo uses one grid cell generation.

Architecture Difference



Faster RCNN



YOLO v8

Faster R-CNN:

- **Two-stage Detection:** Faster R-CNN follows a two-stage detection process. In the first stage, it proposes regions of interest (RoIs) using a region proposal network (RPN). In the second stage, it classifies these proposed regions.
- **Region Proposal Network (RPN):** The RPN generates region proposals by sliding a small network over the convolutional feature map.
- **Region of Interest Pooling:** After obtaining the region proposals, Faster R-CNN uses a process called region of interest pooling to extract fixed-size feature maps from these regions, which are then fed into a classifier.
- **Backbone Network:** Typically, Faster R-CNN uses a pre-trained convolutional neural network (CNN) such as VGG16, ResNet, or a similar architecture as its backbone network to extract features from the input image.

YOLO (You Only Look Once):

- **Single-stage Detection:** YOLO operates as a single-stage detector, meaning it performs both object localization and classification in a single pass through the network.
- **Grid-based Detection:** YOLO divides the input image into a grid and predicts bounding boxes and class probabilities directly from the grid cells.
- **Unified Framework:** YOLO predicts bounding boxes and class probabilities using a single neural network architecture, without needing separate region proposal and classification stages.
- **Loss Function:** YOLO uses a joint loss function that combines localization loss (for bounding box regression) and classification loss, optimizing both tasks simultaneously.
- **Feature Extraction:** YOLO has its own feature extraction network, typically composed of convolutional and pooling layers, which extracts features directly from the input image.