CV Assignment 4

Name 1: Abdelrahman Salem Mohamed
ID 1: 6309
Name 2: Reem Abdelhalim
ID 2: 6114

Object Detection

In this assignment, you will work on Cocco dataset which is a large-scale object detection, segmentation, and captioning dataset. You are required to run 3 different object detectors on this dataset. We will learn to use and differentiate between the architectures.
Either PyTorch or TensorFlow are allowed to run your models.

Dataset

We test models using coco 2017 validation set which consists of 5000 different images with their meta data from segmentation points and object bounding boxes.

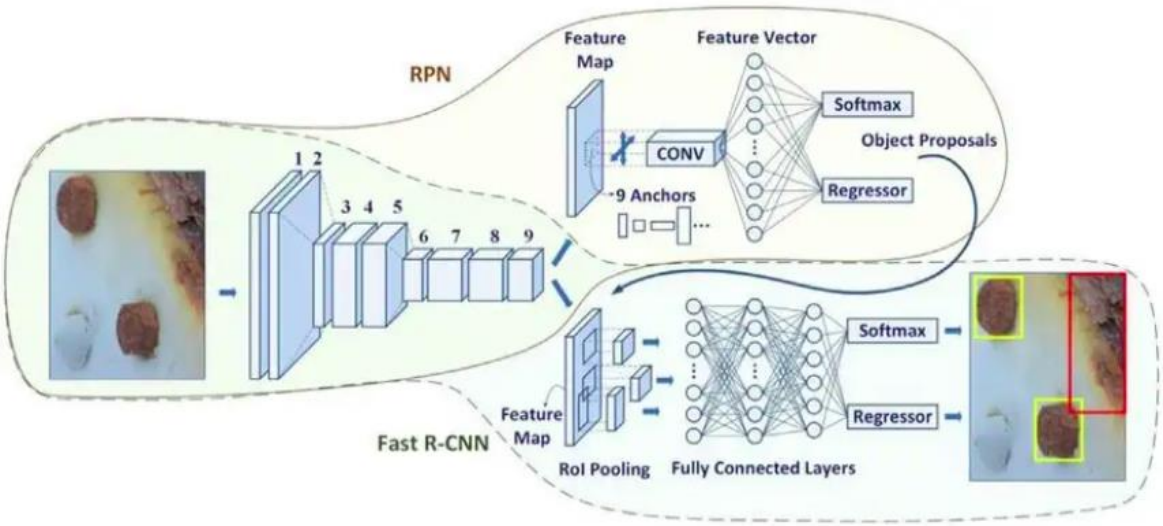
Models

We used three different models:

- Faster R-CNN (Two stage Model)
- RetinaNet One stage Model)
- Single shot detector (One stage Model)

1- Faster R-CNN

Architecture diagram



Discussion

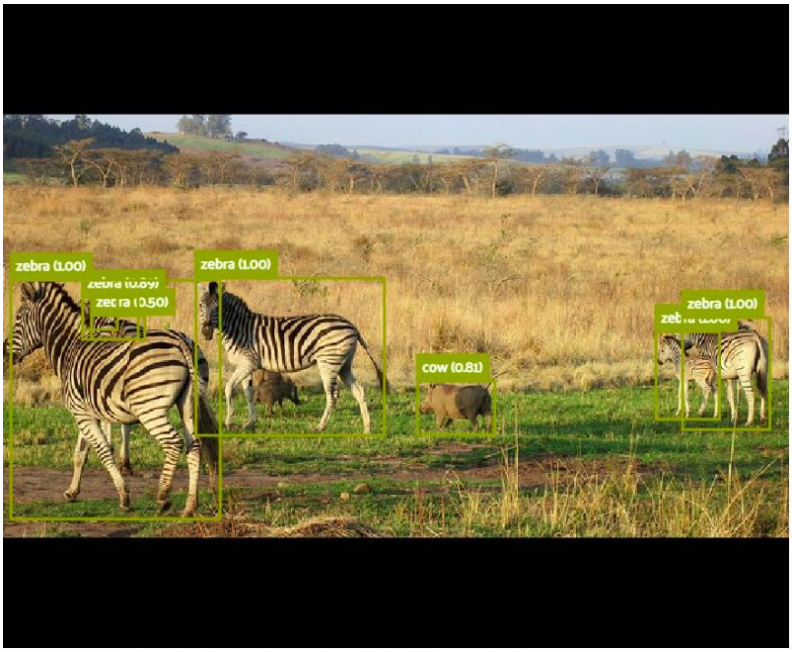
Two stage based model where, first the model learn how to extract Region of proposals, then for every Region of proposal fe perform the normal Fast R-CNN network from ROI

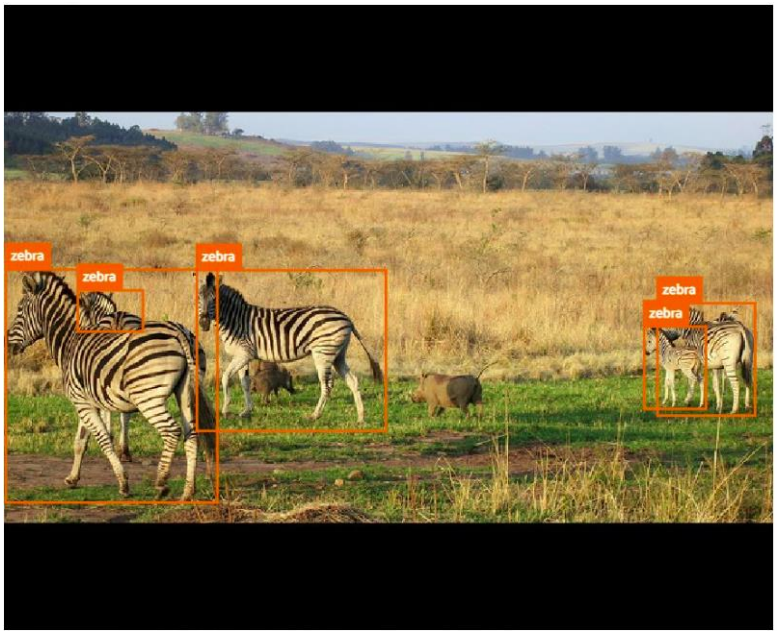
pooling then from fully connected layer compute the two losses one for object box (regression), and one for class type (classification).

Each network has its own loss functions and the whole system weights consider the both losses path, and during the test the network needs first to identify the ROI and apply Object detection technique, so Faster R-CNN consider slower than one stage models but have better mAP.

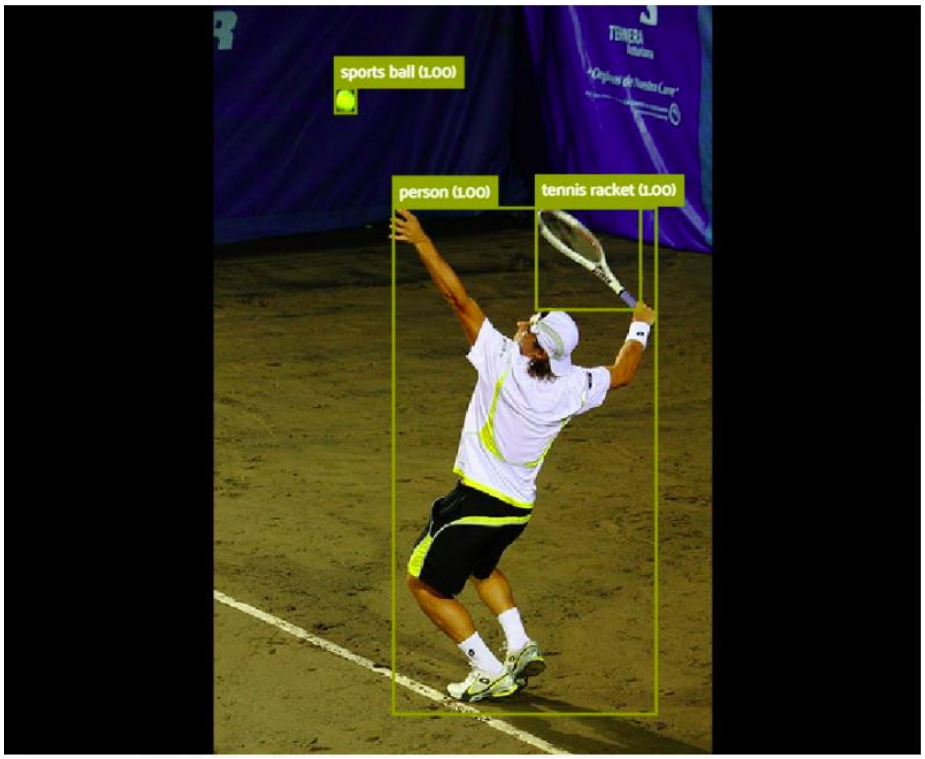
The backbone network is usually a dense convolutional network like ResNet or VGG16

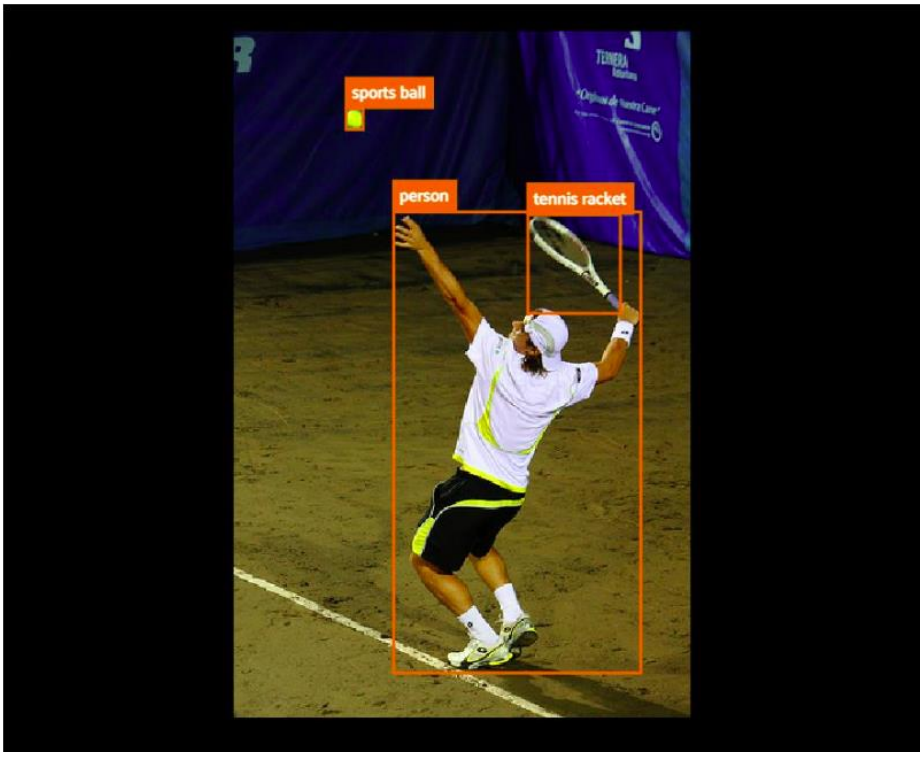
Results on COCO dataset



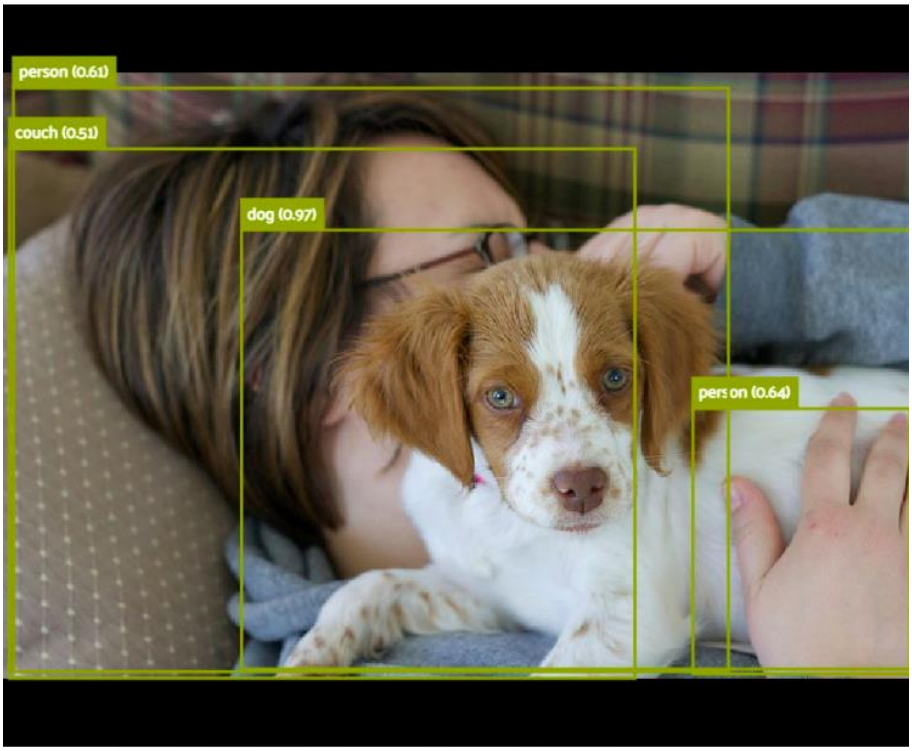


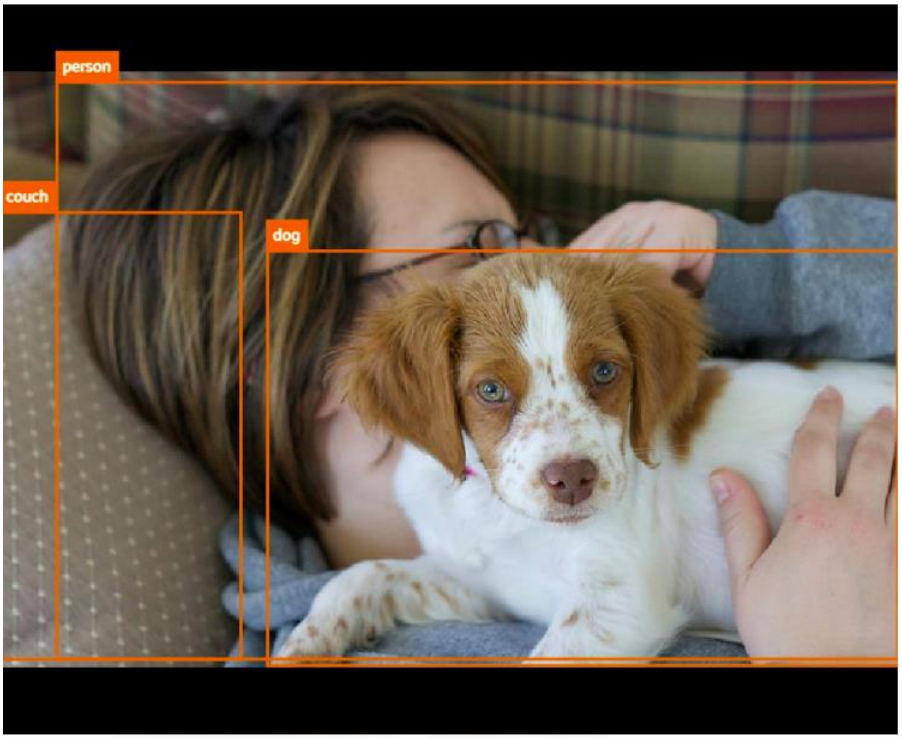
Note: The Model detected all objects and detected not annotated cow/pig



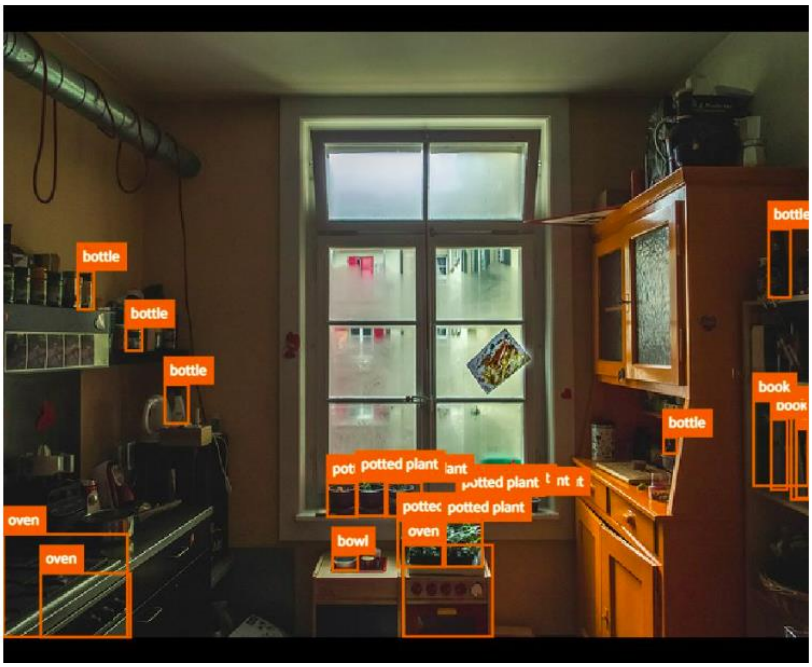


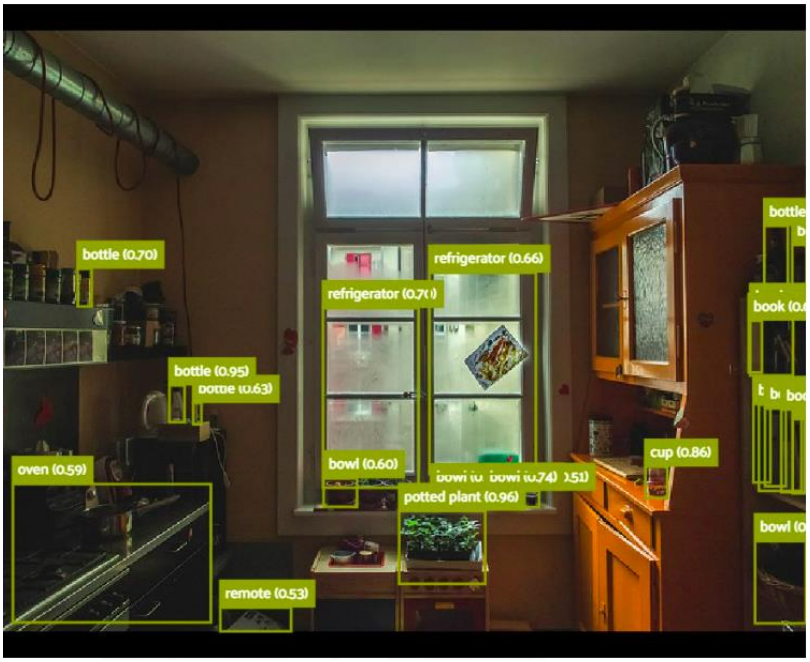
Note: The Model Detected the all object exactly as the GT with 100% confidence





