medium.com

Object Detection and Instance Segmentation: A detailed overview

Shaunak Halbe

10-12 minutes

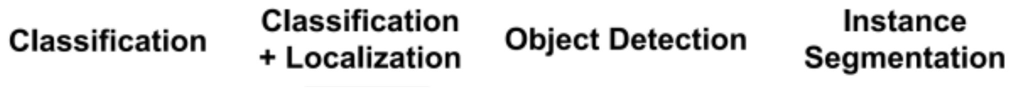
Object Detection is by far one of the most important fields of research in Computer Vision. Researchers have for a long time been interested in this field, but significant results were produced in the recent years owing to the rise of Convnets as feature extractors and Transfer Learning as method of passing on previous knowledge. Early object detectors were based on handcrafted features, and employed a sliding window based approach which was computationally inefficient and less accurate. Modern techniques include Region Proposal Methods, Single Shot Methods, Anchor Free Methods and so on .

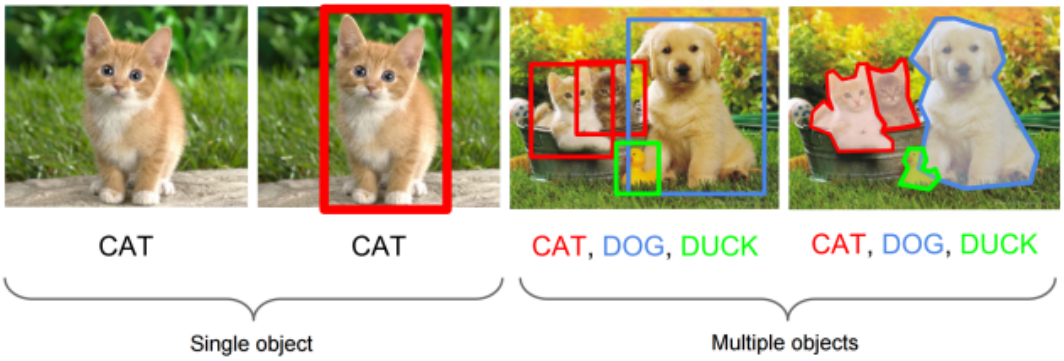
A) Object Detection : Object Detection refers to the method of identifying and correctly labeling all the objects present in the image frame.

This broadly consists of two steps :

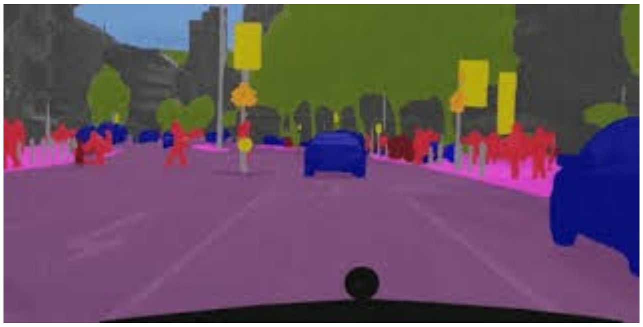
1 : **Object Localization** : Here, a bounding box or enclosing region is determined in the tightest possible manner in order to locate the exact position of the object in the image.

2: **Image Classification**: The localized object is then fed to a classifier which labels the object.

