Object Detection

CS698N Final Project Presentation

AKSHAT AGARWAL

SIDDHARTH TANWAR

Problem Description

- Arguably the most important part of perception
- Long term goals for object recognition:
  - Generalization from a few examples
  - Robust illumination, deformation, scale invariance
- The 100/100 tracking challenge, discussed by Dieter Fox in ICRA'16 looking to identify and track 100% of the objects and activities in a scene, with 100% accuracy
- Learn semantic info about objects, like how to interact with them, observe and learn their behavior itself

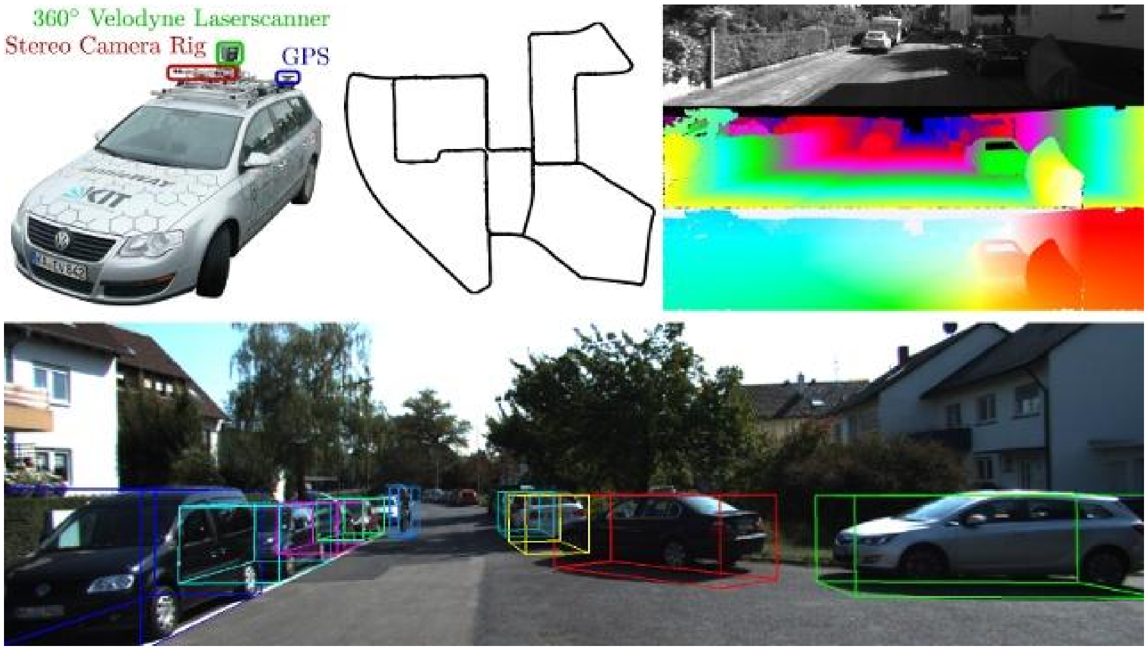
Dataset - PASCAL VOC

- PASCAL Visual Object Classes (VOC) challenge
- Contains a number of visual object classes in realistic scenes
- Contains 20 classes, ~10k images with ~24k annotated objects
- Annotation contains both object class and bounding box
- Released in 2007
- Evaluation: Average Precision over the entire range of recall, with a good score having both high precision and high recall

Everingham, Mark, et al. "The pascal visual object classes (voc) challenge."*International journal of computer vision* 88.2 (2010): 303-338.