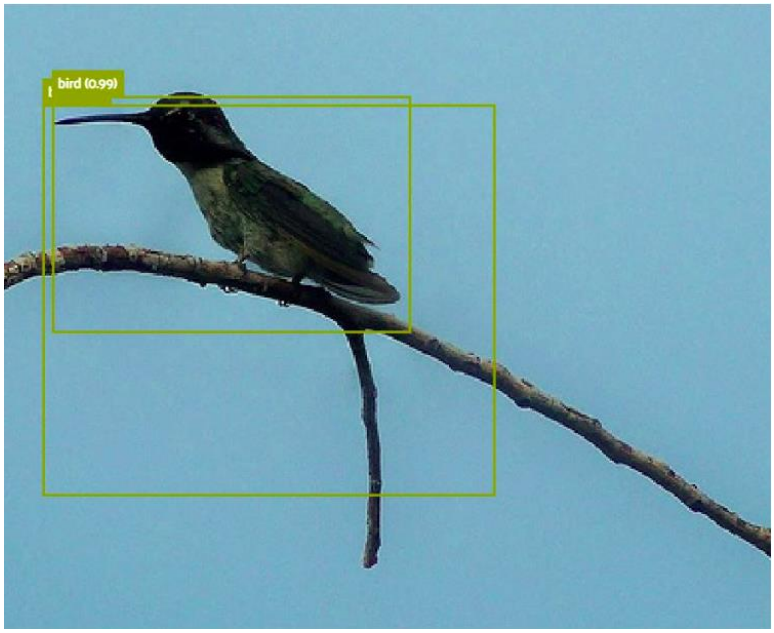
Note: The model detected the hand as human with is not exactly correct but detect the crouch with low confidence and existence of person there

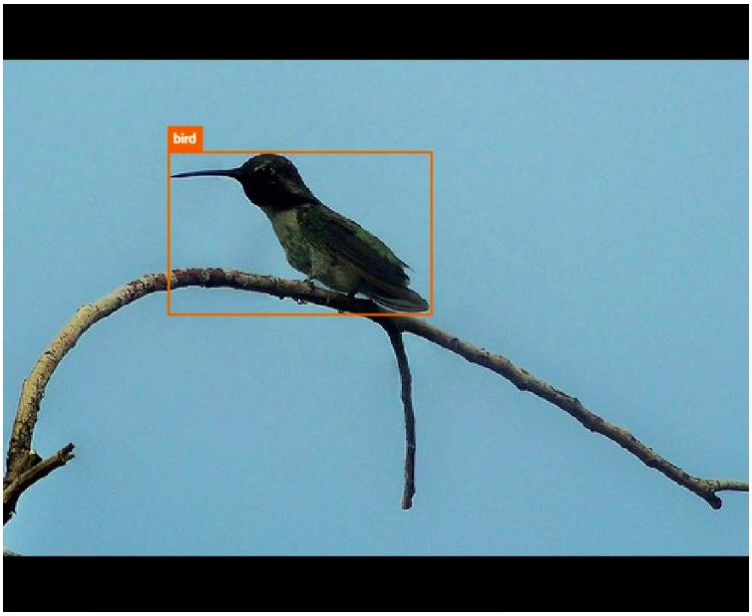


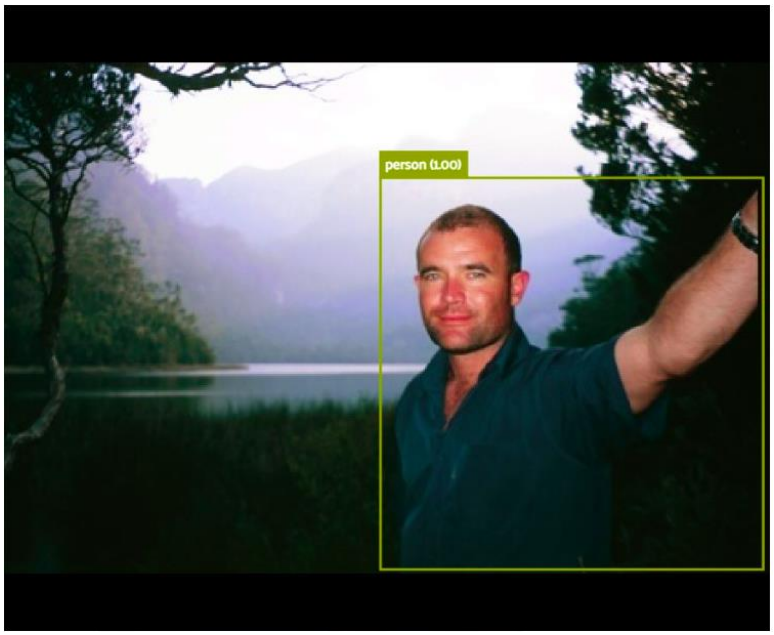


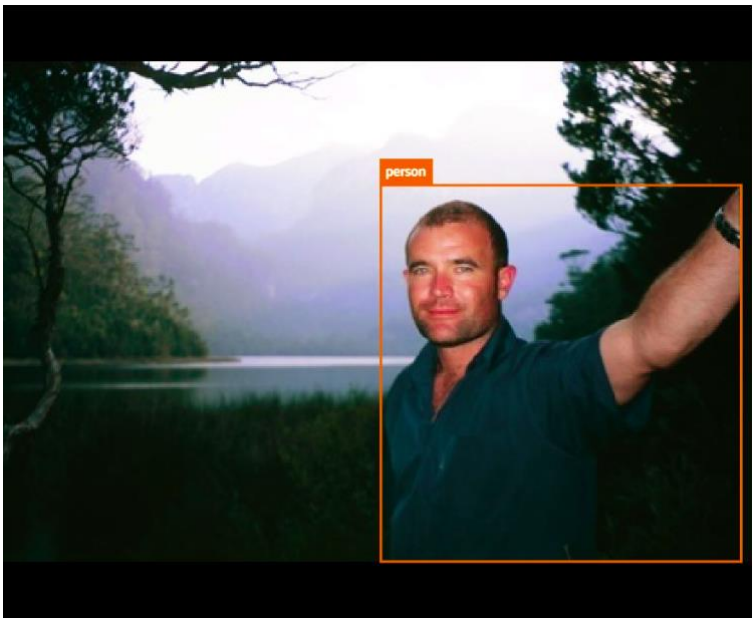
Note: As the image is so crowded, the model predicted most of the object but detected the window as refrigerator and some bottle as cup etc.

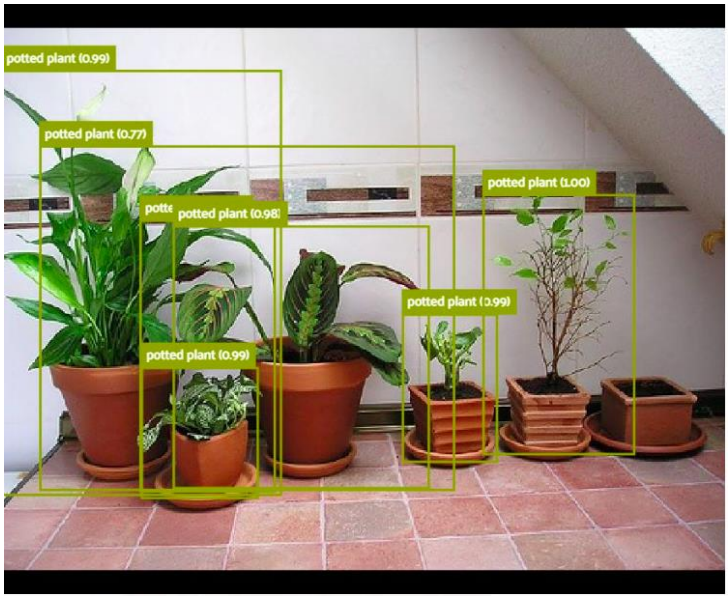
Results on Pascal VOC 2012 dataset

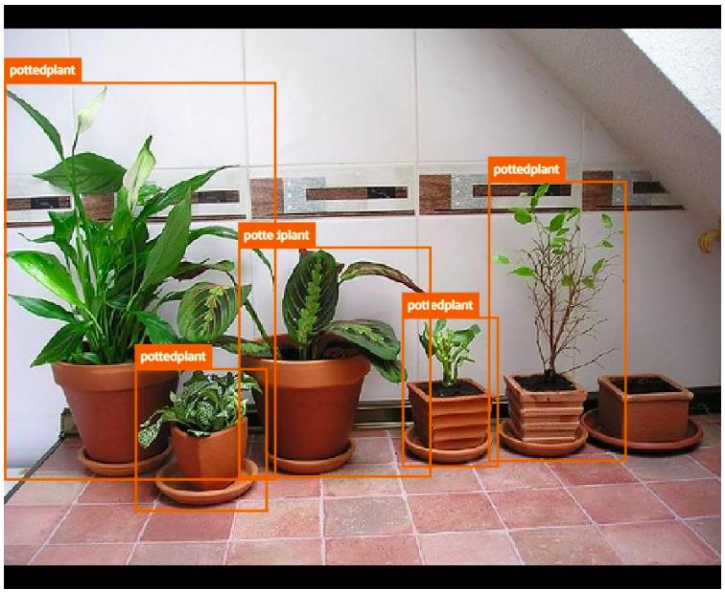





person (1.00)

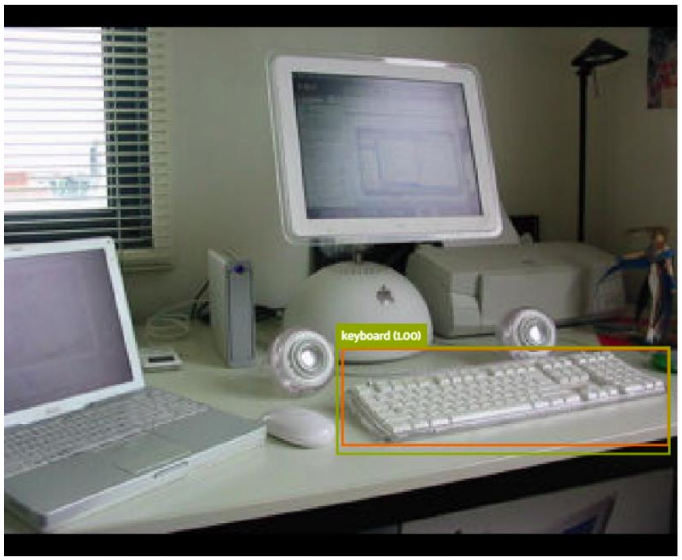

person

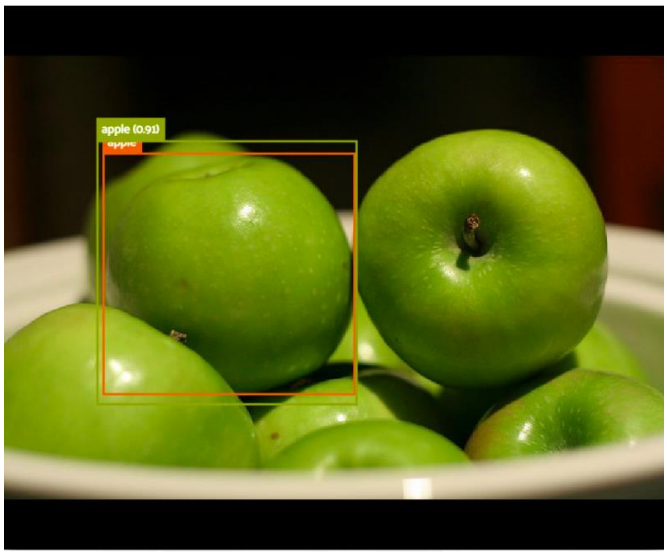


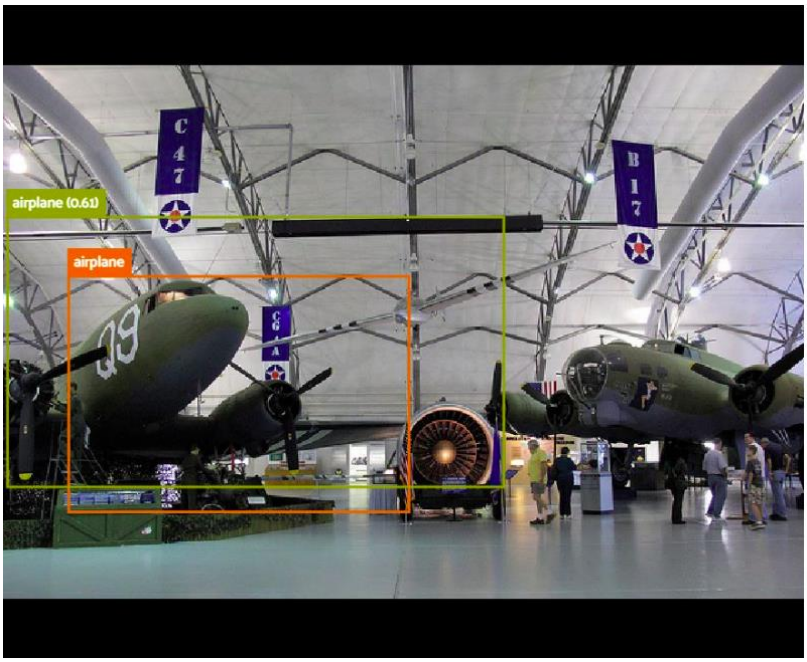


Note: Some bounding boxes are overlapped in the last image but the previous ones are correct with good match

COCO IOU





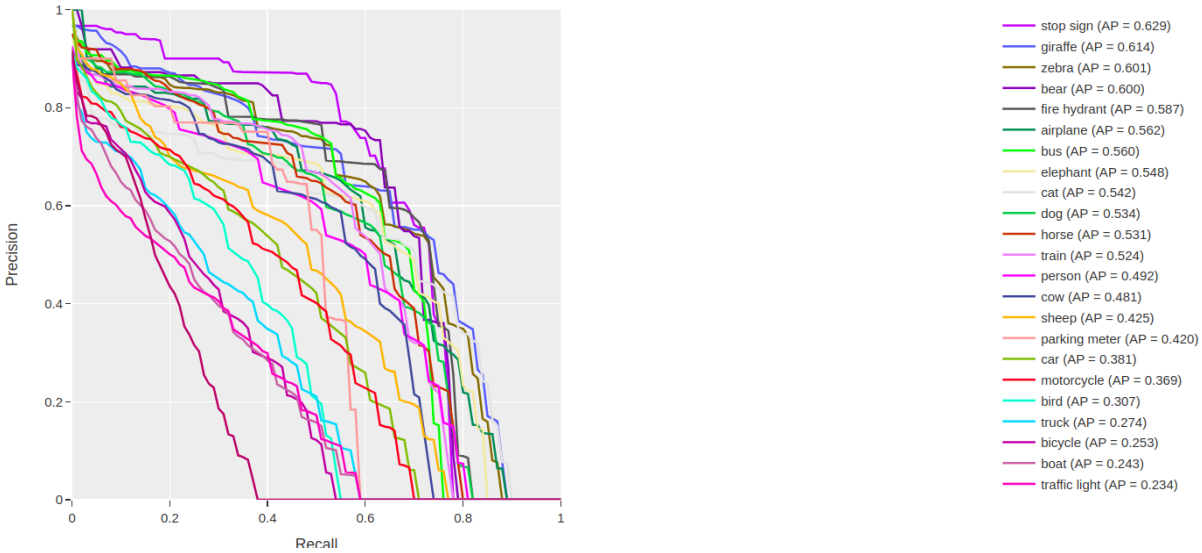


Note: for specific boxes and their GT we compare the IOU, as shown most of them are exactly matched except the last one the box is badly matched .