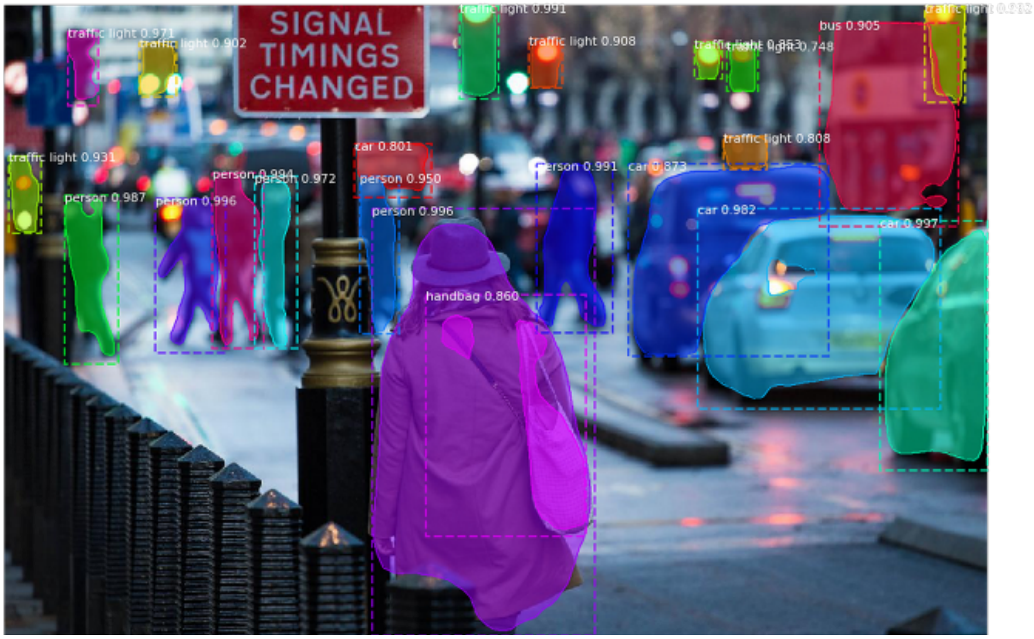




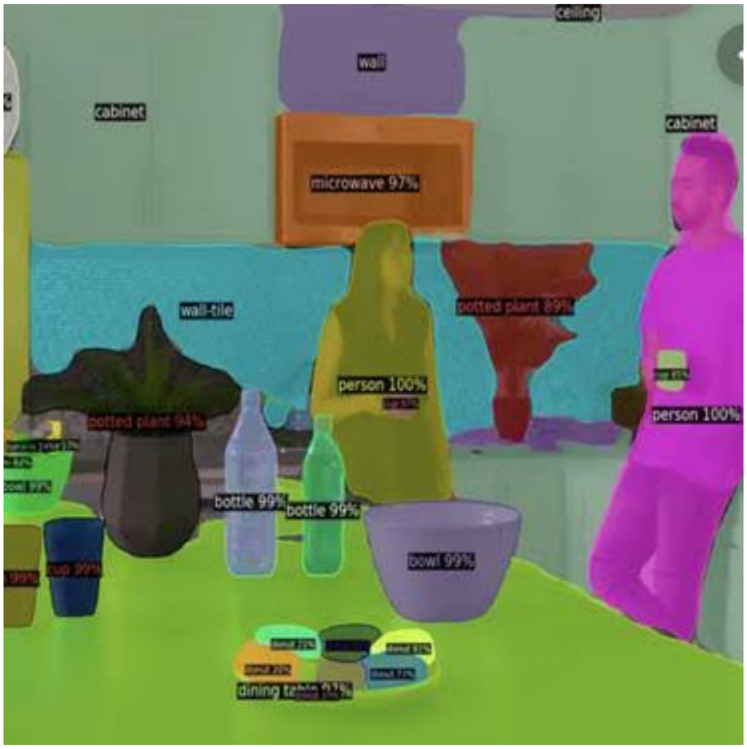
B) Semantic Segmentation: It refers to the process of linking each pixel in the given image to a particular class label. For example in the following image the pixels are labelled as car, tree, pedestrian etc. These segments are then used to find the interactions / relations between various objects.



C) Instance Segmentation: Here, we associate a class label to each pixel similar to semantic segmentation, except that it treats multiple objects of the same class as individual objects / separate entities.



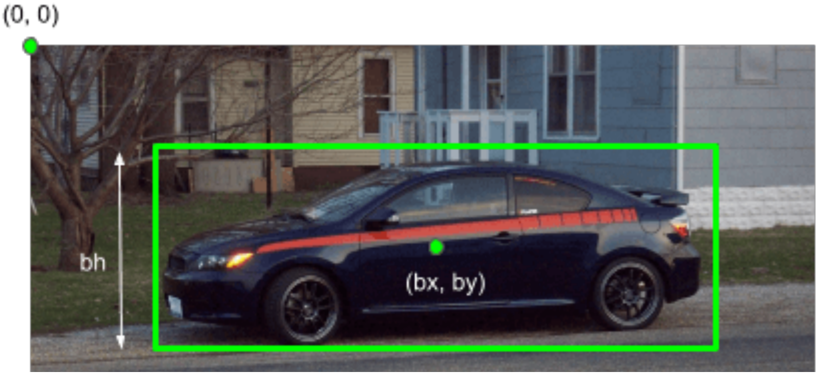
D) Panoptic Segmentation: It is a combination of Instance and Semantic Segmentation in a way that we associate with each pixel two values: Its class label and a instance number. It also recognizes the sky, the road, and other background elements collectively known as stuff.



