Object Detection

CV Project - Team “Kuch bhi”

Team Members:
2018101033 - Jay Sharma
2018102021 - Tanmay Garg
2018102040 - Shantanu Agrawal
2020900019 - Anirudh Polatpally

Method - Overview

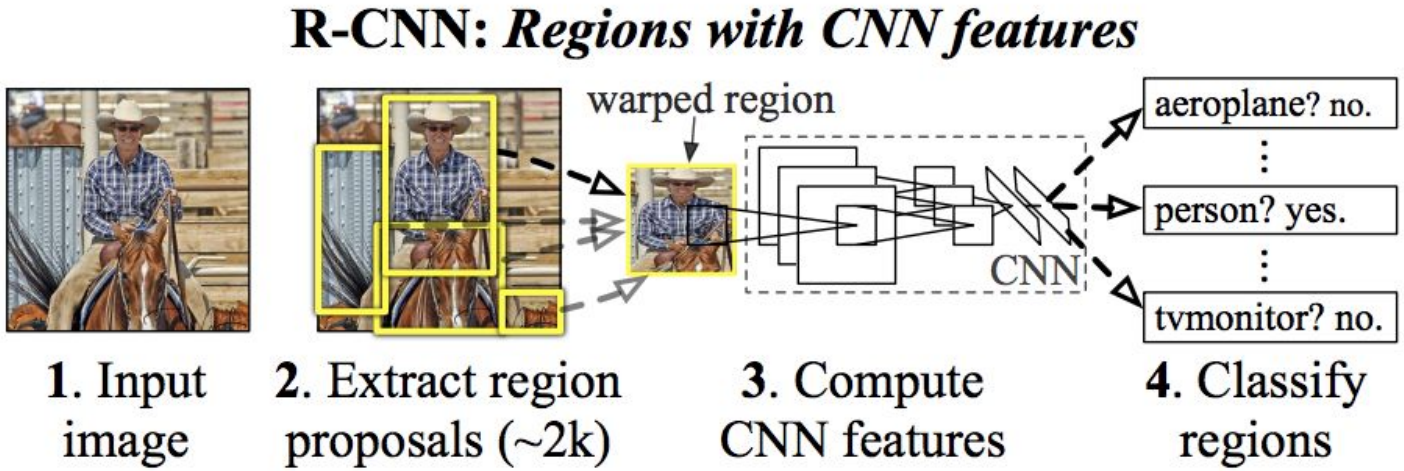


Fig 1: RCNN workflow
Source: [1]

Region Proposals

1. Used Selective Search Algorithm
2. Generated a small set of region proposals from 10 images
3. Considered regions with IoU greater than 0.7
4. Obtained 401 proposals
5. Proposal dimensions - 224, 224, 3

Fig 2: Selective Search at different scales
Source: [2]

Feature Extraction - CNN

- Used VGG-16 as backbone feature extractor.
- *3×3* convolutional layers stacked on top of each other
- Maxpooling to reduce volume in between
- Fully Connected layers at the end followed by softmax.
- Final layers changed to perform classification and localisation.

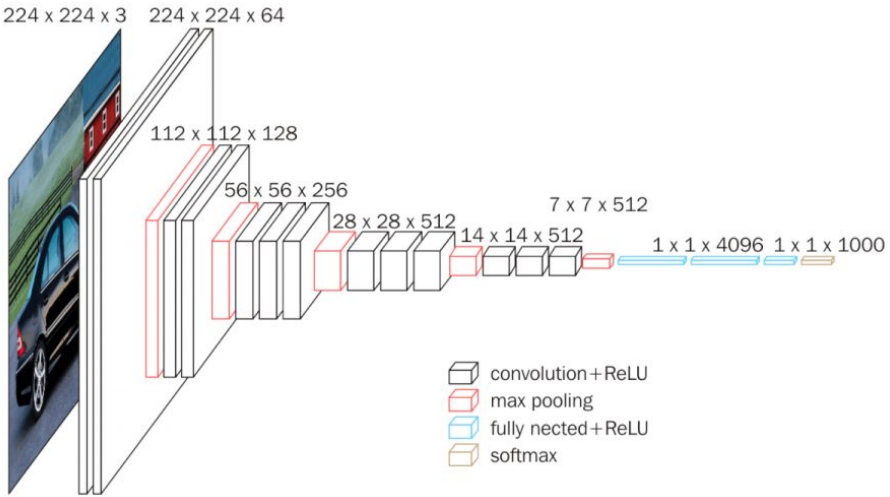


Fig 3: VGG16, Source: [3]

Classification Head

- Perform Classification
- Final output - scores for each of the classes in consideration (21 in our case)