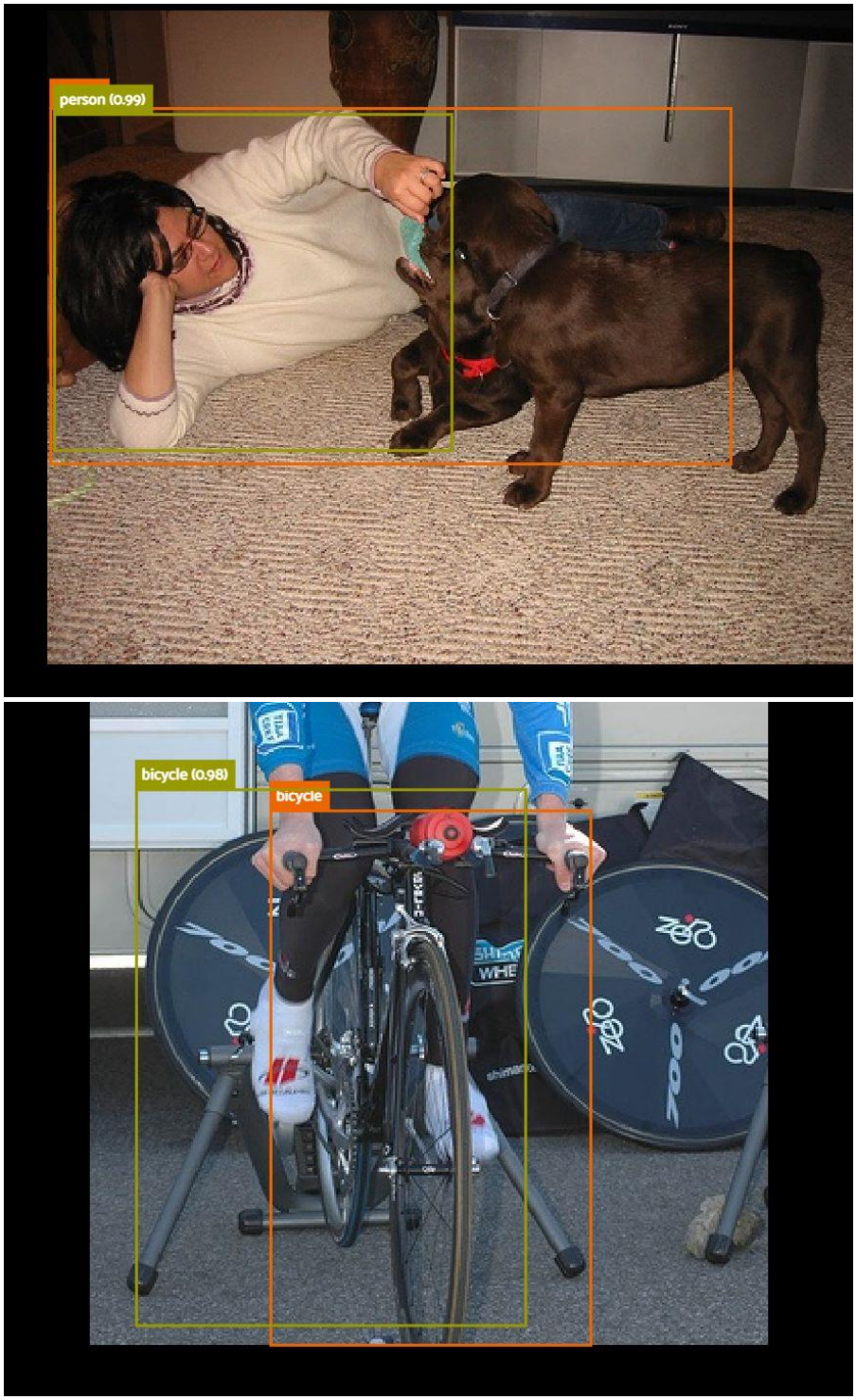
mAP and IOU

For the whole validation set the **mAP = 0.344 and IOU = 0.407**

Precision vs Recall Graph



COCO IOU

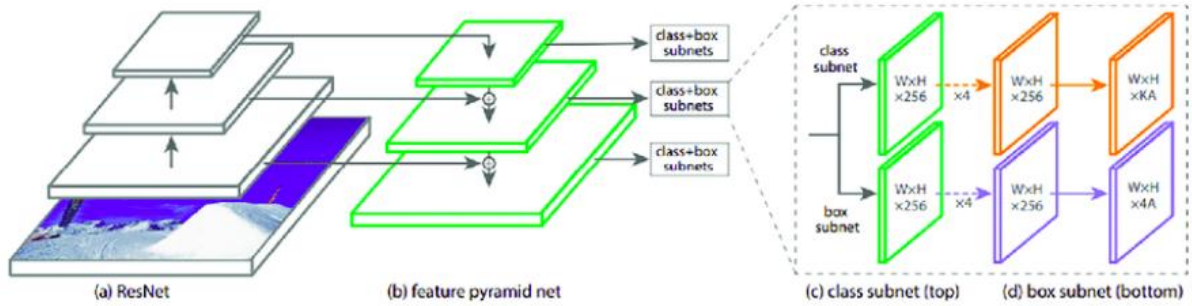


mAP and IOU

For the whole validation set the **mAP = 0.348 and IOU = 0.252**

2- RetinaNet

Architecture diagram



Discussion

RetinaNet is one of the best one-stage object detection models that has proven to work well with **dense and small scale objects**. For this reason, it has become a popular object detection model to be used with aerial and satellite imagery.

There are four major components of a RetinaNet model architecture :

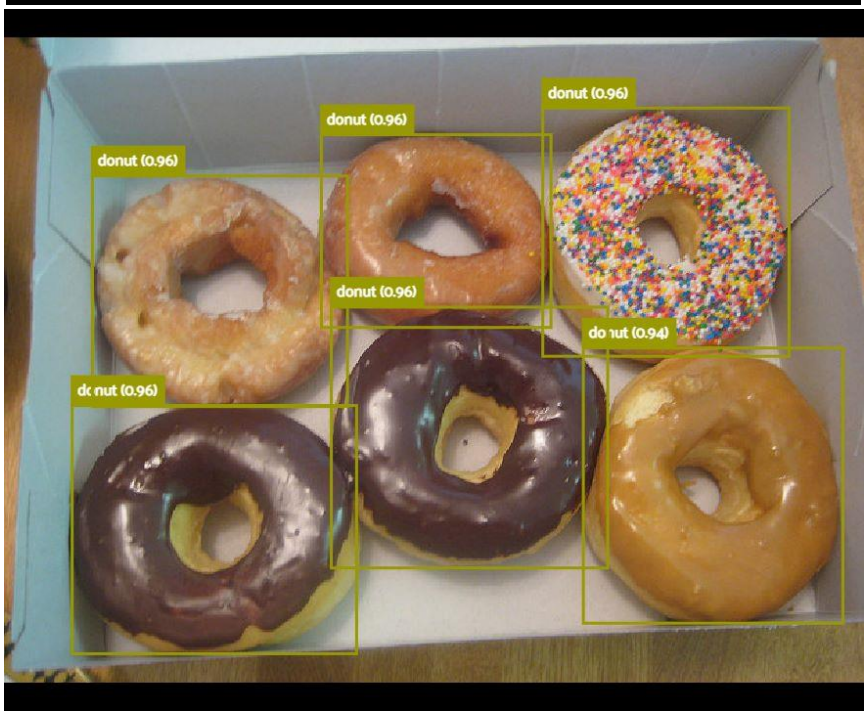
a) Bottom-up Pathway - **The backbone network (e.g. ResNet)** which calculates the feature maps at different scales, irrespective of the input image size or the backbone.

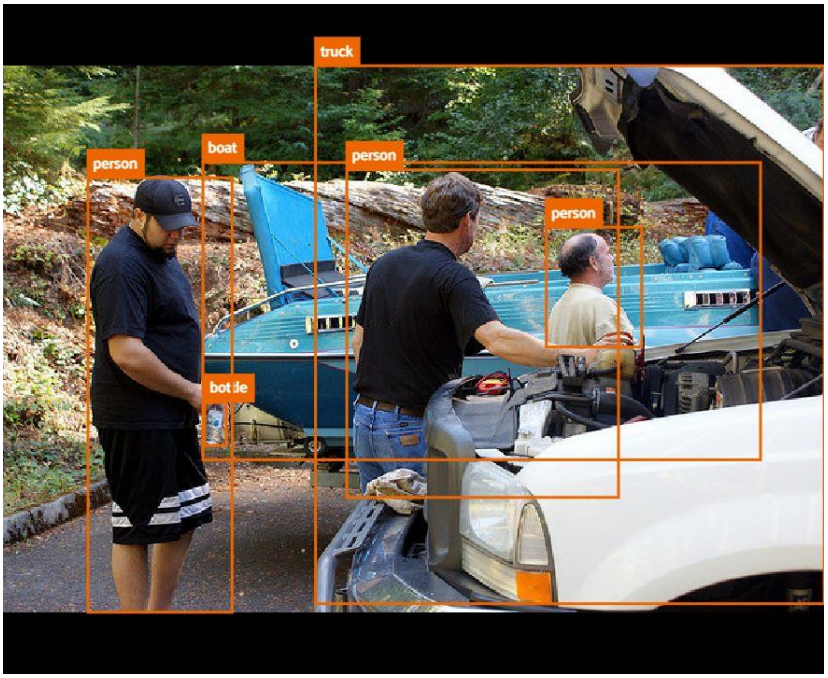
b) Top-down pathway and Lateral connections - The top down pathway upsamples the spatially coarser feature maps from higher pyramid levels, and the lateral connections merge the top-down layers and the bottom-up layers with the same spatial size.

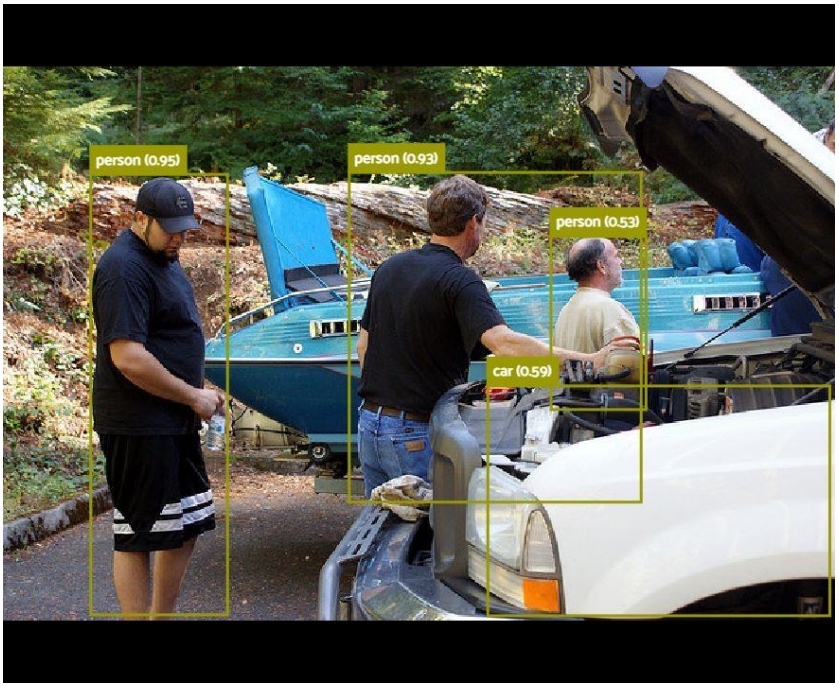
c) Classification subnetwork - It predicts the probability of an object being present at each spatial location for each anchor box and object class.

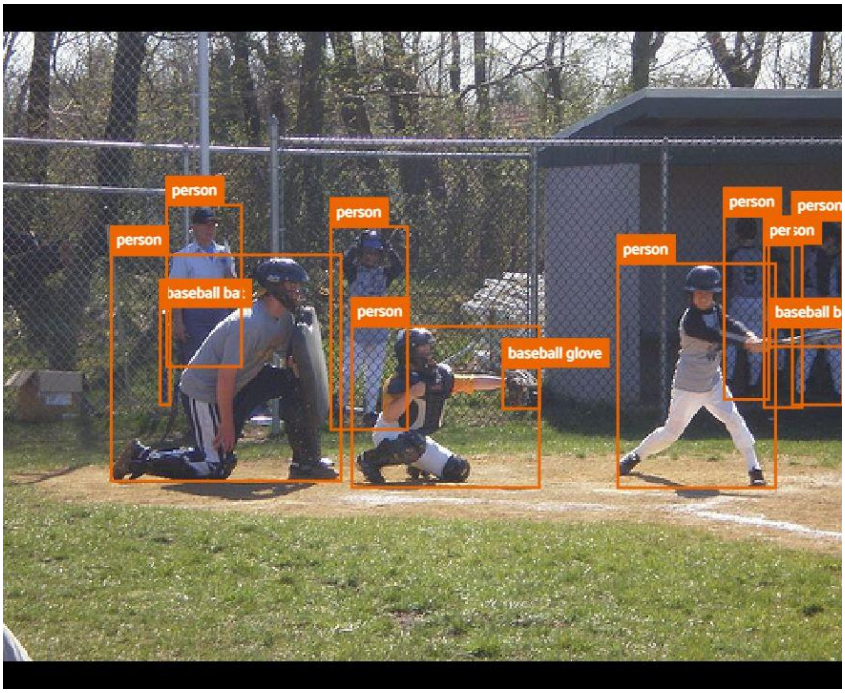
d) Regression subnetwork - It regresses the offset for the bounding boxes from the anchor boxes for each ground-truth object.

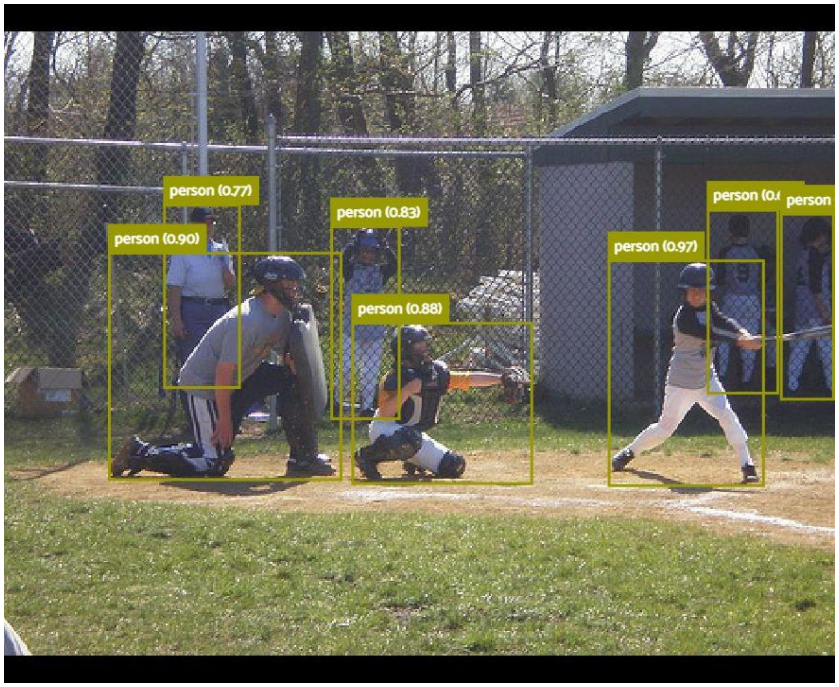
Results on COCO dataset

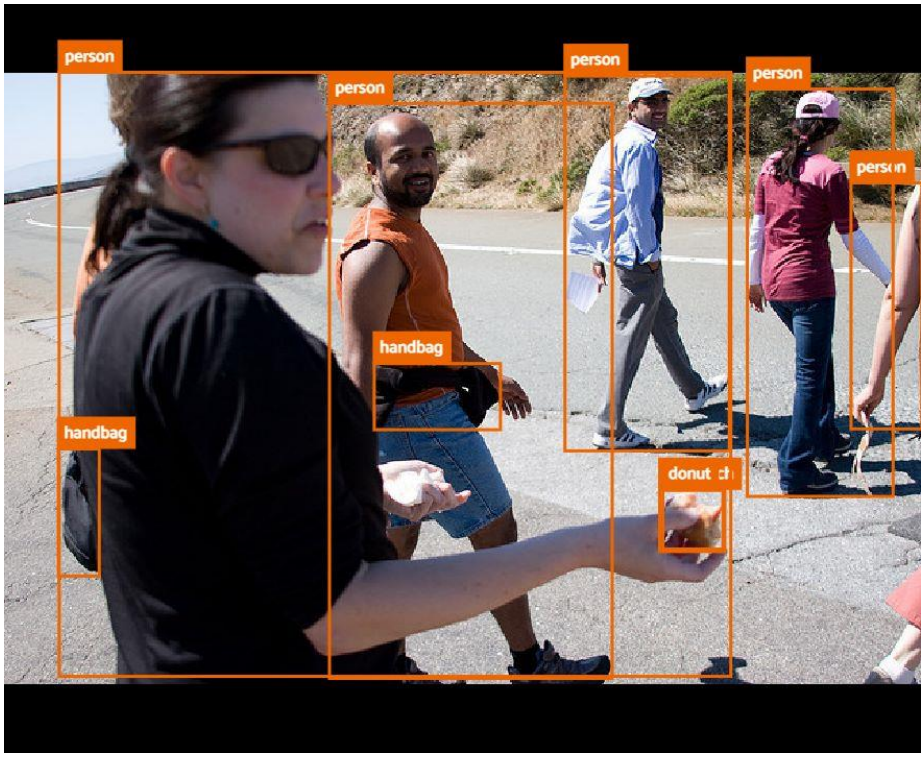


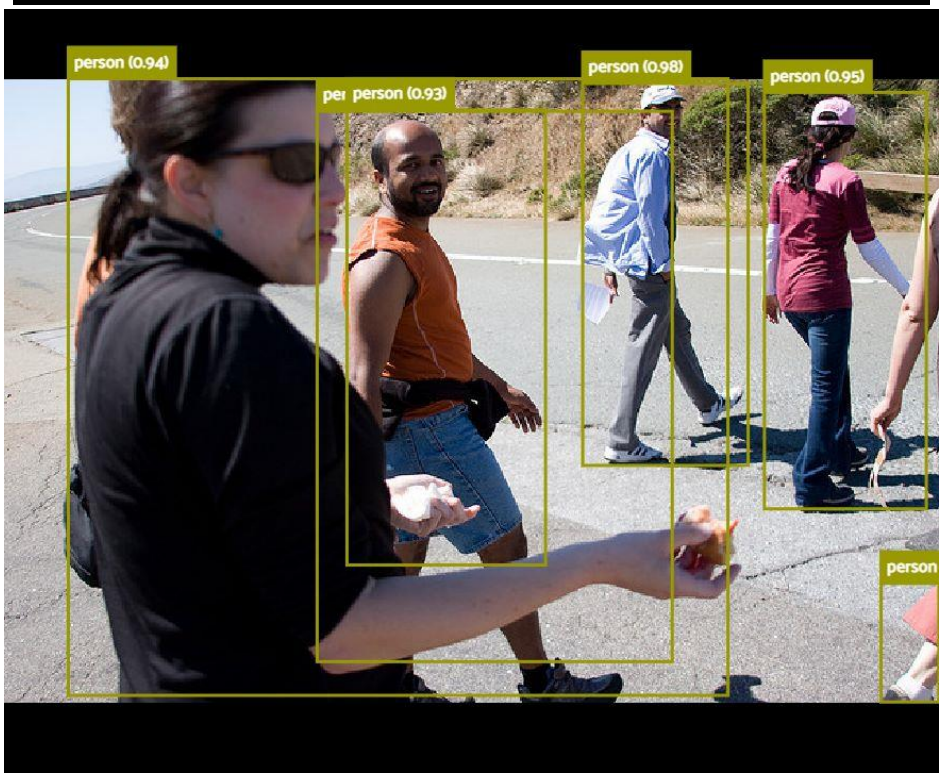

truck
boat
person
person
person
bot le


person (0.95)
person (0.93)
person (0.53)
car (0.59)