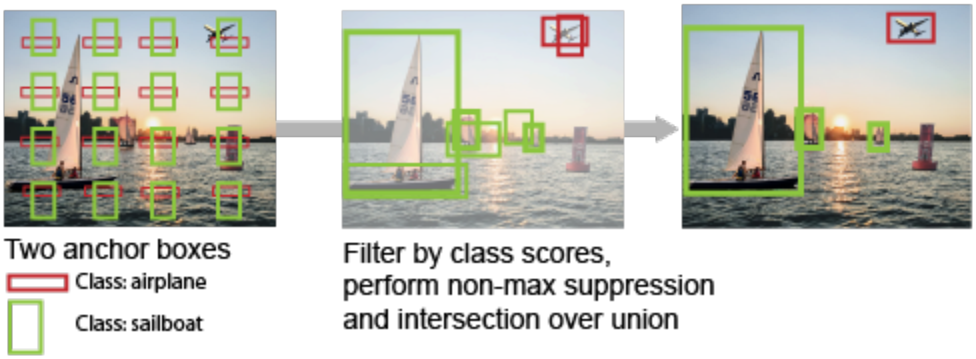
Important Concepts :

1. **Bounding Box**: It is a tight rectangle used to enclose the object of interest. It is generally described by four values : (bx, by, bh, bw) .

   Where (bx, by) are the co-ordinates of the center of the box and bh, bw are the height and width of the box respectively measured on a scale of 0 to 1.



2. Anchor Boxes: These are a set of predefined bounding boxes of a certain height and width. These boxes are defined to capture the scale and aspect ratio of specific object classes you want to detect and are typically chosen based on object sizes in your training data-sets. During detection, the predefined anchor boxes are tiled across the image. The network predicts the probability and other attributes, such as background, intersection over union (IoU) and offsets for every tiled anchor box. The predictions are used to refine each individual anchor box. You can define several anchor boxes, each for a different object size.



Thus the network refines these anchor boxes to finally output the tight bounding boxes. These are defined by the scale and aspect ratio.

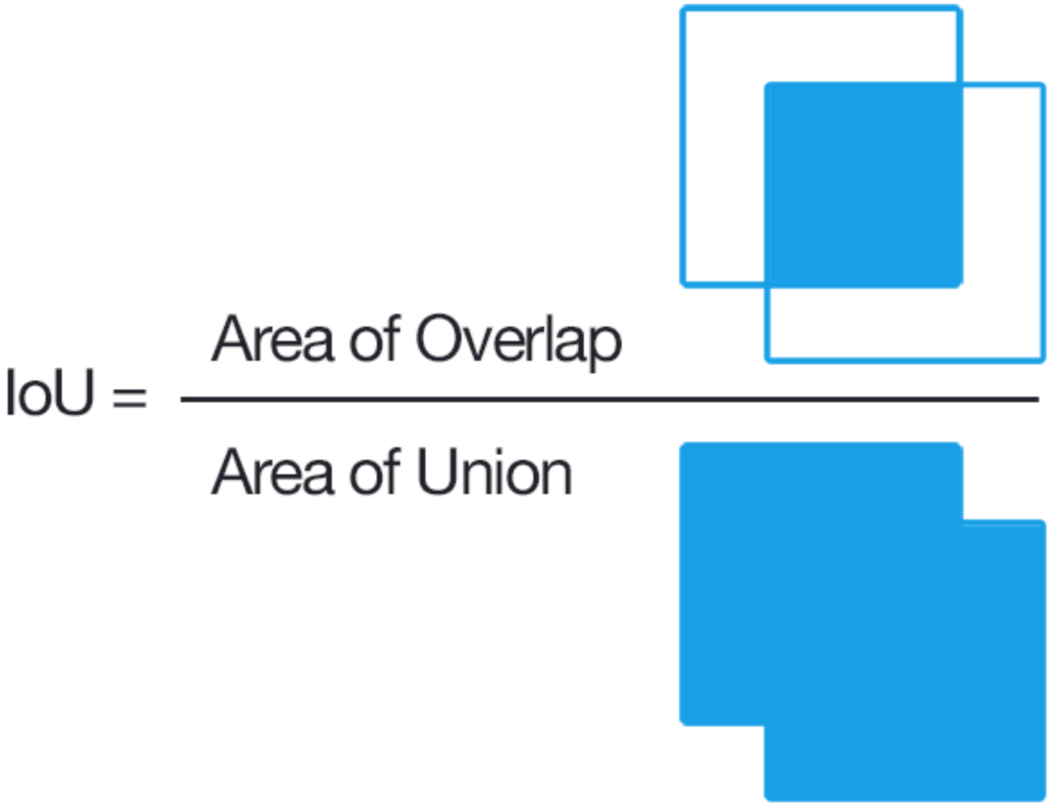
Aspect Ratio is the width / height of the box.