Dataset - KITTI

- Geared towards autonomous driving
- 15k images, 80k labeled objects
- Provides ground truth data with LIDAR
- Dense images of an urban city with up to 15 cars and 30 pedestrians visible in one image
- 3 classes: Cars, Pedestrians and Cyclists



Geiger, Andreas, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? the kitti vision benchmark suite." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

Related Work

- DPM: Uses HOG features at multiple scales, filters and then classifies (33.4)
- Selective Search: Bottom up segmentation by merging similar regions (35)
- R-CNN: Extracts features from region proposals, then classifies (59.2)
- Fast R-CNN: Introduced an RoI pooling layer which creates fixed size feature maps for each RoI and then regresses (70.7 (w YOLO))
- Faster R-CNN: Introduced a Region Proposal Network (RPN) which generates proposals using CNN features shared with the image (75.9)
- SSD: Uses default bounding boxes over different aspect ratios and scales for conv. feature maps at multiple resolutions, no proposals (75.8)
- PVANET: Redesign feature extraction part, fewer channels with more layers. Uses concatenated ReLU, inception modules and combines multi-scale intermediate outputs (82.5)
- R-FCN: Fully convolutional network, learns position sensitive score maps, gets region proposals from an RPN and applies RoI pooling
- **Current Leader - R-FCN + ResNet ensemble trained on VOC+COCO (mAP 88.4)**

Problems

- Most of these methods have :
    - Large, complex detection pipeline
    - Independently precisely tuned stages
    - Slow forward pass
- Selective Search takes 2 secs/image to generate proposals!
- R-CNN takes 40 secs/image at test time!
- Faster R-CNN runs at 5FPS
- R-FCN takes 170 ms/image (~6FPS)

Dai, Jifeng, et al. "R-FCN: Object Detection via Region-based Fully Convolutional Networks." *arXiv preprint arXiv:1605.06409* (2016).
Ren, Shaoqing, et al. "Faster R-CNN: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems*. 2015.
Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.
Uijlings, Jasper RR, et al. "Selective search for object recognition."*International journal of computer vision* 104.2 (2013): 154-171.
Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *arXiv preprint arXiv:1506.02640* (2015).

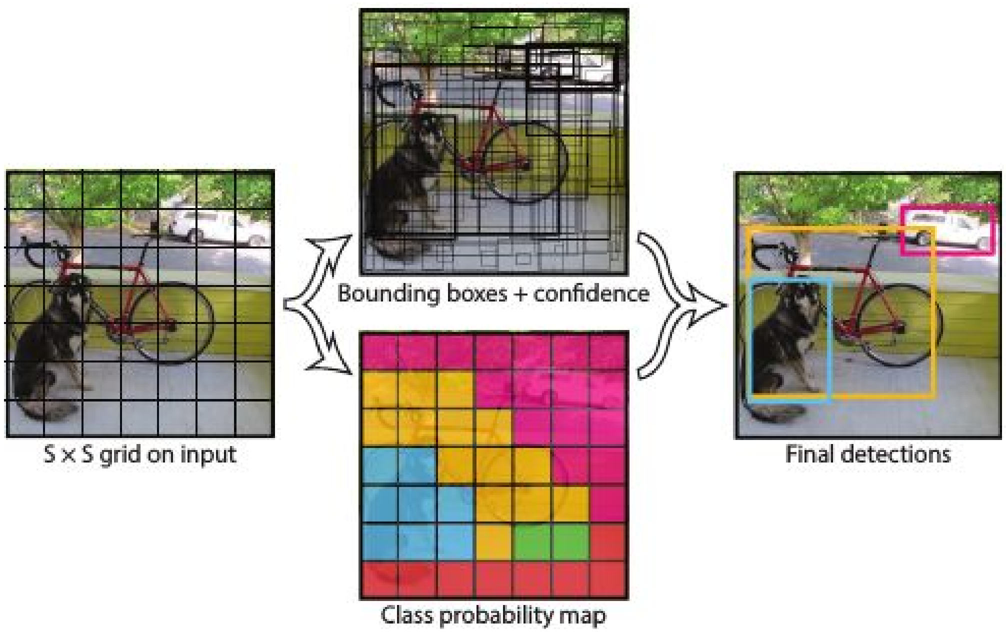
YOLO - Method Overview

YOLO replaces the pipeline with a **single convolutional neural network** which :

- Trains features in-line and optimizes them for detection
- Predicts bounding boxes
- Performs non-maximal suppression
- Predicts class probabilities for each bounding box

Only a single network evaluation is required to predict which objects are present and where they are.