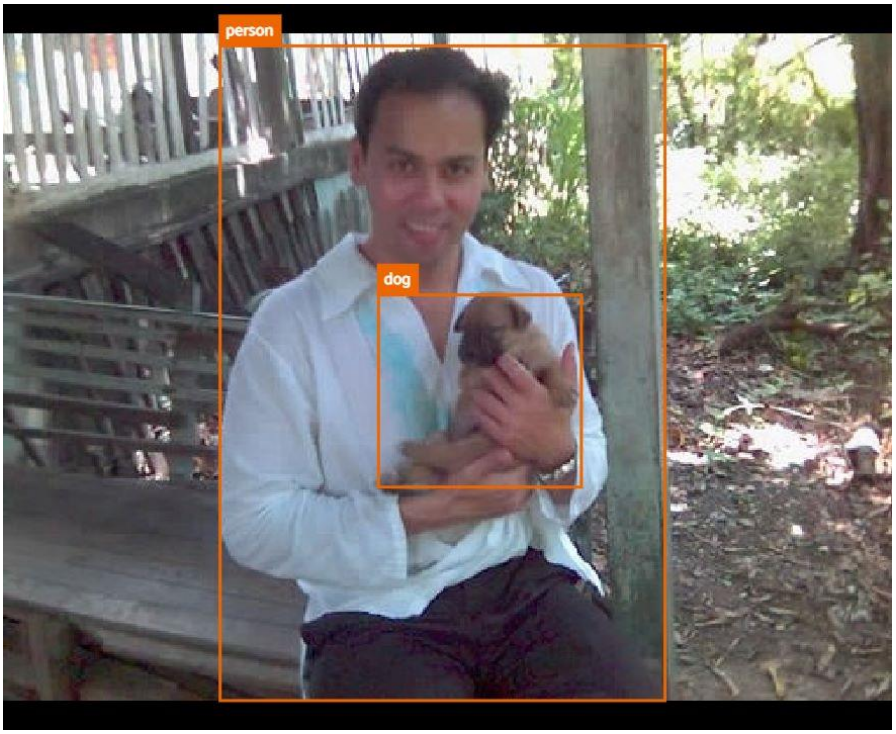

person
person
person
baseball ba
person
baseball glove
person
person
person
baseball b

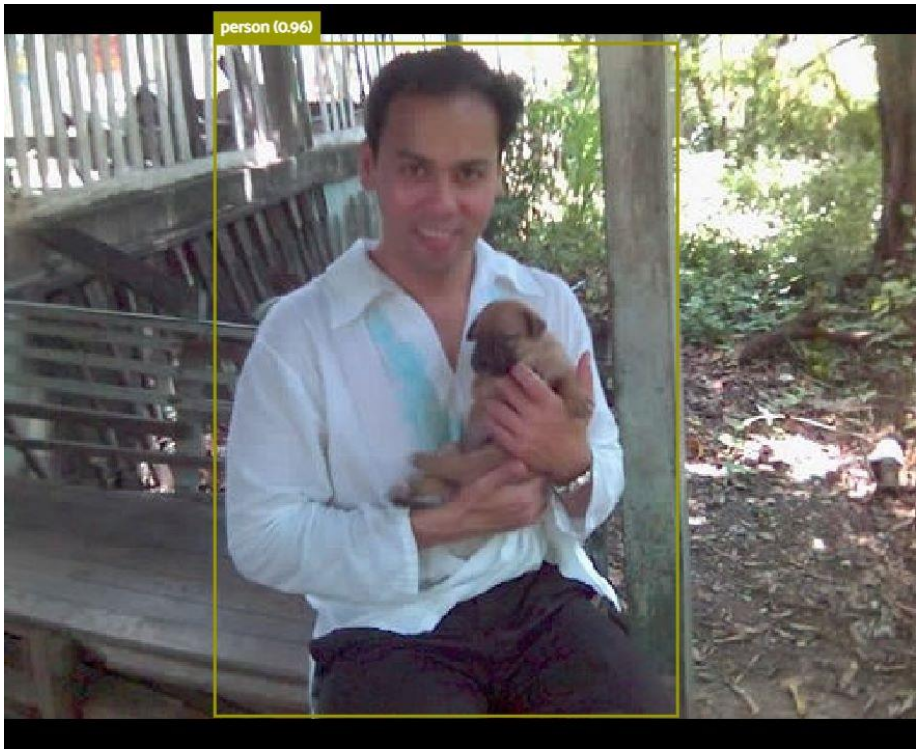

person (0.77)
person (0.90)
person (0.83)
person (0.88)
person (0.97)
person (0.4
person

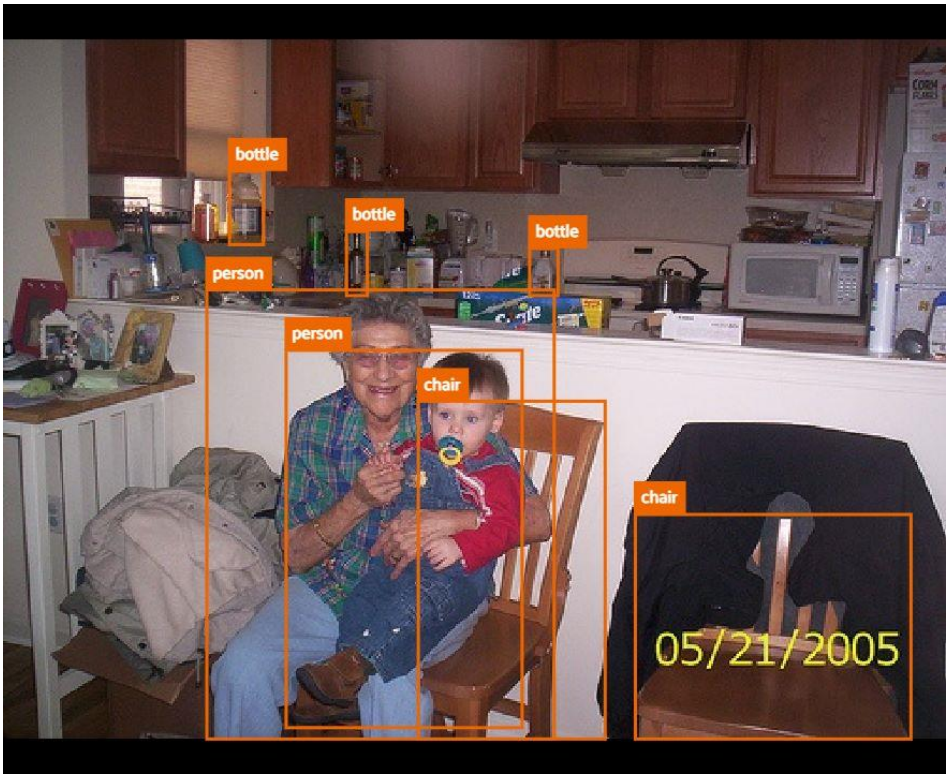

person
person
person
person
handbag
handbag
donut

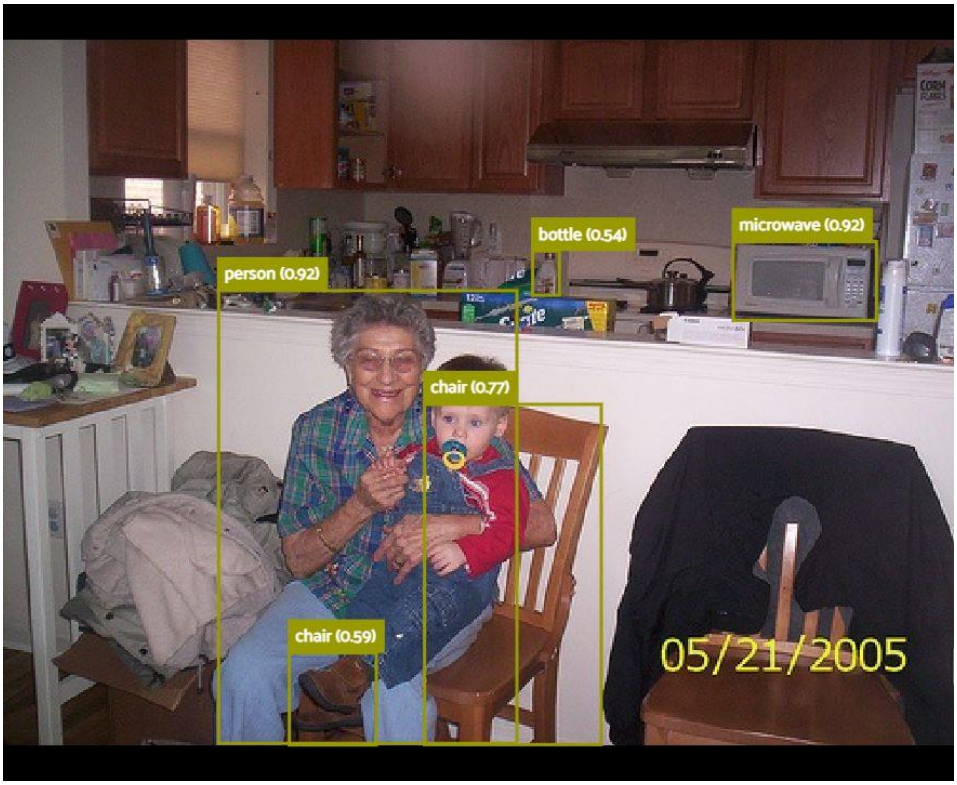

person (0.94)
person (0.93)
person (0.98)
person (0.95)
person

Results on different dataset



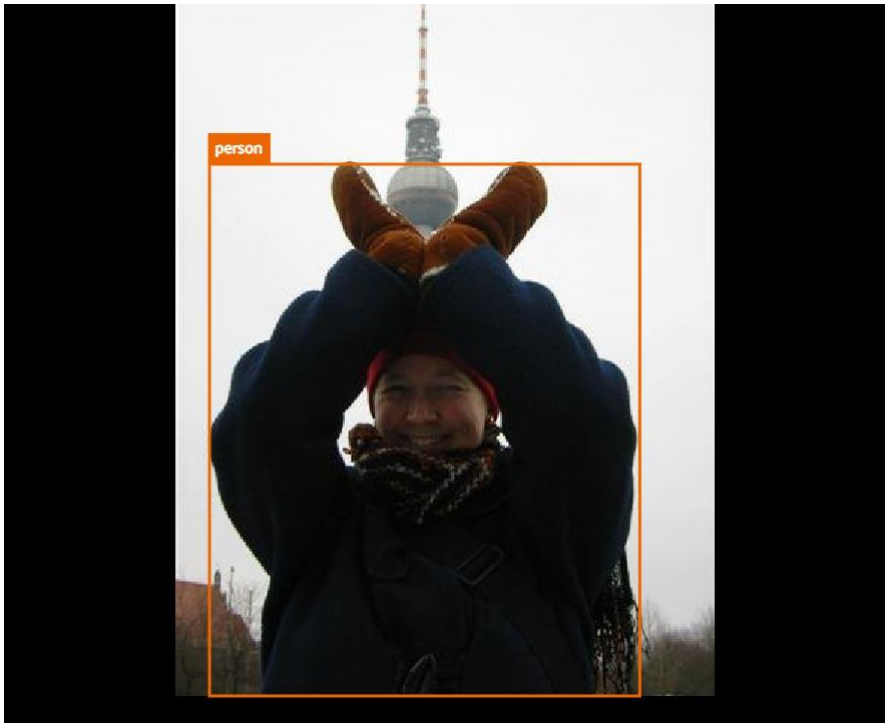


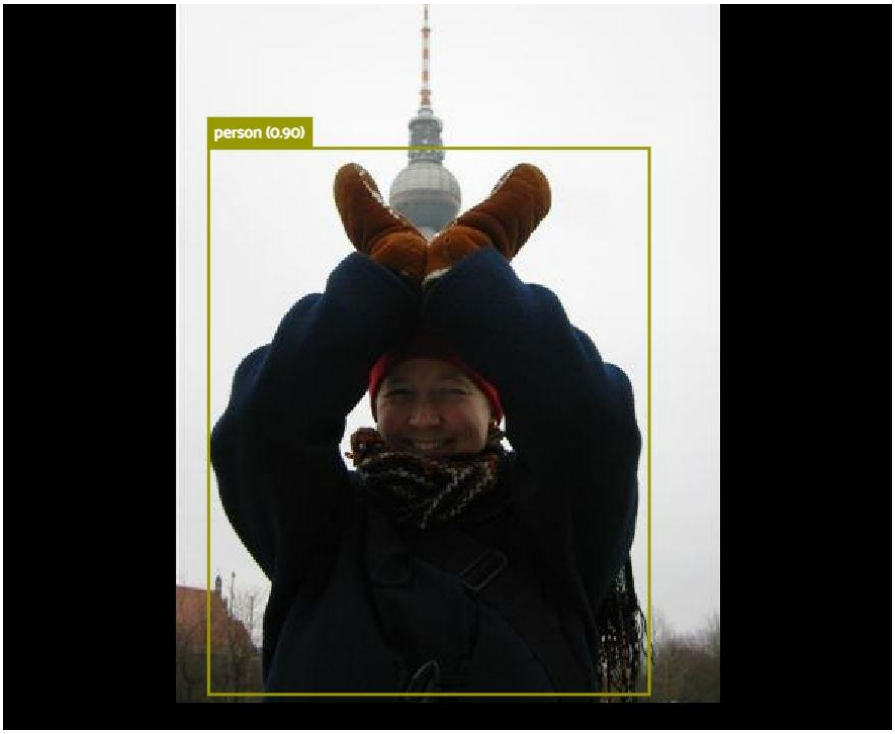

bottle
bottle
bottle
person
person
chair
chair
05/21/2005


bottle (0.54)
microwave (0.92)
person (0.92)
chair (0.77)
chair (0.59)
05/21/2005

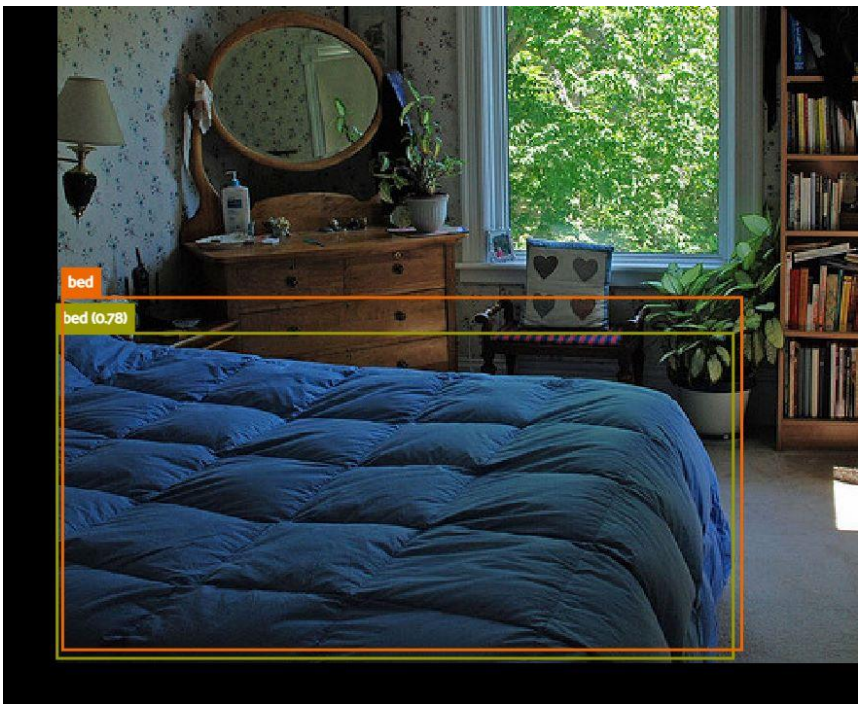

bus
VICTORY LINER
1888
1888


bus (0.99)
person (0.57)
VICTORY LINER
1888
1888


person


person (0.90)