Size is the height and width of the box. eg (256 x 256)

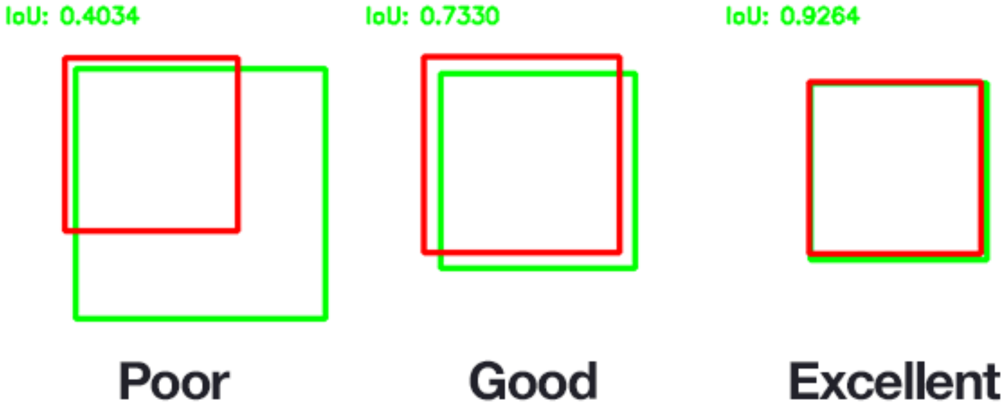
Scale is the multiplying factor of the required box w.r.t to base box

3. Intersection over Union (IOU) :

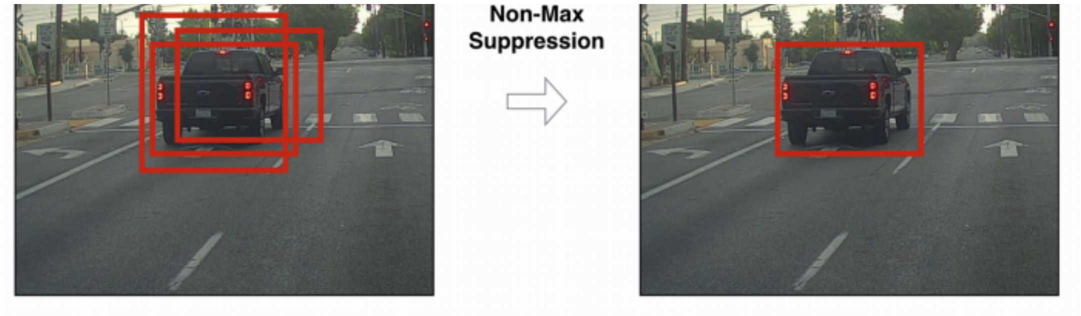
It is an evaluation metric used to check the accuracy of the predicted bounding box w.r.t the actual ground truth.



An IOU of > 0.5 is considered as a good prediction and is used for further evaluation.