Unified Detection



S x S grid cells

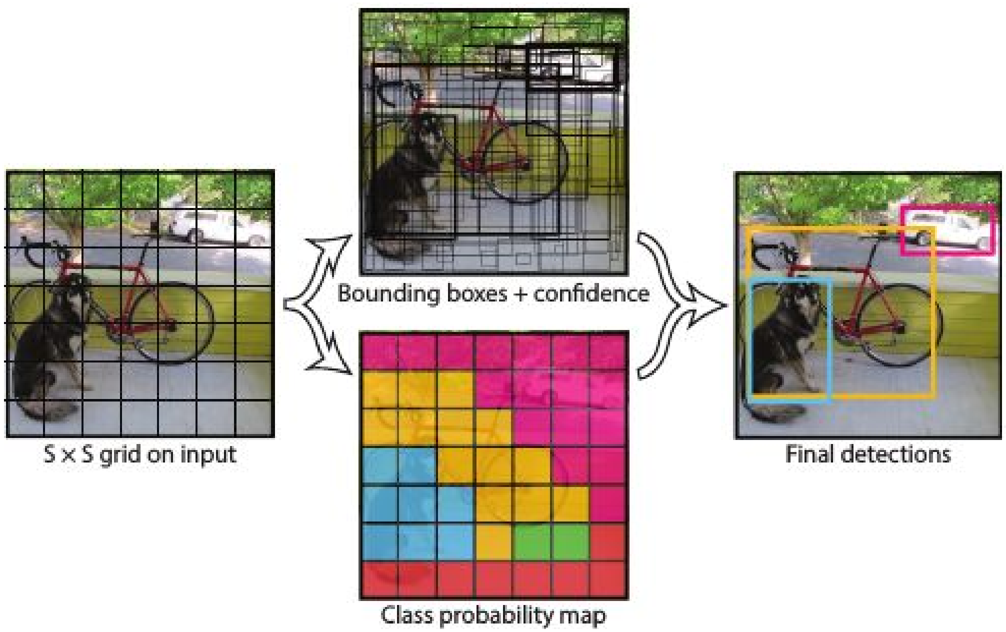
B bounding boxes per cell

The prediction is encoded as a S x S x (B * 5 + C) tensor

For evaluating on PASCAL VOC, S = 7 , B = 2 and C =20 is used.

Image Source : Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *arXiv preprint arXiv:1506.02640* (2015).

Unified Detection



- Each grid cell also predicts C conditional probabilities :

$$Pr(class_i \mid object)$$

Conditioned on the cell containing an object

$$Pr(Class_i|Object) * \; Pr(Object) * {}^{truth}IOU_{pred} = Pr(Class_i) * {}^{truth}IOU_{pred}$$

Image Source : Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *arXiv preprint arXiv:1506.02640* (2015).

Our Network - Scaled version of YOLO

- 8 convolutional layers alternating with maxpool layers, followed by 1 FCL
- Has been pre-trained on the ImageNet classification task
- The fully connected layer outputs a S*S*(B*5+C) vector, where S is the number of grid cells, B is the number of bounding boxes per cell and C is the number of classes (20, for the VOC dataset)
- We used batch sizes of 32, 64 and 128 to optimize usage of GPUs in the space available
- Dropout of 0.5 to prevent co-adaptation
- Network was trained for roughly 250 epochs
- Random scaling and translations of 20% of original image size introduced for data augmentation
- Leaky ReLU used as activation function for conv layers

Results on VOC
Comparison with reported numbers

- Since VOC evaluation is not open, we could not use the VOC2012 validation set during training and instead tested on it.
- We have ~10.7k training images, and ~5.8k testing images
- Author has used VOC2012 validation set for training, his results are with ~16.5k training images and ~11k testing images