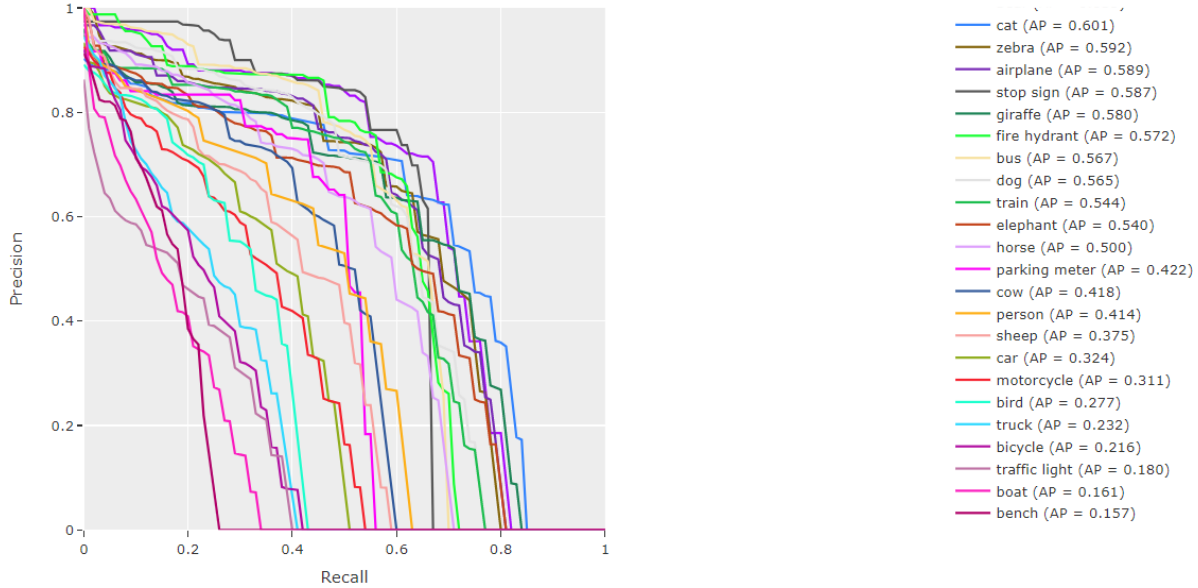
COCO IOU



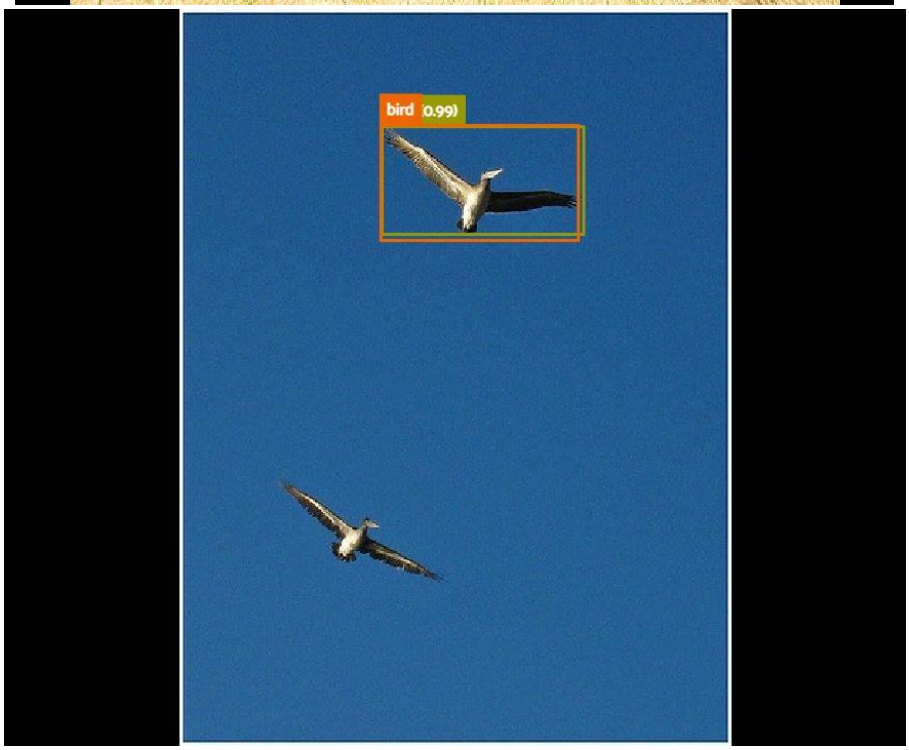
mAP and IOU

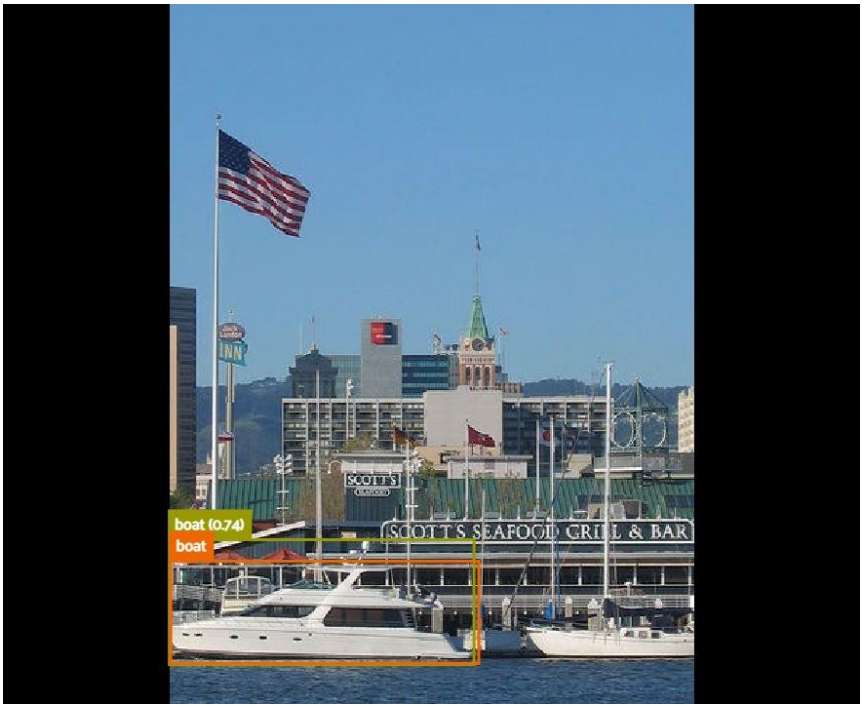
For the whole validation set the **mAP = 0.297 and IOU = 0.368**

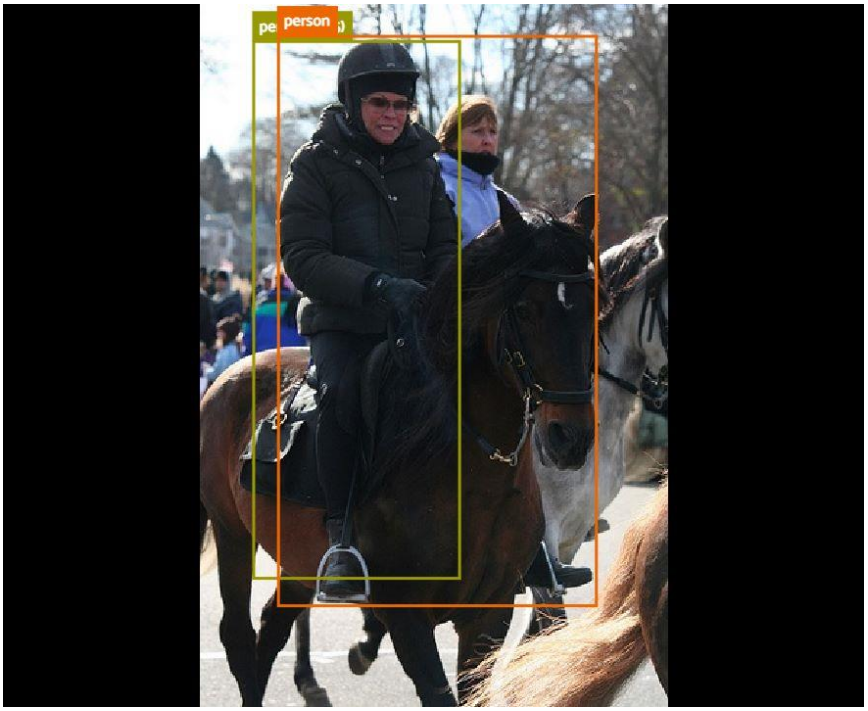
Precision vs Recall Graph

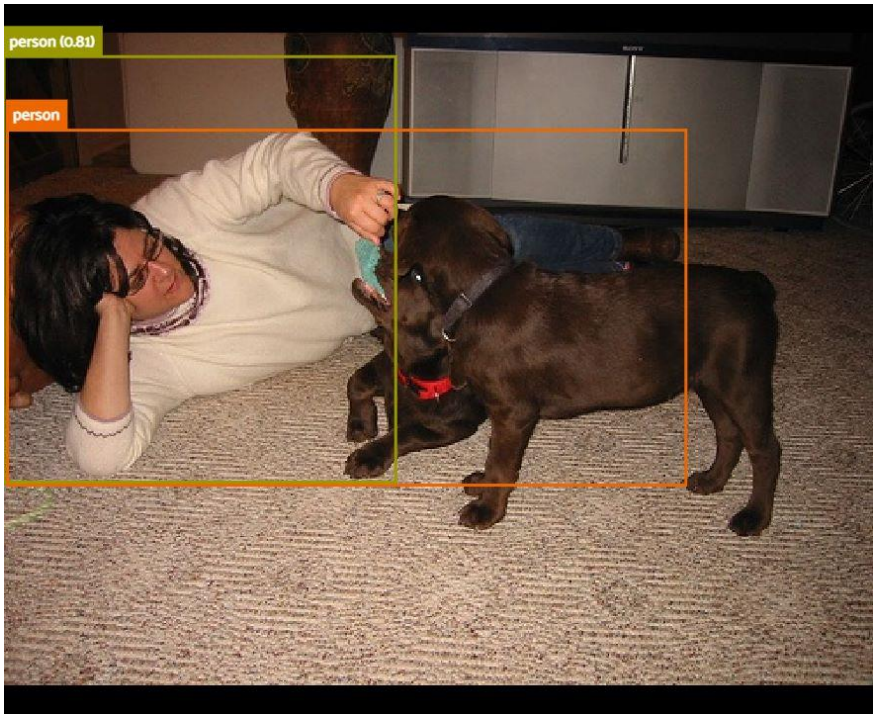


VOC IOU




boat (0.74)
boat
SCOTT'S
SEAFOOD GRILL & BAR


person

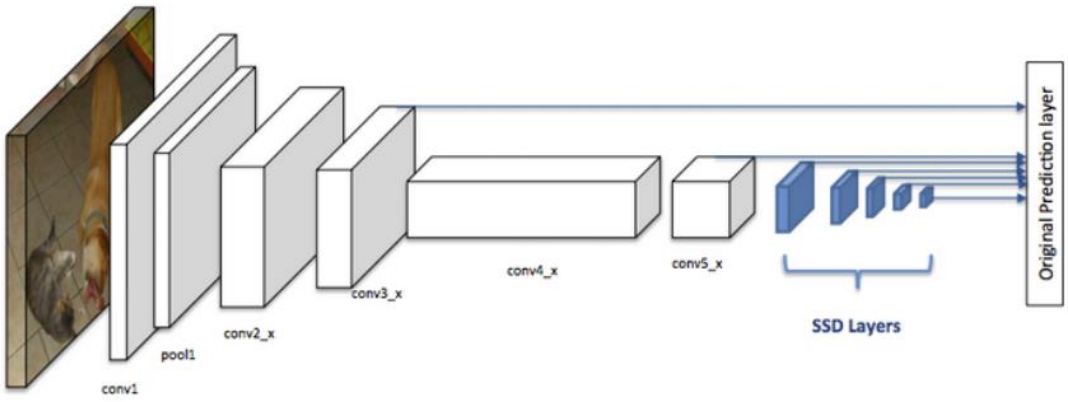


mAP and IOU

For the whole validation set the **mAP = 0.35 and IOU = 0.36**

3- Single Shot Detector

Architecture diagram

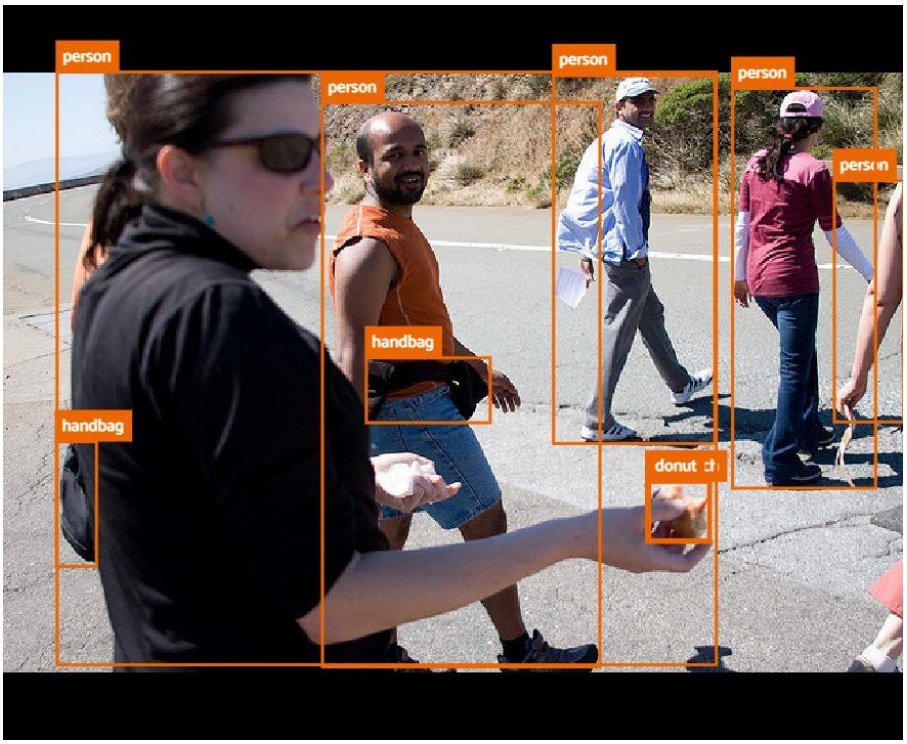


Discussion

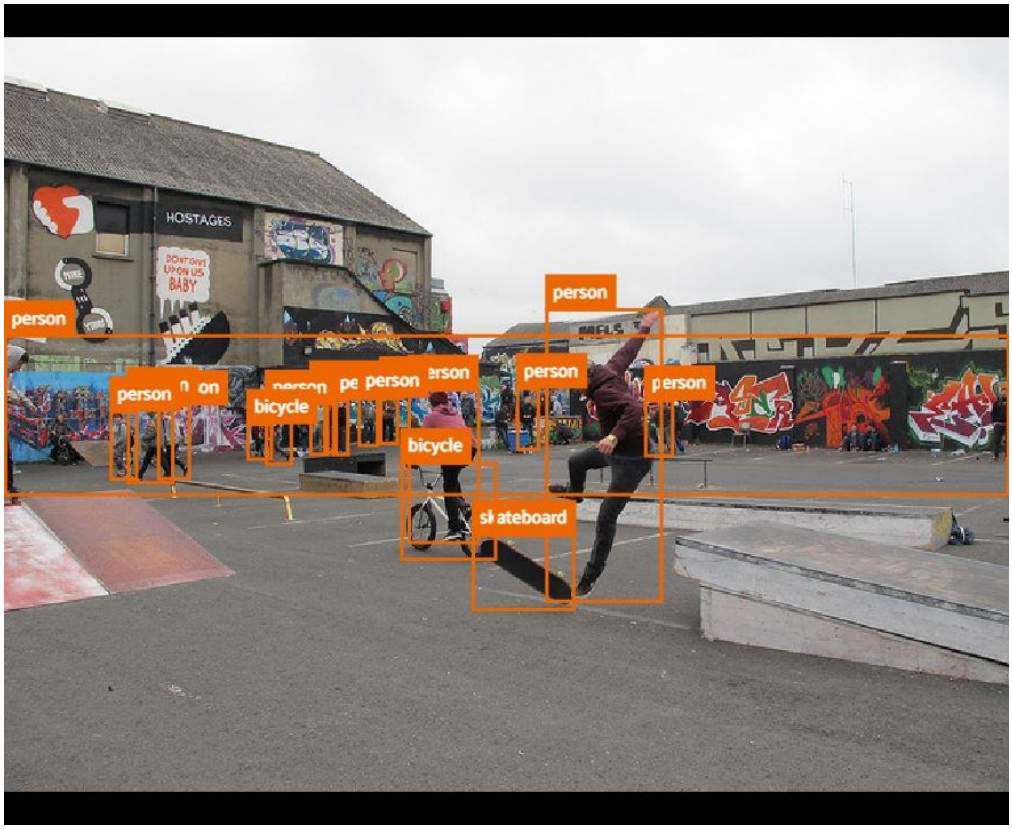
SSD has two components: a backbone model and SSD head.

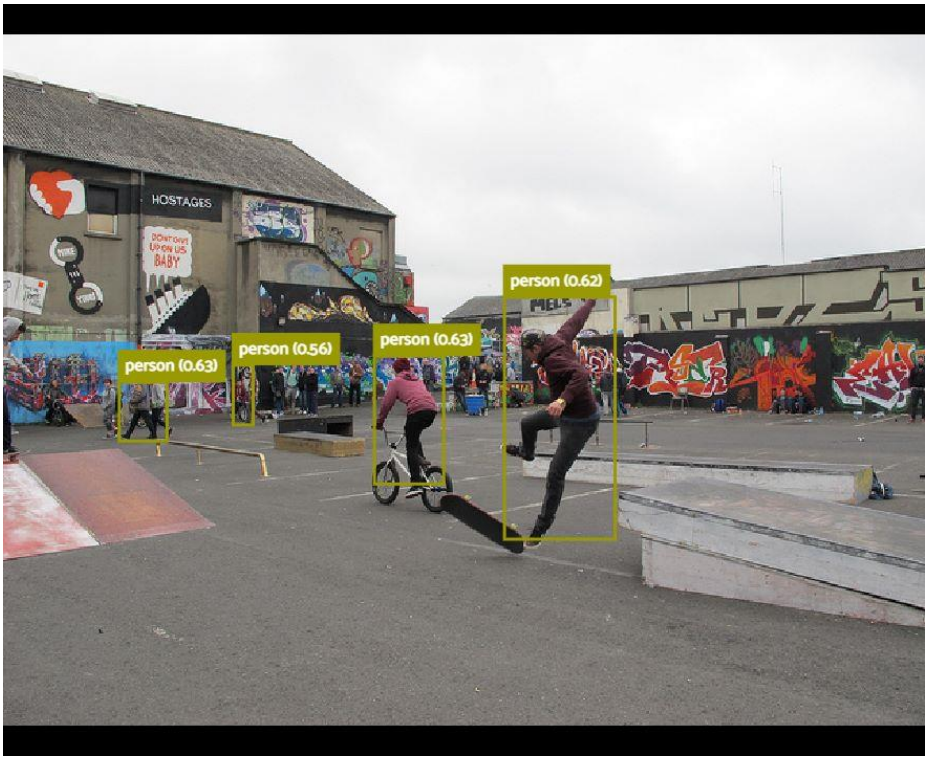
Backbone model usually is a pre-trained image classification network as a feature extractor. This is typically a network like **ResNet trained on ImageNet** from which the final fully connected classification layer has been removed. We are thus left with a deep neural network that is able to extract semantic meaning from the input image The *SSD head* is just one or more convolutional layers added to this backbone and the outputs are interpreted as the **bounding boxes and classes of objects in the spatial location of the final layers activations**.