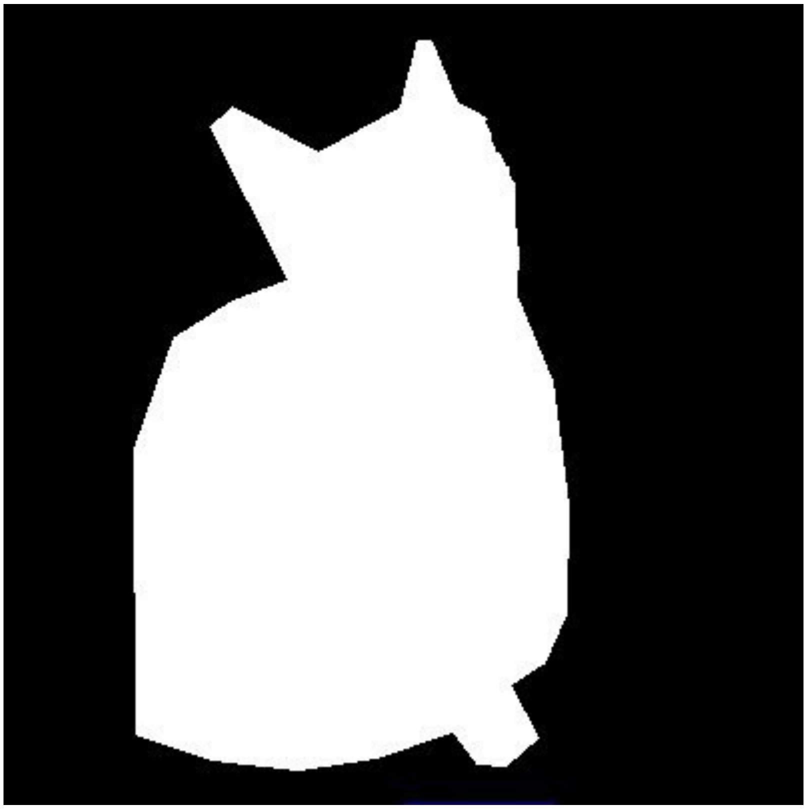


4. Non- max suppression: If multiple boxes are present for a given object then, as the name suggests, this technique discards all boxes except the one having the maximum IOU.



5. Binary Mask: It is a 2D array, that has a data point representing the same pixel width & height of the image.

Each pixel in our mask is labeled either a `1` or `0` (`true` or `false`) for whether or not it belongs to the predicted instance.



Binary Mask for a cat

Metric : Mean Average Precision:

Mean Average Precision or mAP is the metric used to quantify the accuracy of object detectors.

Firstly,

$$precision = \frac{true\ positives}{true\ positives + false\ positives} = \frac{terrorists\ correctly\ identified}{terrorists\ correctly\ identified + individuals\ incorrectly\ labeled\ as\ terroists}$$

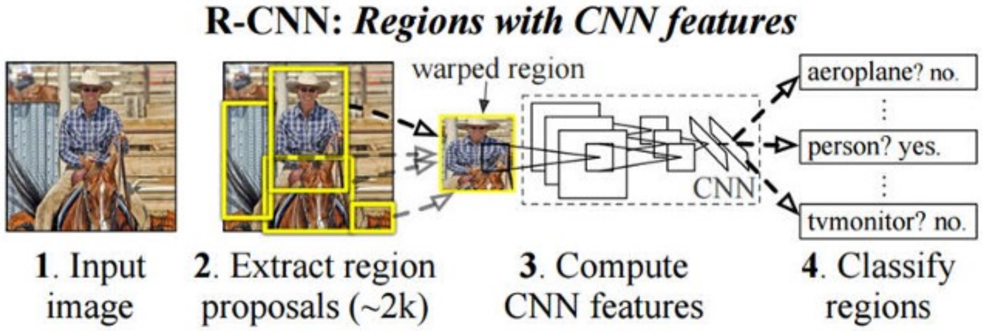
Average Precision for a image means precision averaged over all instances of objects present in the image.

mAP is the average precision averaged over IOU of 0.5 to 0.95 with a step size of 0.05.

As a convention, mAP is expressed as a percent value.

REGION PROPOSALS:

A) RCNN: The RCNN is a region proposal based object detection algorithm. Its stands for region based convolutional neural network.



R-CNN workflow

Steps involved :

1) Segmentation:

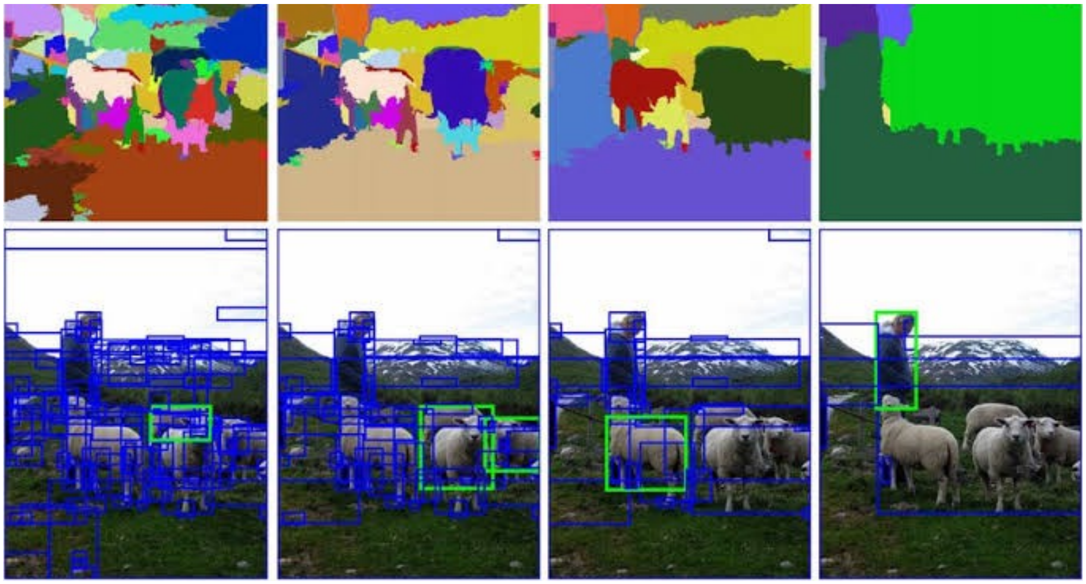
The original rcnn paper [1] by Girshick et. al. Uses the Selective Search method for generating around 2000 region proposals.

1.1) **Selective Search**:

Selective Search uses a Hierarchical Grouping Algorithm to generate the region proposals.

1.1.1) **Generating Initial Regions**:

It first runs a graph based image segmentation algorithm to obtain the initial regions as seen in the leftmost column of the image below.



1.1.2) Similarity Measure:

We find the similarity between regions based on the following criteria:

1. Color

2. Texture

3 .Size

4. Shape Compatibility

A Similarity Metric is obtained as follows:

s(ri,rj) =a1Scolour(ri,rj) +a2Stexture(ri,rj) +a3Ssize(ri,rj)+a4Sfill(ri,rj)

1.3) Recursive Grouping:

Starting from these initial regions we recursively group these regions based on the similarity metric. We stop once the required number or proposals is attained.

1.2) **Warping**: Each of the region proposal is resized(scaled) to the required input size of the Convnet and enclosed in a tight box.

1.3) **Feature Extraction**: Each of these warped region is fed one y one to a Convnet which outputs a 4096 length feature vector.

1.4) **Classification**: The 4096 length feature vector is then fed to a

SVM which classifies whether a object is present and assigns a label to it.

1.5) **Bounding Box Regressor**: In addition to the class label the rcnn uses a linear regressor which outputs the bounding box co-ordinates for the object.

6) **IOU and non-max suppression**: In case of overlaps the best scored region is chosen and the rest are discarded.

B) Fast RCNN:

It is an improvised version of the rcnn as it eliminates some of the shortcomings of the rcnn.

Advantages:

1. Higher detection quality (mAP) than R-CNN, SPPnet

2. Time of Computation is reduced as it is a single stage process.

3. Does not require any extra disk storage to cache intermediate features.

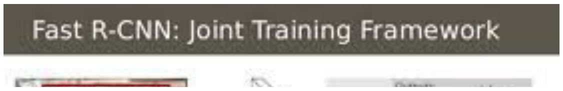
4. Lesser parameters as compared to rcnn and SPPnet.

Process:

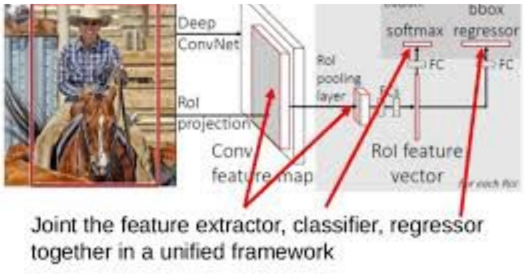
1) Feature Map Generation: Entire image is fed in along with object proposals to a Convnet. On passing through the Conv layers and Max Pooling layers a feature map is obtained.

2)ROI Pooling: The Region of Interest (ROI) in the feature map is given y (r,c,h,w) co-ordinates. This ROI is then passed through a ROI pooling layer to get a H x W feature map.

3. Fully Connected Layers: This feature map is then extracted to a FC layer and then passed through FC layers to a softmax for class probability prediction and a regressor for the bounding box regression outputs.



Fast R-CNN: Joint Training Framework



C) Faster RCNN:

Faster RCNN model was proposed by Ross Girshick et. al. [3] as a computationally efficient solution to object detection.