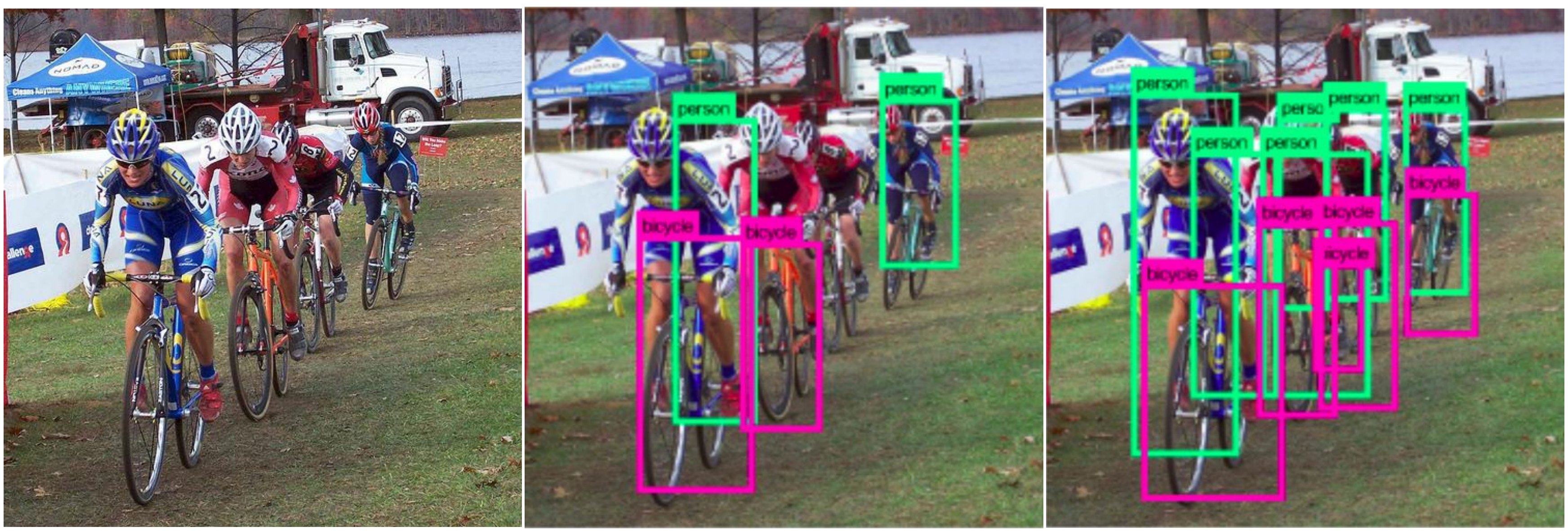
Network pre-trained on ImageNet(mAP)	Network trained by us on VOC(mAP)	Reported by author(mAP)
0.27	44.65	**52.7**

Source: http://host.robots.ox.ac.uk:8080/pascal/VOC/voc2007/dbstats.html

Results
Parameter Study

- Analysis was done varying two parameters :
  - Number of grid cells
  - Number of Bounding Boxes per cell

This was done to improve YOLO’s performance in cluttered scenes



a) Original Image
4 bicyclists

b) Detections with s=7
2 people and 2 bicycles

c) Detections with s=9
5 people and 5 bicycles

Results

Parameter Study - Quantitative Analysis

Class	**S = 5 (mAP)**	**S = 7 (mAP)**	**S= 9 (mAP)**
Aeroplane	58.69	61.70	**67.10**
Bus	64.17	61.74	**67.11**
Cat	61.57	59.10	**70.09**
Train	59.4	58.44	**63.44**
Person	47.88	48.84	**55.19**
Overall	38.86	36.92	**44.65**

- Results obtained from varying grid size S, with B kept fixed at 2
- S = 9 heavily outperforms S = 7 in all object classes
- S = 5 outperforms S = 7 in 'big' object classes - Because finer grid not required?