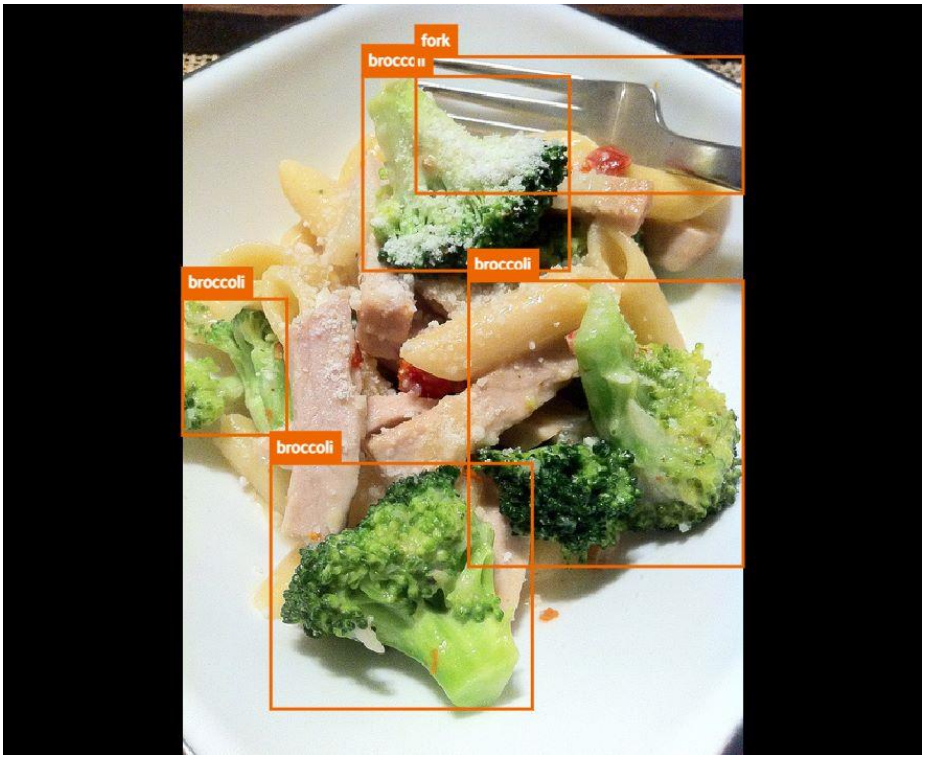
Results on COCO dataset

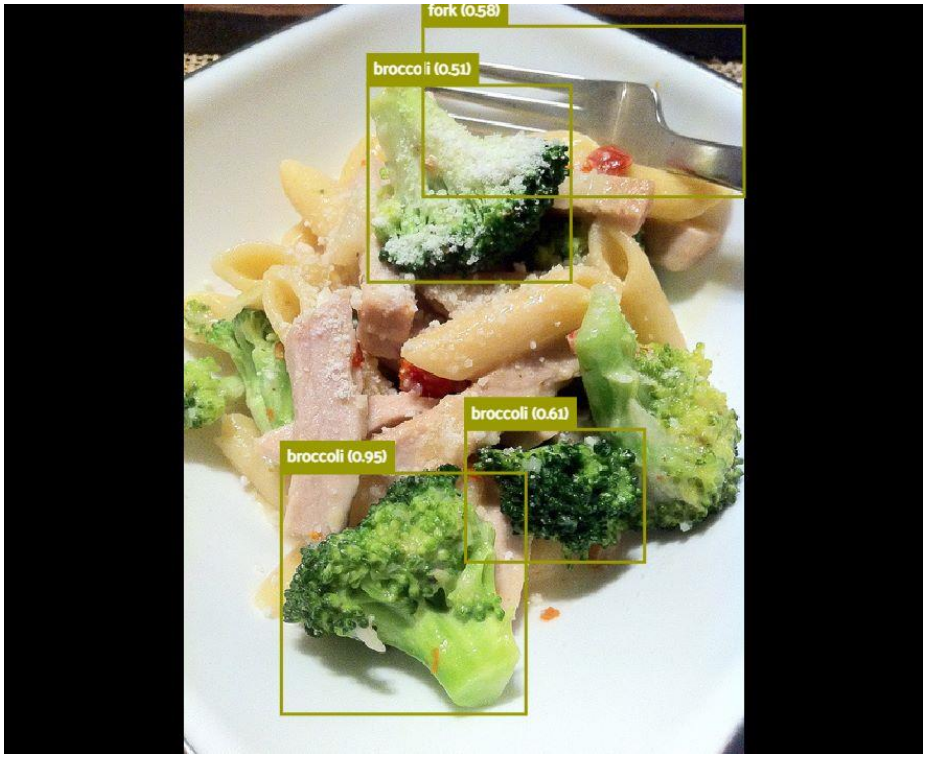


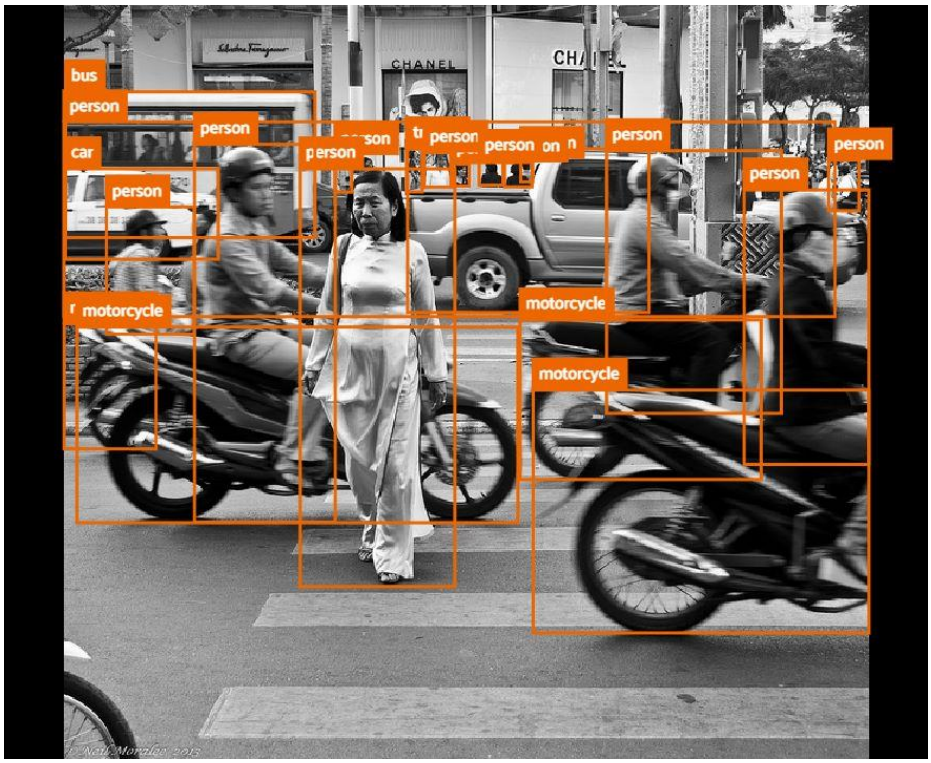

person (0.98)
person (0.82)
person (1.00)
person (1.00)

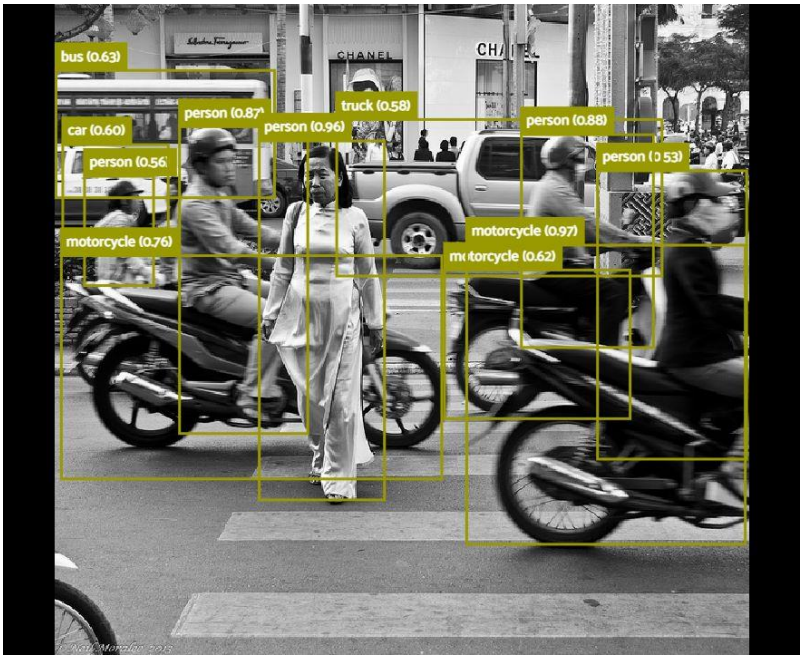

HOSTAGES
person
person
bicycle
bicycle
person
person
person
skateboard


HOSTAGES
person (0.62)
person (0.63)
person (0.56)
person (0.63)


fork
broccoli
broccoli
broccoli


fork (0.58)
broccoli (0.51)
broccoli (0.61)
broccoli (0.95)

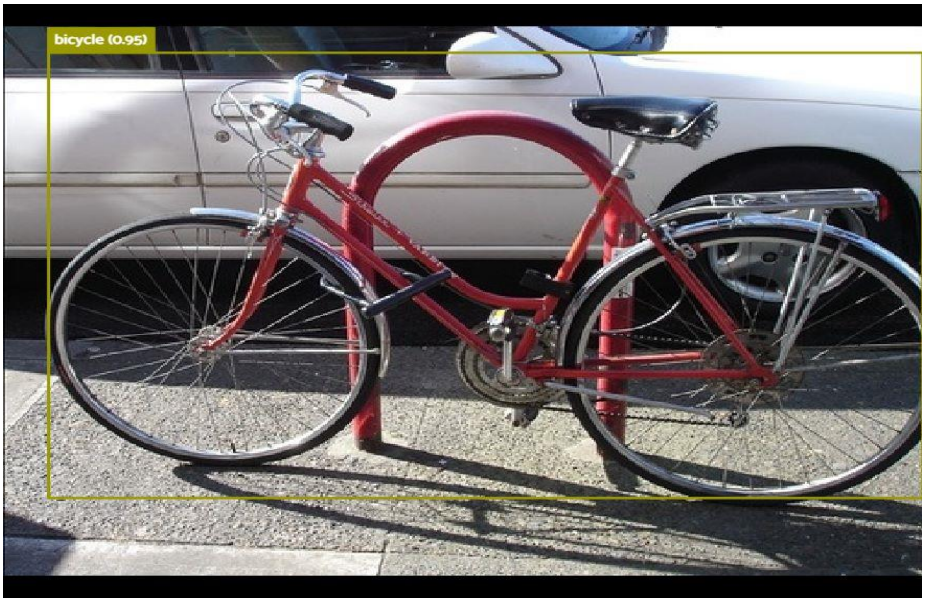

bus
person
car
motorcycle
motorcycle
motorcycle

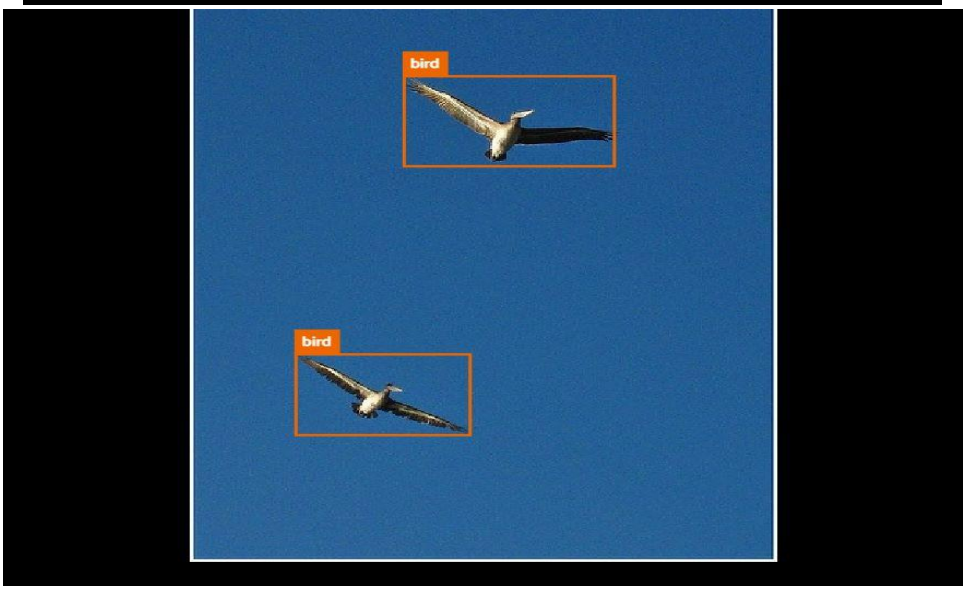


Results on different dataset

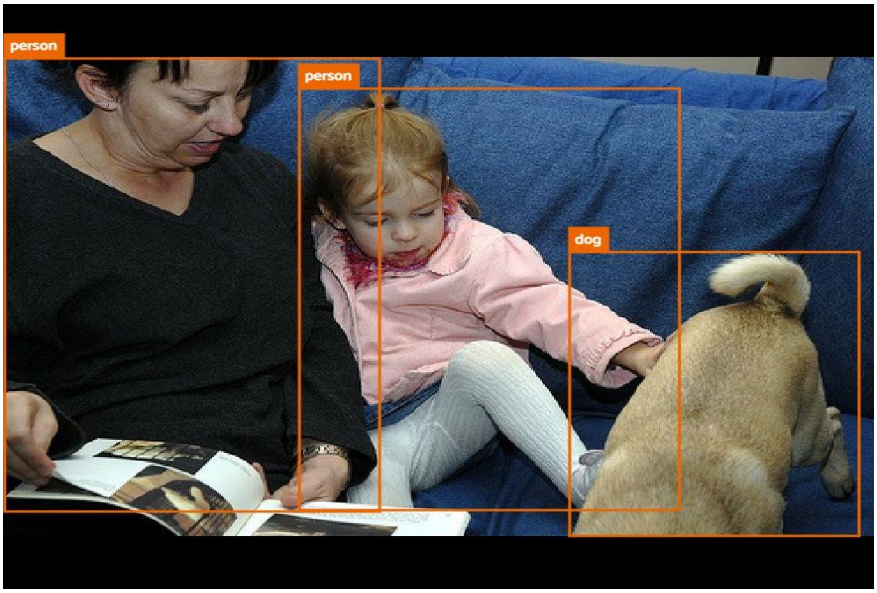

horse (0.99)

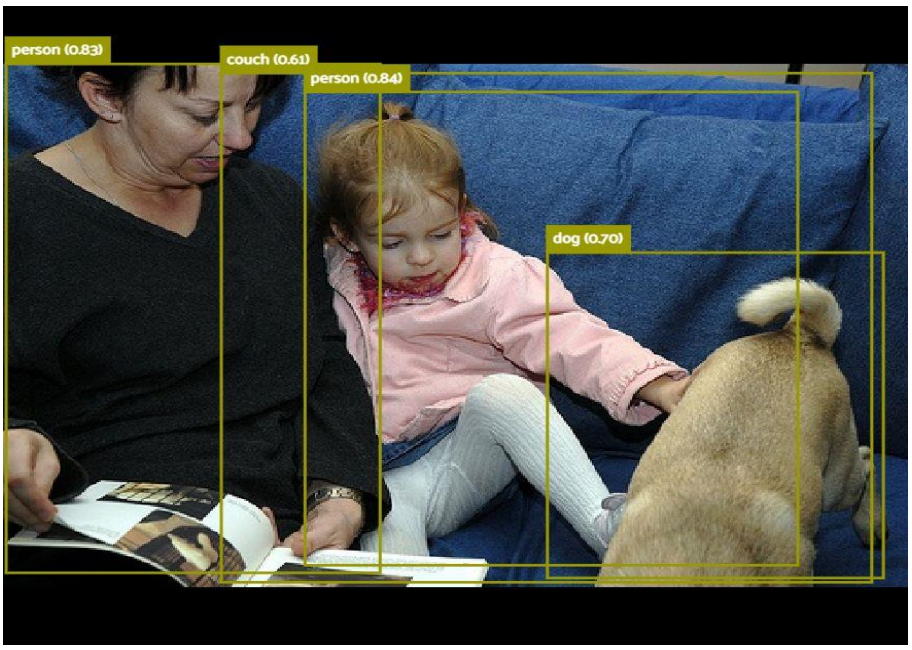

car
bicycle


bicycle (0.95)

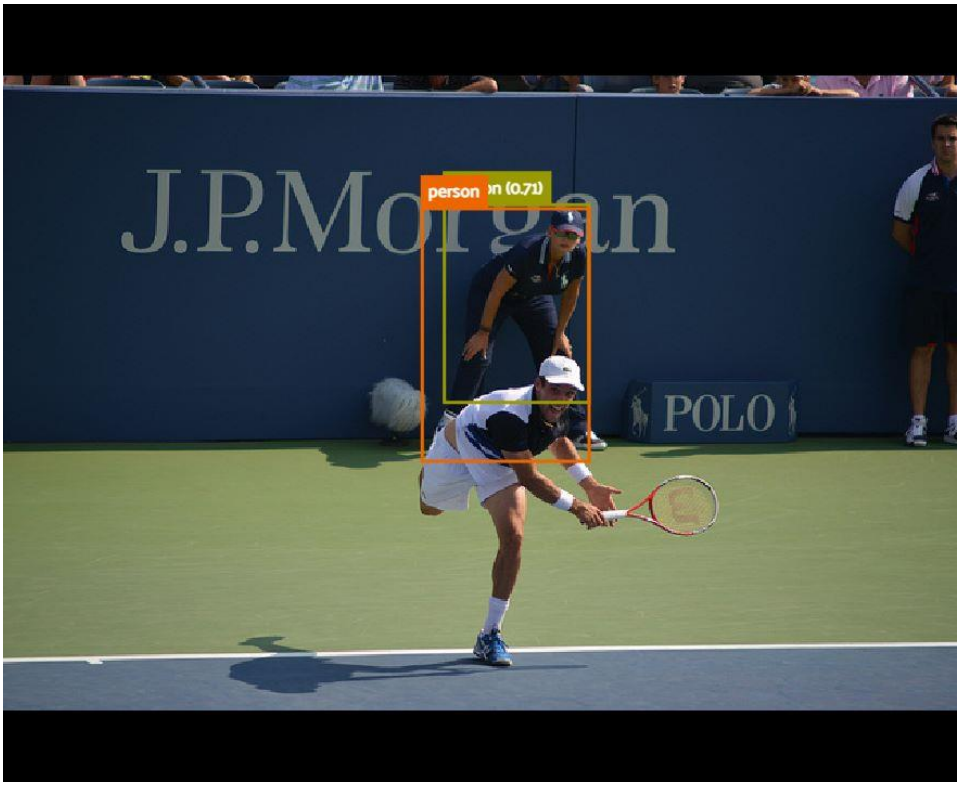

bird
bird

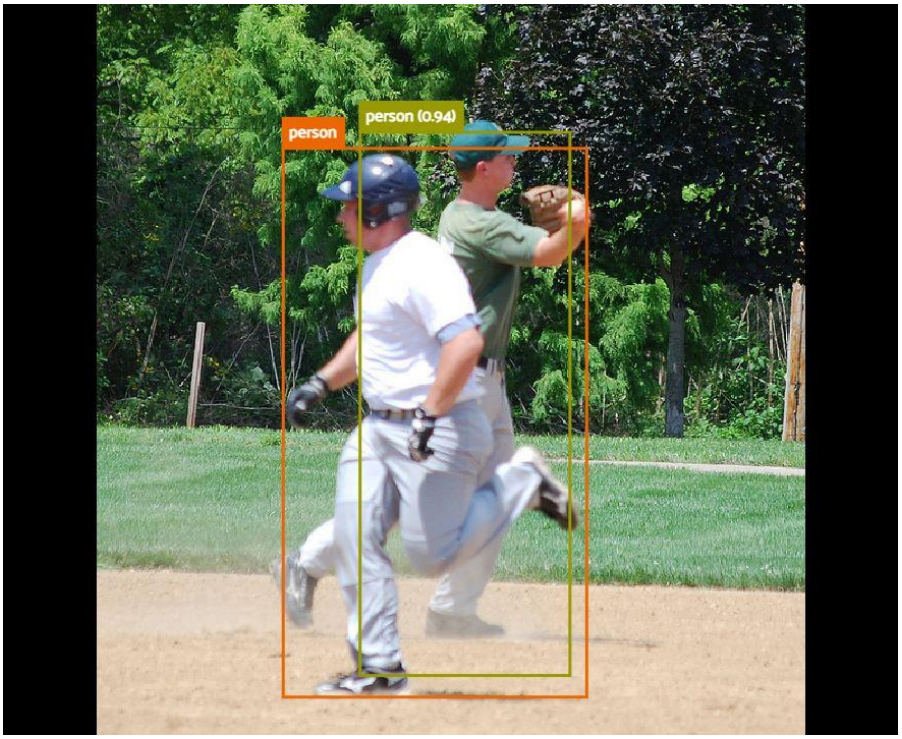

airplane (0.64)