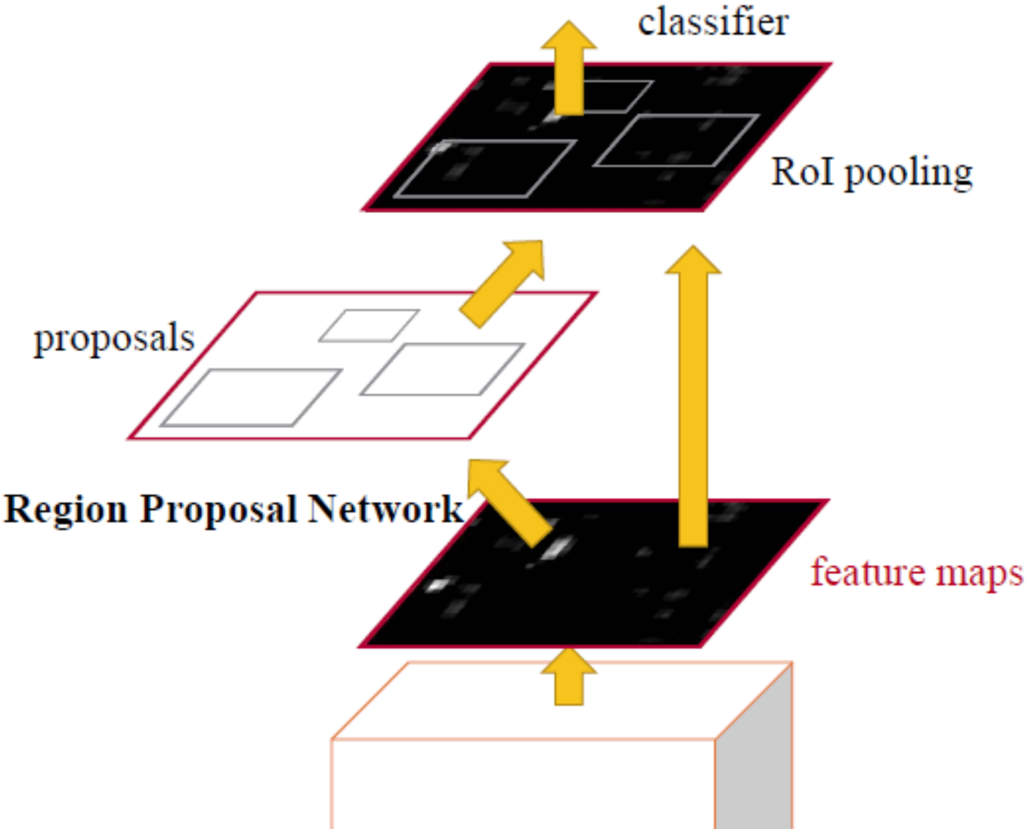
Merits over Fast RCNN:

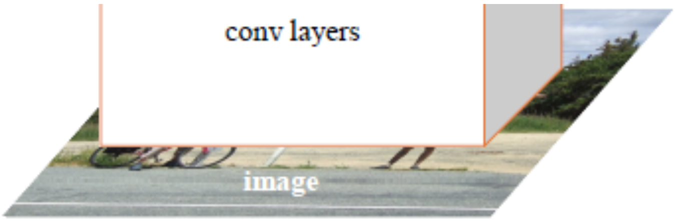
1. It eliminates the computational bottleneck of determining region proposals from a image.

2. It uses a Fully Convolutional neural network for this purpose which makes it a single flow pipeline.

3. The RPN introduced in this paper[3] shares the features with the object detector as well.

Architecture and Operation:





1) Feature Map Generation: The image is passed through Conv layers which output a feature map.

2)Region Proposal Network: A sliding window is used in RPN for each location over the feature map.

3) Anchors: For each location, k (k=9) anchor boxes are used (3 scales of 128, 256 and 512, and 3 aspect ratios of 1:1, 1:2, 2:1) for generating region proposals.

4) Classification : A cls layer outputs *2k* scores whether there is object or not for *k* boxes.

5) Regression: A reg layer outputs *4k* for the coordinates (box center coordinates, width and height) of *k* boxes.

6) Detection network: Except for the RPN part the Detection network is the same as that of Fast rcnn.

7)Alternate Training: The RPN and Detection part are trained alternately so that they share the features learnt by each other.