Results

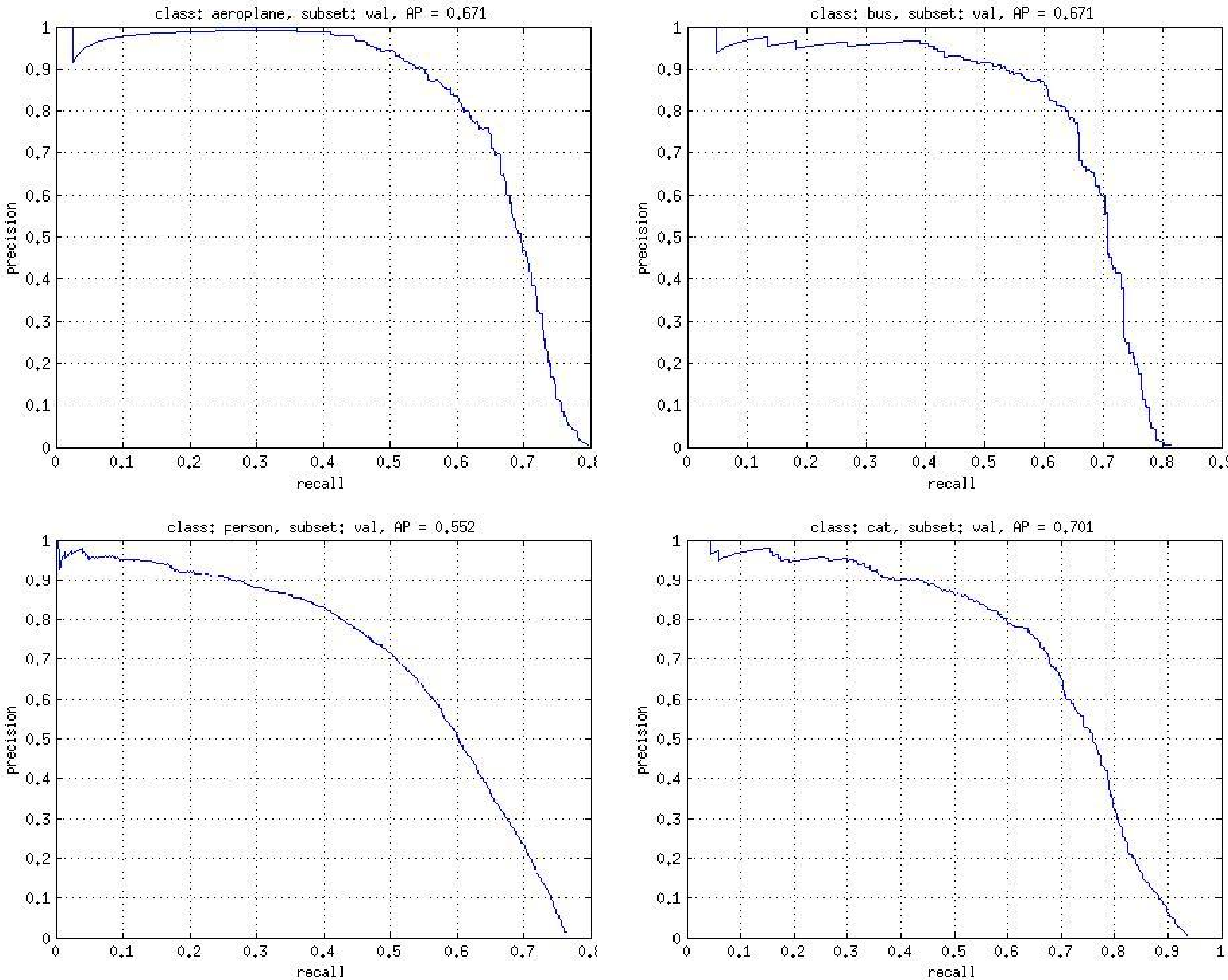
Parameter Study - Quantitative Analysis

Class	**B = 2(mAP)**	**B = 4 (mAP)**	**B = 6 (mAP)**
Aeroplane	**67.10**	60.92	56.53
Bus	**67.11**	65.09	55.01
Cat	**70.09**	59.6	50.07
Train	**63.44**	61.14	52.35
Person	**55.19**	50.81	45.77
Overall	**44.65**	38.10	27.38

- Results obtained from varying number of bounding boxes per grid cell B, with S kept fixed at 9
- Surprisingly, B = 2 outperforms B = 4 and 6 by a significant margin. We postulate that this is because greater B means that there is a greater chance of the wrong box being selected from that grid cell.