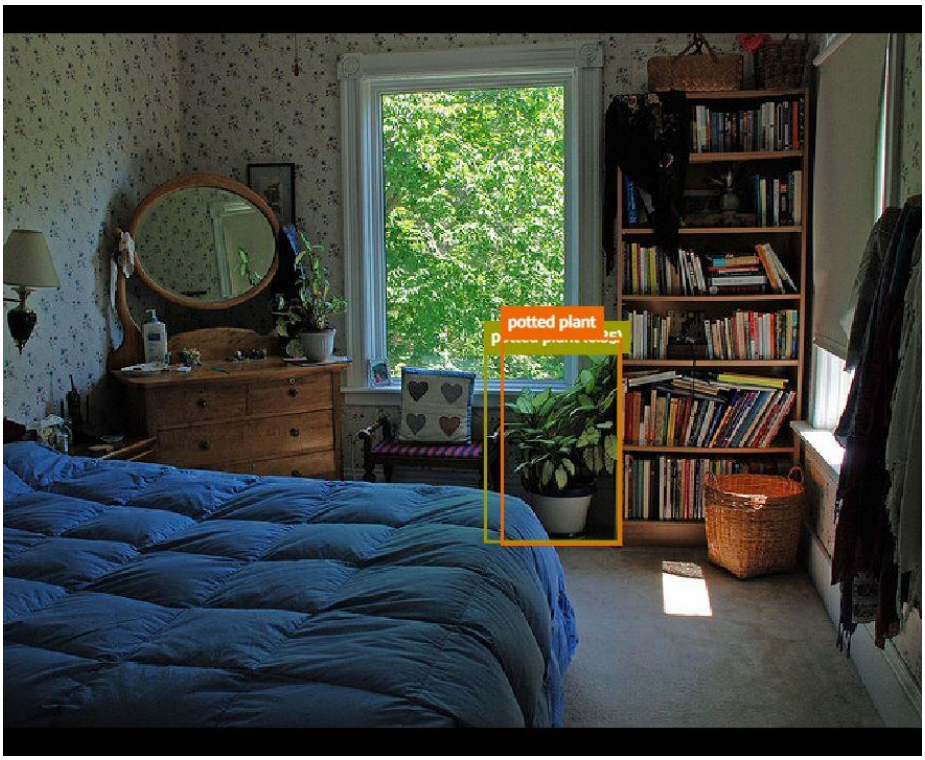

person
person
dog


person (0.83)
couch (0.61)
person (0.84)
dog (0.70)

COCO IOU



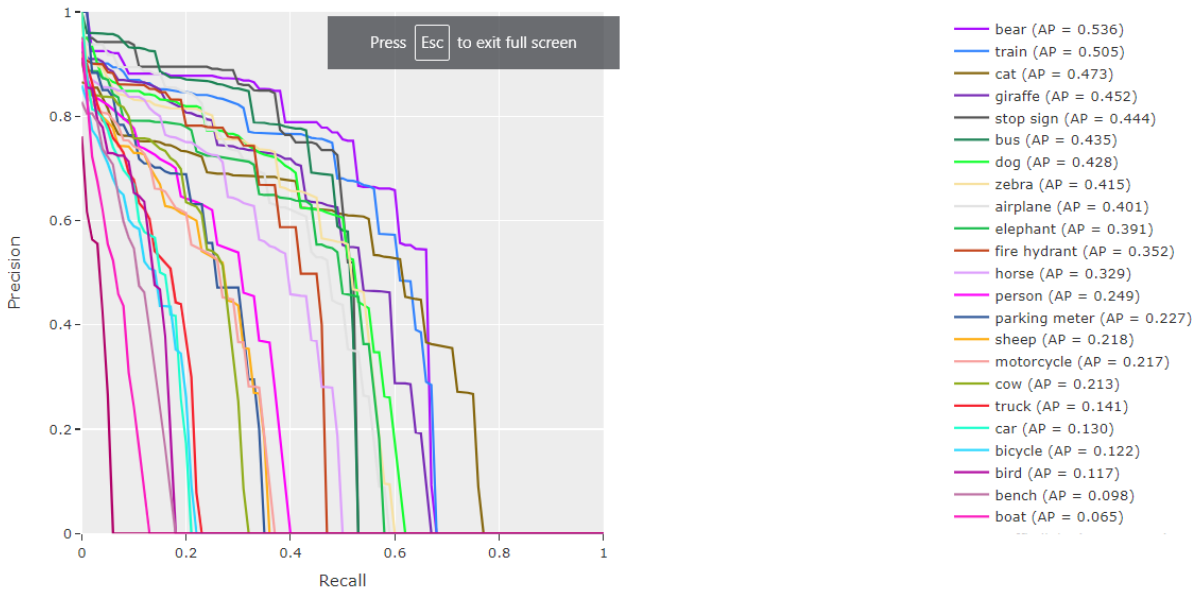




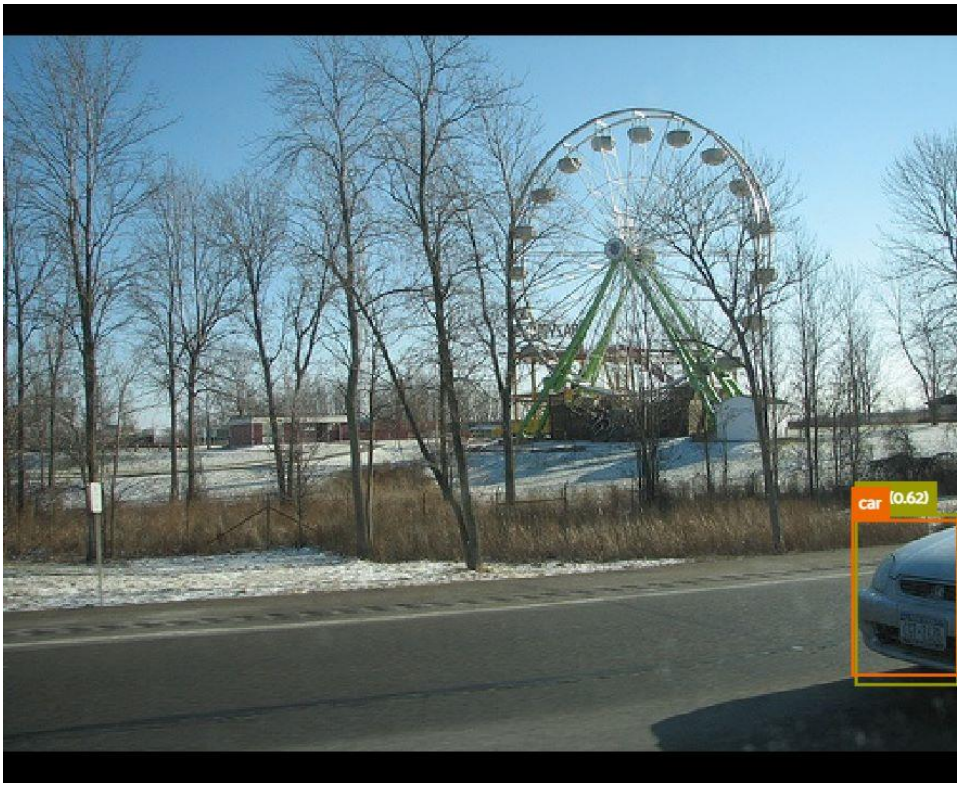
mAP and IOU

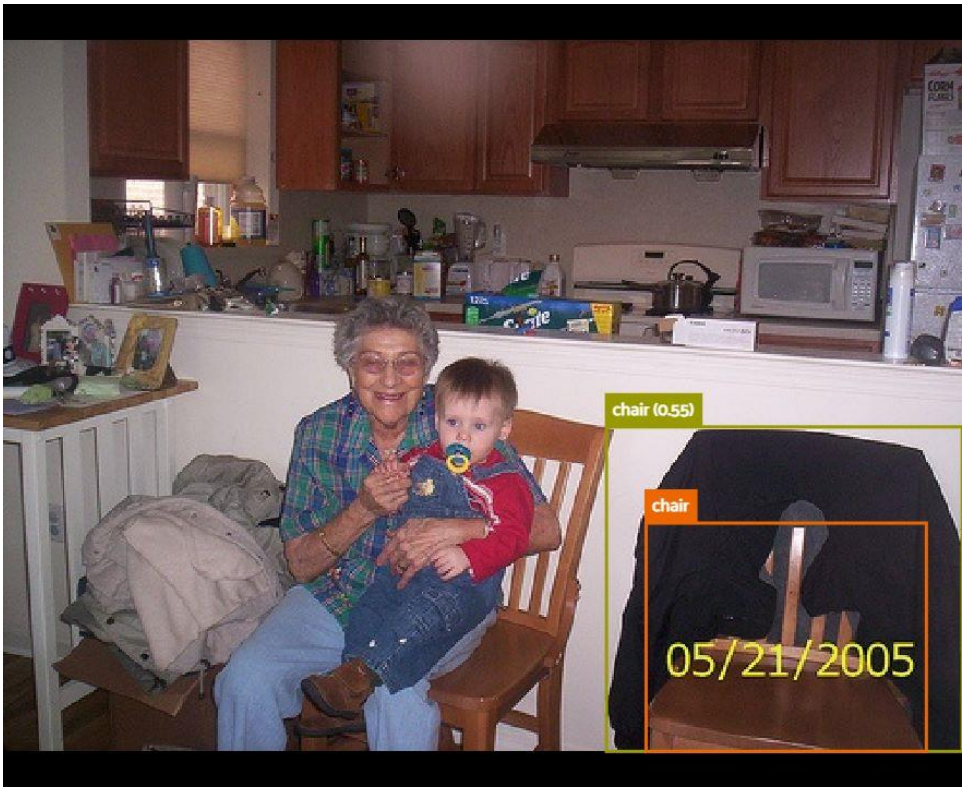
For the whole validation set the **mAP = 0.182 and IOU = 0.221**

Precision vs Recall Graph

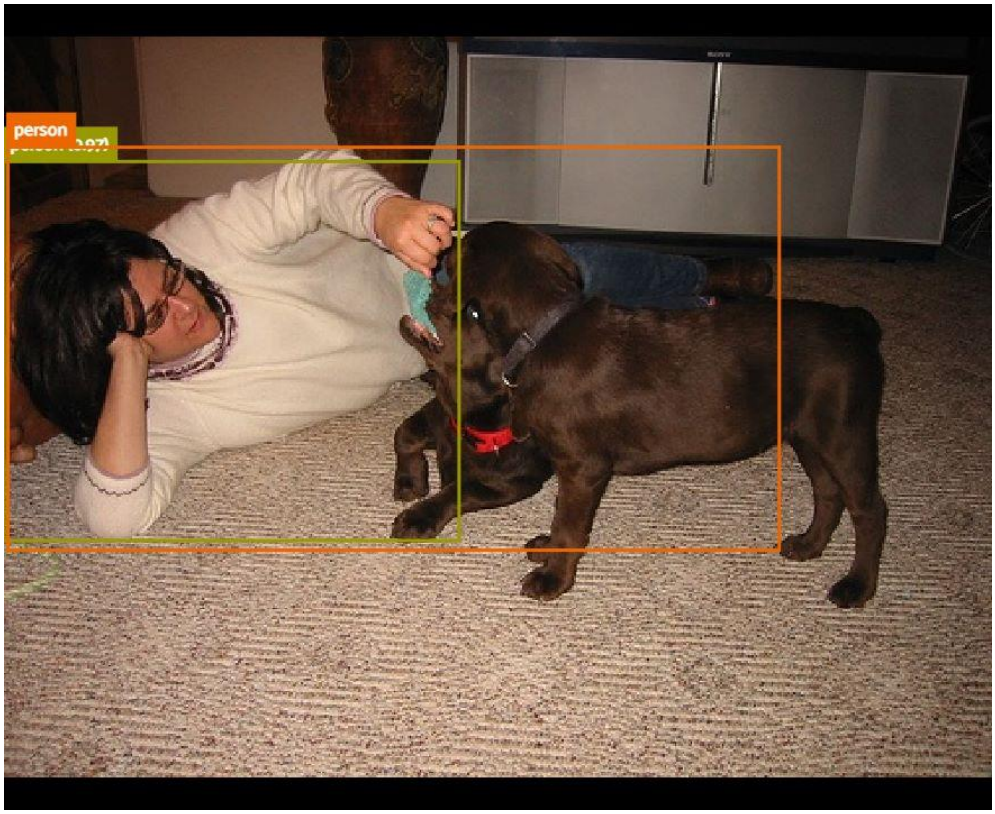


Voc IOU




chair (0.55)
chair
05/21/2005

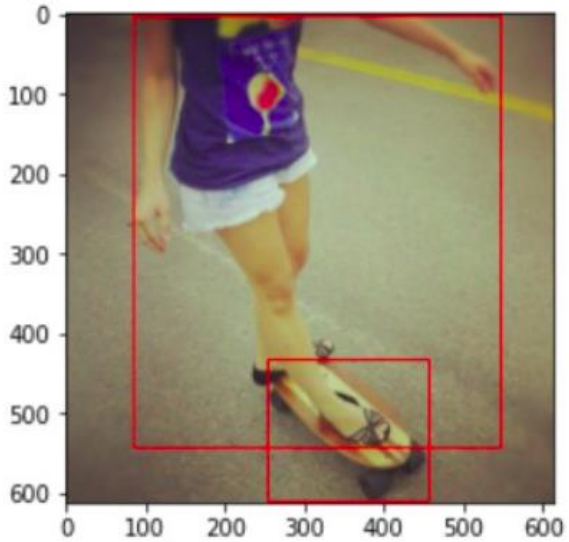

person
person (0.55)
Saab Ice Experience

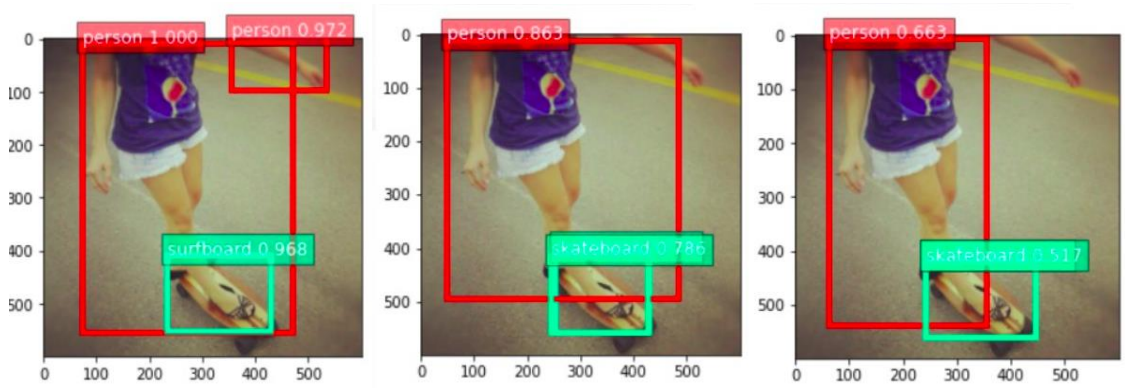


mAP and IOU

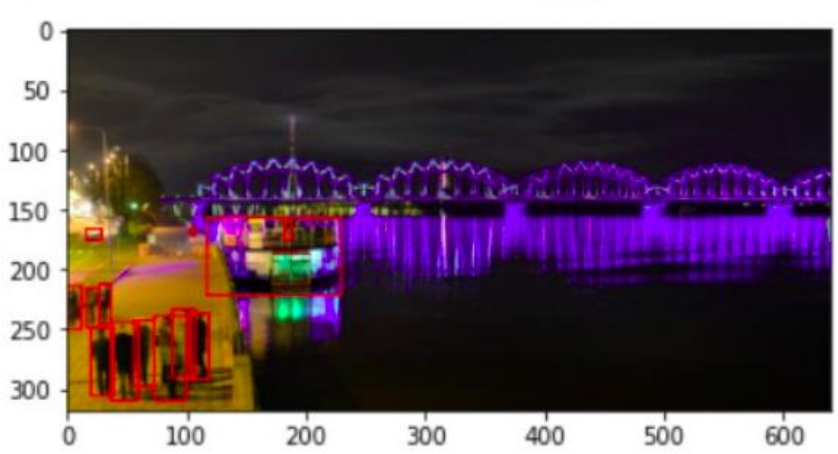
For the whole validation set the **mAP = 0.27 and IOU = 0.323**