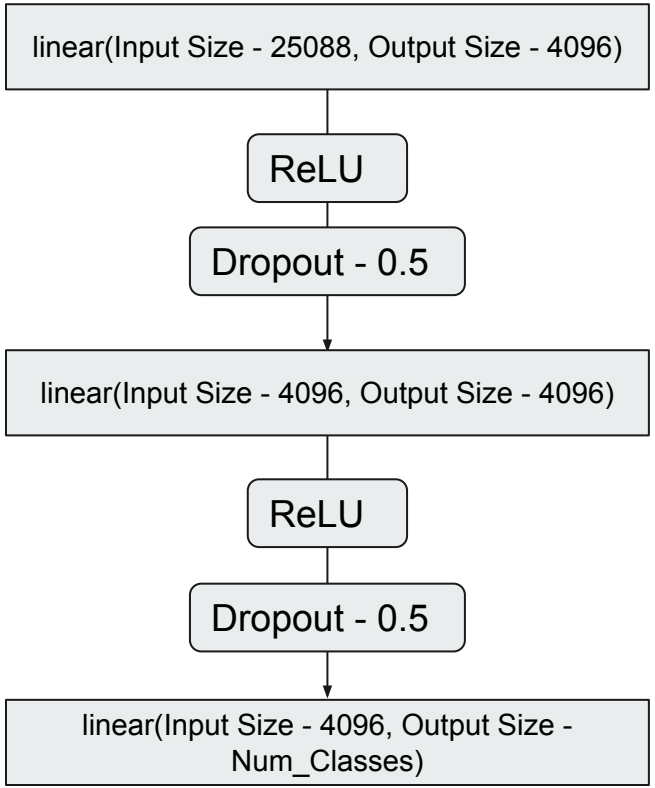


Fig 4: Classification Head

Regression Head

- Perform Bounding Box Regression
- Final output - Predicted Bounding Box (4 values)

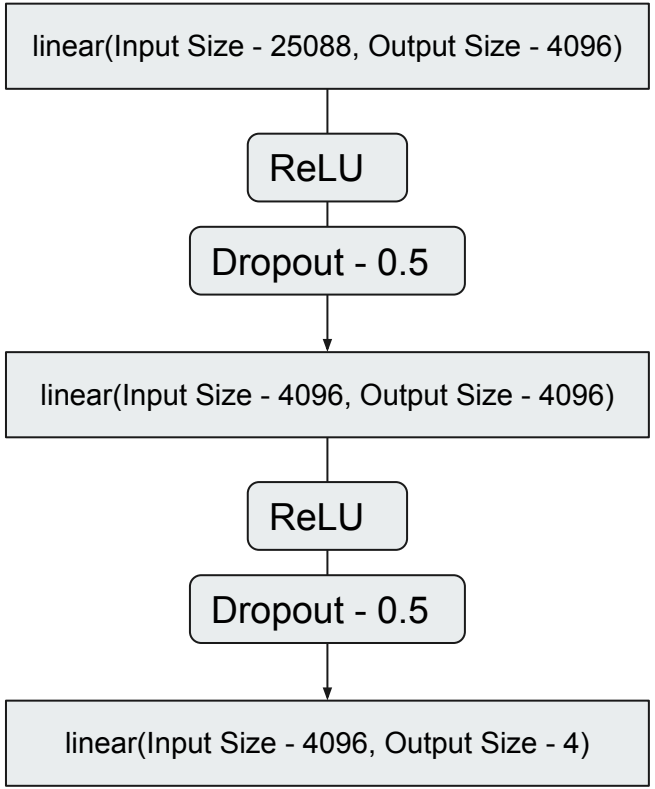


Fig 5: Regression Head

Training

- Trained on the proposals generated using selective search
- Initially trained classification head, then followed by regression head.