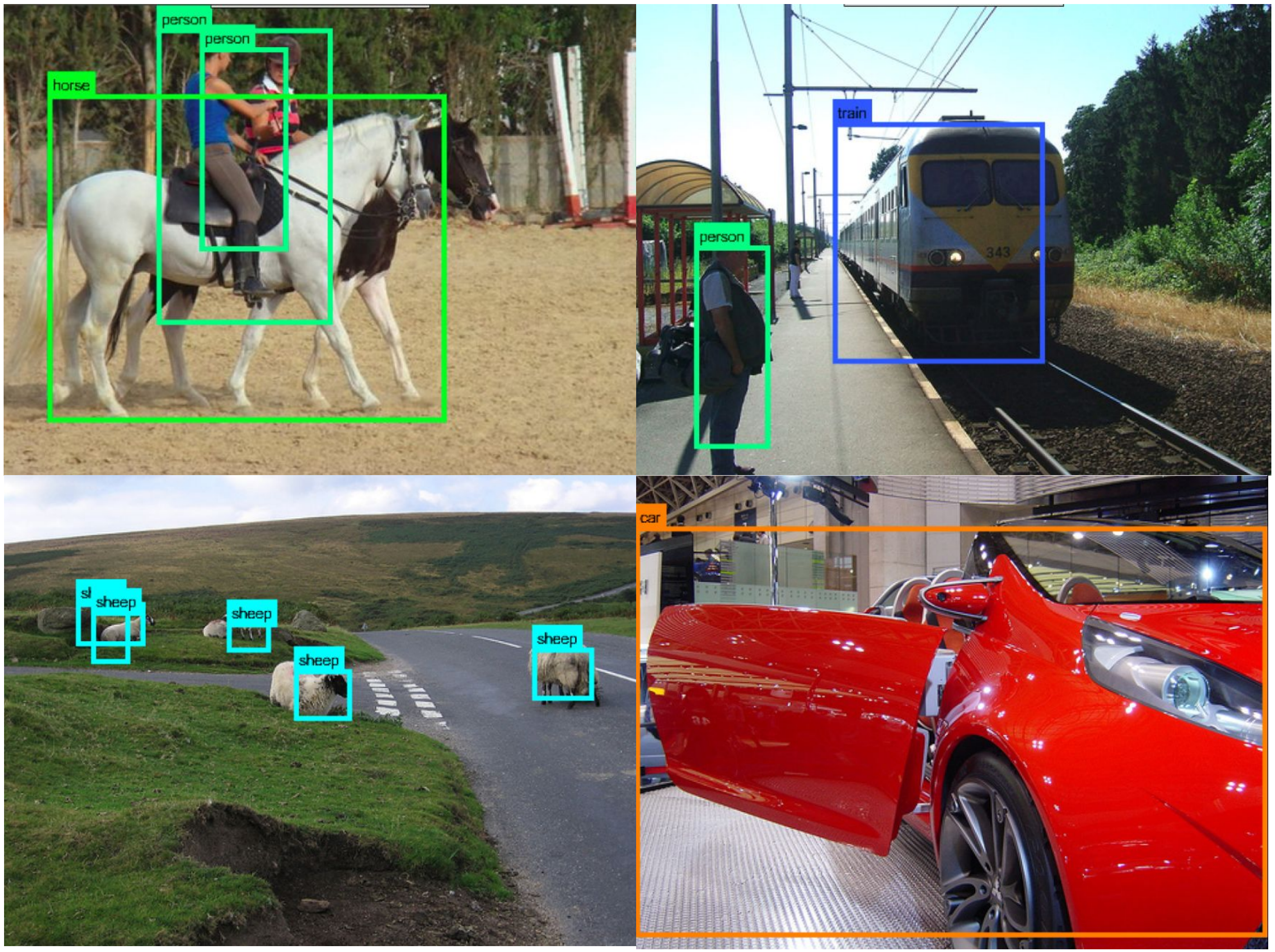
Results

Precision-Recall curves



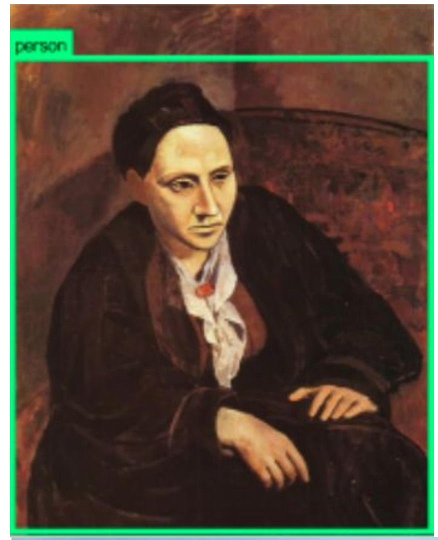
Results

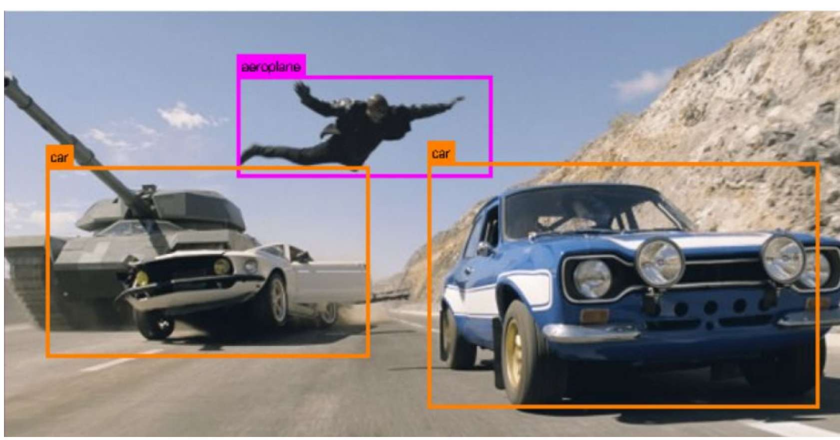
Some examples on VOC dataset

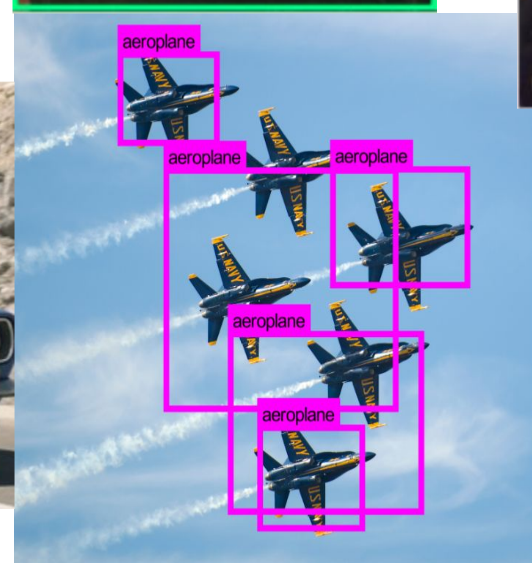


Results

Artwork Corpus & Images in the wild







Results

Examples on KITTI Object Detection task

Results

Videos

- The smaller YOLO model runs at around 50-70 fps on NVIDIA GeForce GTX 760 GPU
- Playing 2 videos, one shows detection using the network trained on VOC and the other using the network trained on KITTI Vision Benchmark suite for object detection.

Our work

- Did an extensive parameter study on YOLO, studying the changes observed on increasing or decreasing the grid resolution, and increasing the number of bounding boxes per grid cell
- Obtained object detection results on the object detection benchmark KITTI by using a YOLO network pre-trained on ImageNet.

Broadly speaking, both of us did most of the work together. However,

- Akshat worked on integrating KITTI data into VOC and Darknet compatible format
- Siddharth worked on the parameter study for YOLO

Thank You