D) Mask RCNN:

Mask RCNN extends Faster Rcnn by adding a parallel mask output branch. It is a very important method used in instance segmentation.

Motivation:

1. Faster Rcnn, Yolo and other object detection algorithms output a bounding box and a class probability label associated with that box.

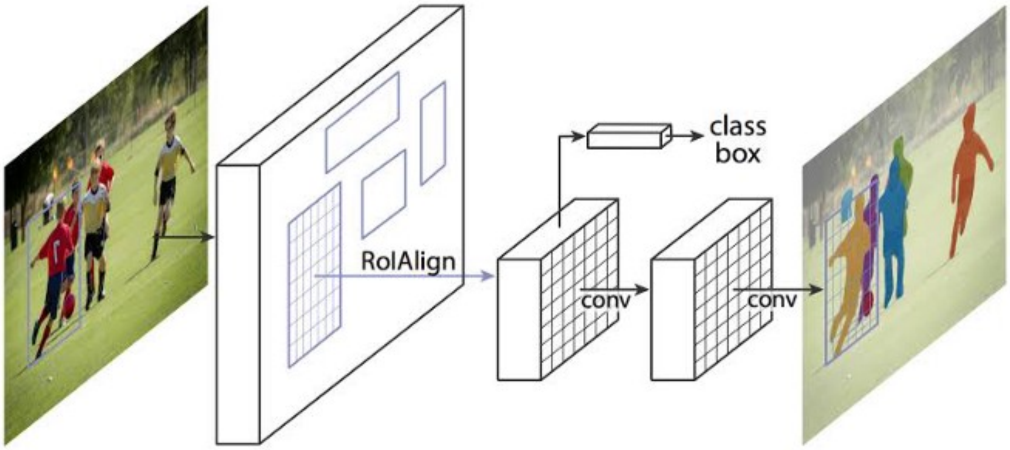
2. We as humans do not locate real life objects by drawing boxes around them, instead we look at the outline and the pose of the object in order to detect it.

3. In this regard, mask rcnn gets closer to human style of object perception.

4. The research on mask rcnn motivates us further leading to areas of panoptic segmentation, person keypoint detection, sports pose estimation etc.

5. All the self driving cars use the fundamental concept behind mask rcnn.

Architecture and Implementation:



1. Mask R-CNN adopts the same two-stage procedure, with an identical first stage (which is RPN).

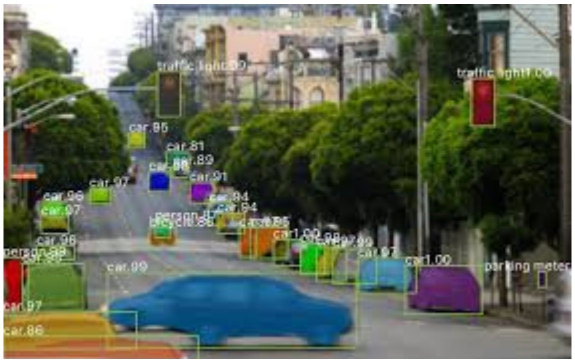
2. In the second stage,in parallel to predicting the class and box offset, Mask R-CNN also outputs a binary mask for each RoI.

ROI Align Layer:

i. The ROI pool layer in Faster Rcnn performs quantizations like flooring the floating point values and aggregation functions like Maxpool.

ii. Such operations result into coarse features destroying the finer pixel to pixel arrangements which are necessary for instance segmentation.

iii. To Counter this, Mask Rcnn uses the ROI Align layer which uses bi-linear interpolation instead of quantization which preserves the pixel alignments and improves mask accuracy.



Current Research and Future Scope:

Panoptic Segmentation: Recent CVPR papers make use of the mask rcnn model and build on top of it to achieve state of the art results on popular data-sets like City-Scapes.

Mesh Rcnn: It is a very accurate system proposed by Georgia Gkioxari et. al. used for 3-D shape prediction which augments the mask rcnn models with a mesh prediction branch to generate voxel representations.

References :

1. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014
2. R. Girshick. Fast R-CNN. In ICCV, 2015
3. Faster R-CNN: To-wards real-time object detection with region proposal net-works. In NIPS, 2015
4. Kaiming He, Georgia Gkioxari, Piotr Doll´ar, and Ross Girshick. Mask R-CNN. In ICCV, 2017

5. Image References: Google

Connect With Me on ***Linkedin*** !!

Thank you !!!