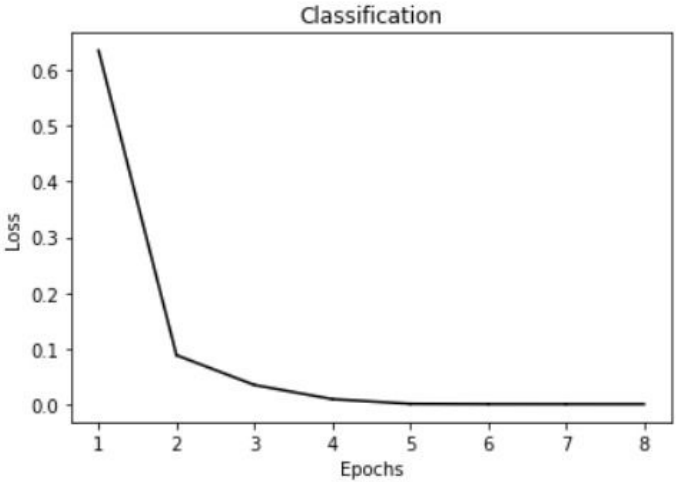
Classification Scores-

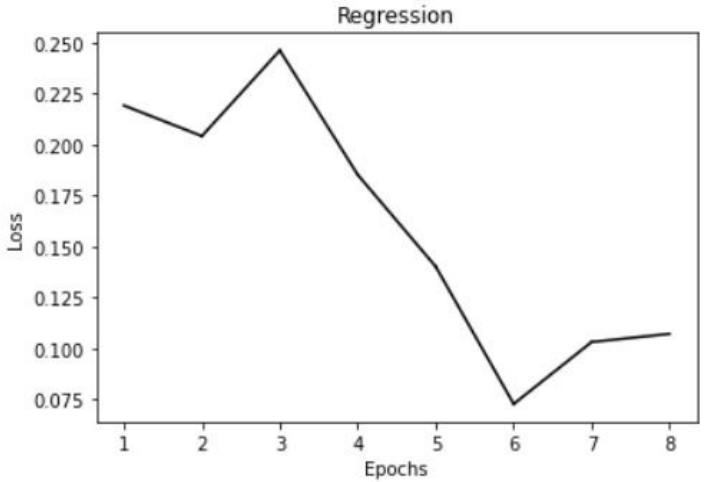
Number of epochs	8
Final loss	5.6e-5
Final acc	1

Regression Scores-

Number of epochs	8
Final loss	0.107
Final acc	-

Losses


Classification
Loss
Epochs
0.0
0.1
0.2
0.3
0.4
0.5
0.6
1
2
3
4
5
6
7
8

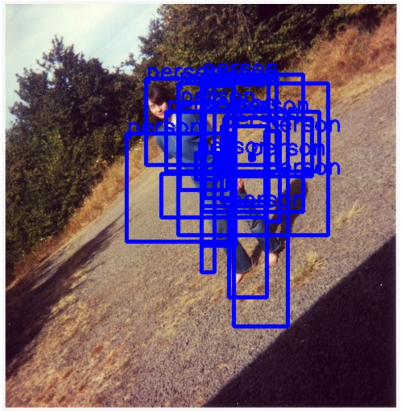

Regression
Loss
Epochs
0.075
0.100
0.125
0.150
0.175
0.200
0.225
0.250
1
2
3
4
5
6
7
8

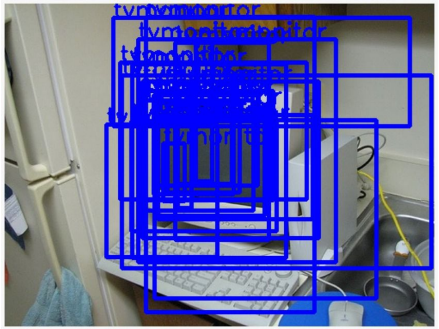
Non Maximum Suppression

- Remove multiple bounding boxes
- Remove boxes with low confidence
- Select a box with highest confidence
- Removes lower scoring boxes which have an IoU greater than iou_threshold with the highest scoring box

Tests

Tested the trained model with some of the proposal images





Up Next

- Train the model for object detection on the entire dataset from scratch
- Perform Ablation tests
- Fine tune the model
- Report Scores

Faster RCNN Experiment

The RCNN training time for any significantly big dataset (>1000 Image) is too high, and possibly infeasible with limited GPU resources, and lack of our understanding to compute selective search on GPUs.

Since the only thing different in faster RCNN is the RPN, we tried building it. We accomplished implementing various sized anchors, but could not complete RPN due to issues with our understanding, and the complexities of vectorizing everything to make training actually fast. Apart from this, the multi-stage non-end-to-end training regime was also seeming to be time-taking and complex, so we left it and got back to implementing RCNN.

If the training and prediction time for RCNN is remains too slow (which we suspect is most certain), and if time permits, we would try implementing Faster-RCNN once again, this time with a clearer base understanding of the architecture.

Reference(s)

1. Rich feature hierarchies for accurate object detection and semantic segmentation - Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik (CVPR 2014) - Link
2. Selective Search for Object Recognition - Uijlings, Jasper & Sande, K. & Gevers, T. & Smeulders, A.W.M. (IJCV 2013) - Link
3. Very Deep Convolutional Networks for Large-Scale Image Recognition - Karen Simonyan, Andrew Zisserman (ICLR 2014) - Link

The End