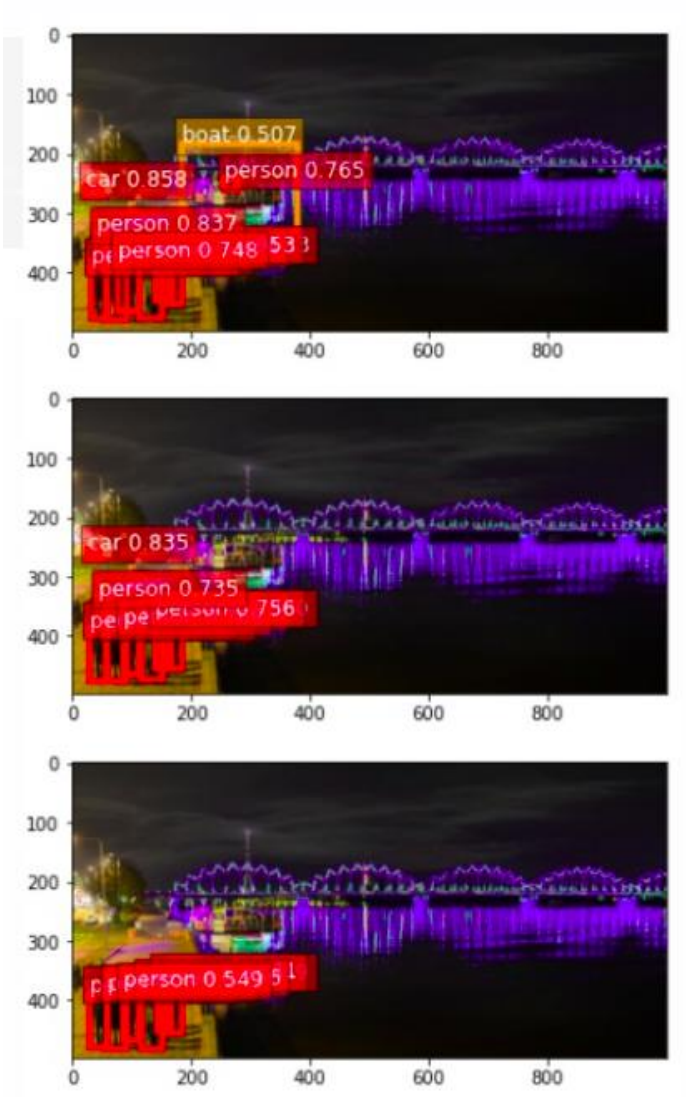
Test Models on Random Images

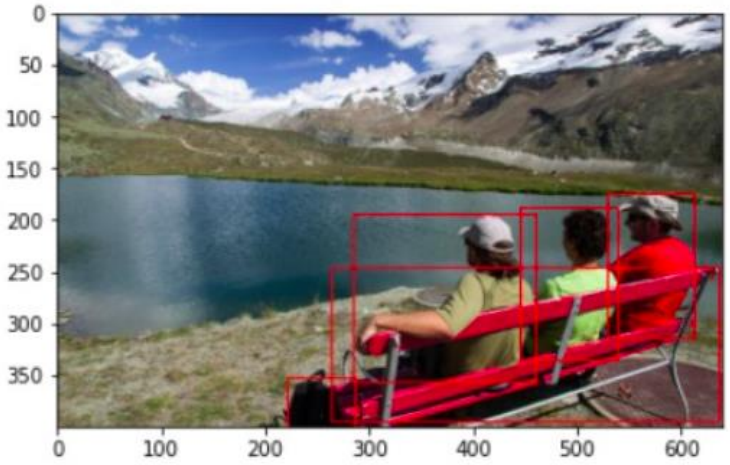


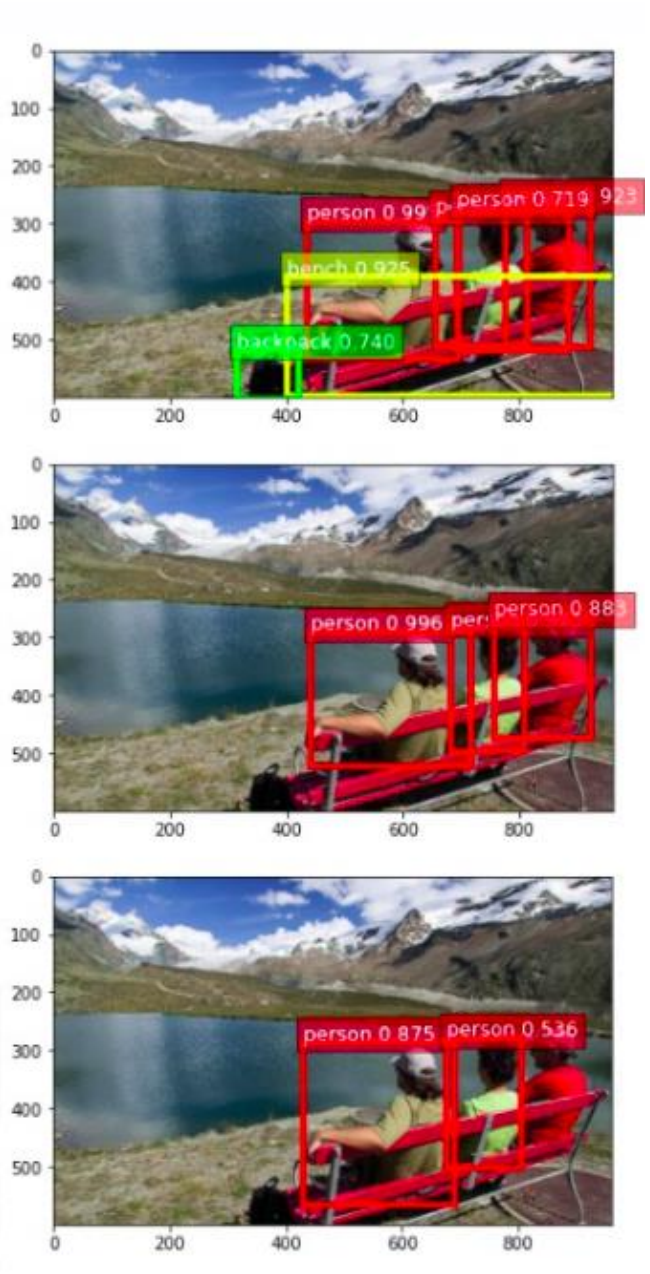


Note: Faster R-CNN gets better confidence and good localization, the SSD has some overlapping but detected all the objects with accepted confidence, Retina has good localization but low confidence.

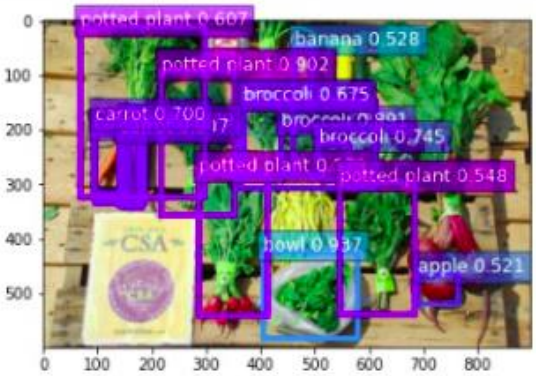


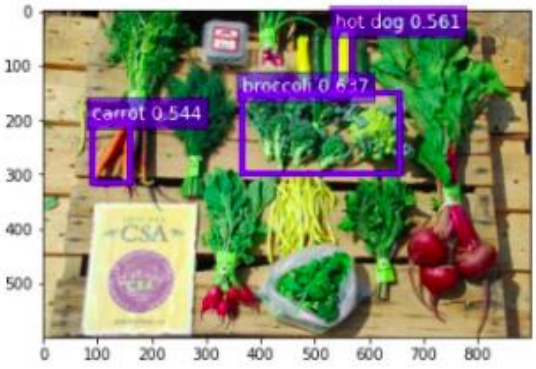

boat 0.507
car 0.858
person 0.765
person 0.837
person 0.748
car 0.835
person 0.735
person 0.549



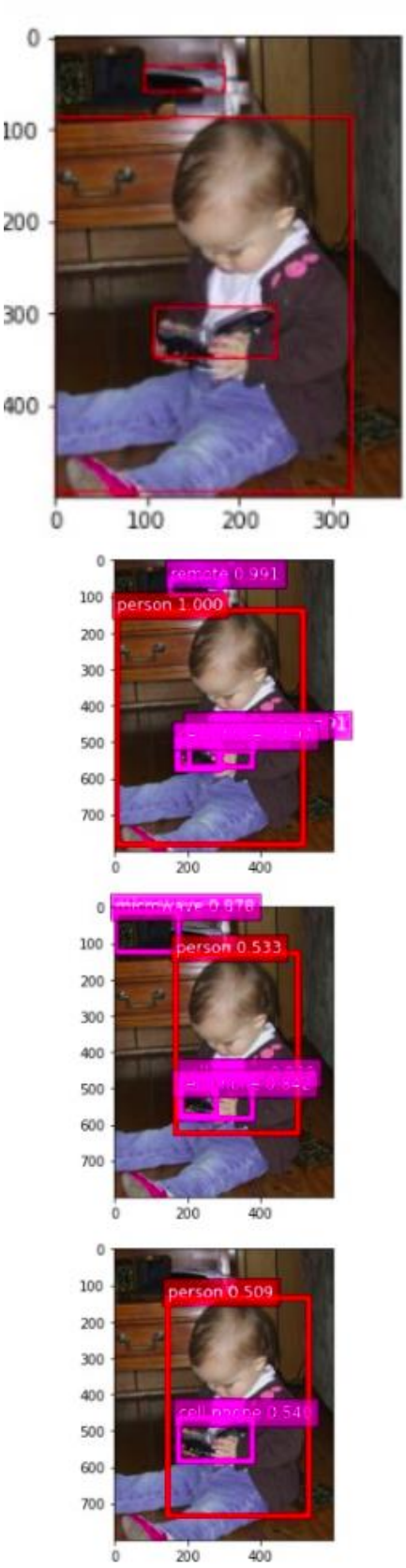

person 0.99
person 0.719
bench 0.925
backpack 0.740
person 0.996
person 0.875
person 0.536


CSA


potted plant 0.607
banana 0.528
potted plant 0.902
broccoli 0.675
carrot 0.700
broccoli 0.745
potted plant 0.548
bowl 0.937
apple 0.521
CSA


hot dog 0.561
broccoli 0.647
carrot 0.544
CSA


carrot 0.557
bowl 0.617
CSA



Note: Overall Faster R-CNN is more accurate while SSD and RetinaNet are faster which gives more convenient in real-time detection.
But for localization and precision Faster R-CNN gives better results.

Comparison

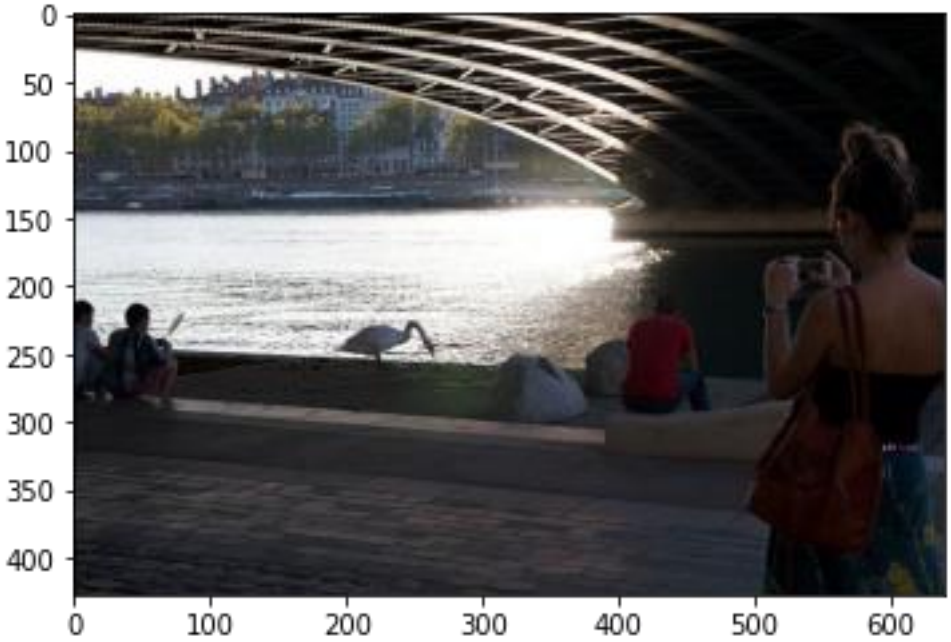
	Coco-RCNN	COCO-SSD	COCO-Retina	VOC-RCNN	VOC-SSD	VOC-Retina
mAP	0.344	0.182	0.297	0.348	0.273	0.351
IOU	0.406	0.221	0.368	0.252	0.323	0.361

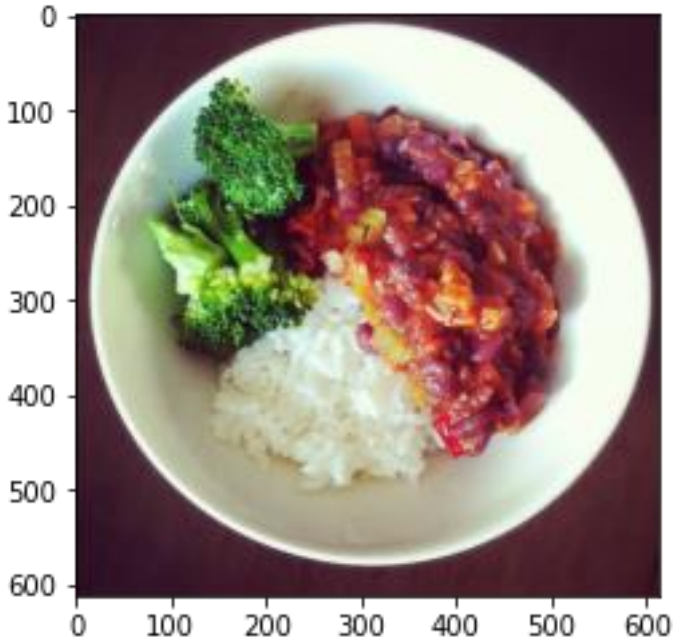
Notebooks

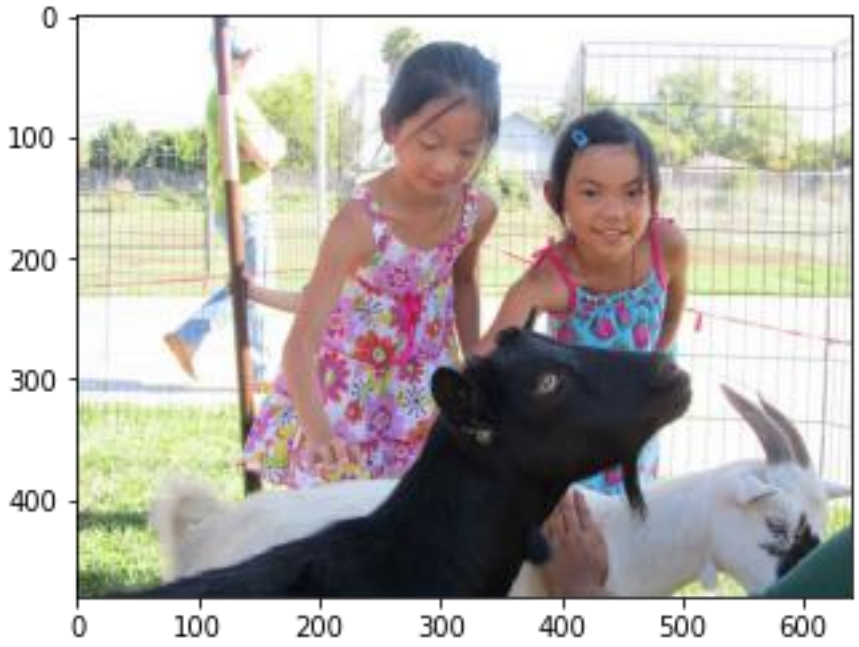
1. Coco dataset:
   1.1. https://colab.research.google.com/drive/1BQKgLPMClvozxpPFRUFEJgwW6EJf8NbK?usp=sharing
   1.2. https://colab.research.google.com/drive/1N5uDqRMsunGsCu6a_OeGhpRYDISunfnK?usp=sharing
   1.3. https://colab.research.google.com/drive/1ovCPIEO4hiVo4XrmjdDYdLe_ivHLP6Hk?usp=sharing#scrollTo=zS8OVB8p3lgz
2. Voc dataset:
   2.1. https://colab.research.google.com/drive/1YaWDKbePhhfOkMSyz9kB_K4KlAQwgNWP?usp=sharing
   2.2. https://colab.research.google.com/drive/18XnoypdZQbbn98sQQkuPXUUYgonaFmgu?usp=sharing
   2.3. https://colab.research.google.com/drive/1DCJSAOzldC4O02EFEqwa0JSBkwOy_Ruv?usp=sharing

Feature Maps

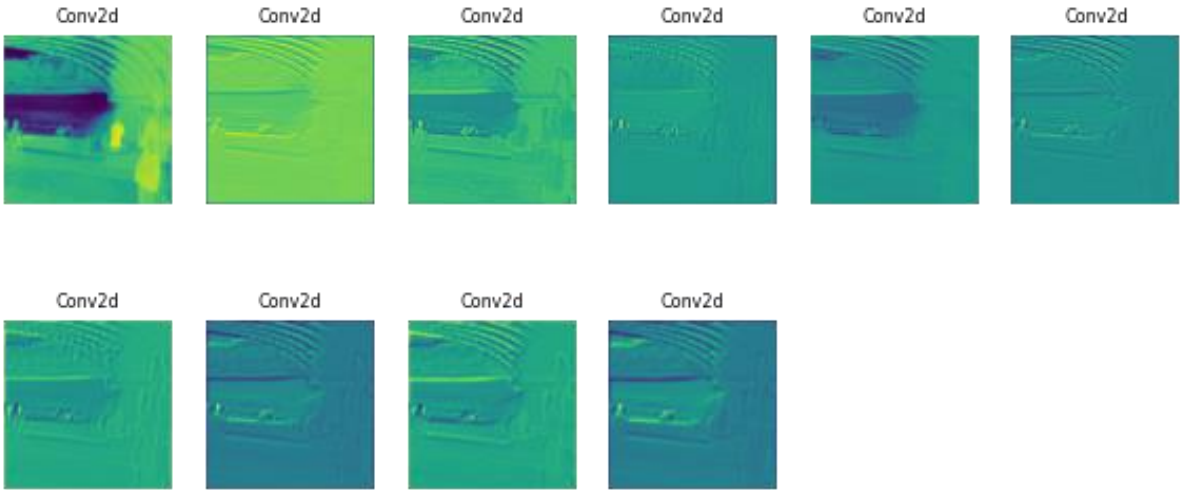
Image samples

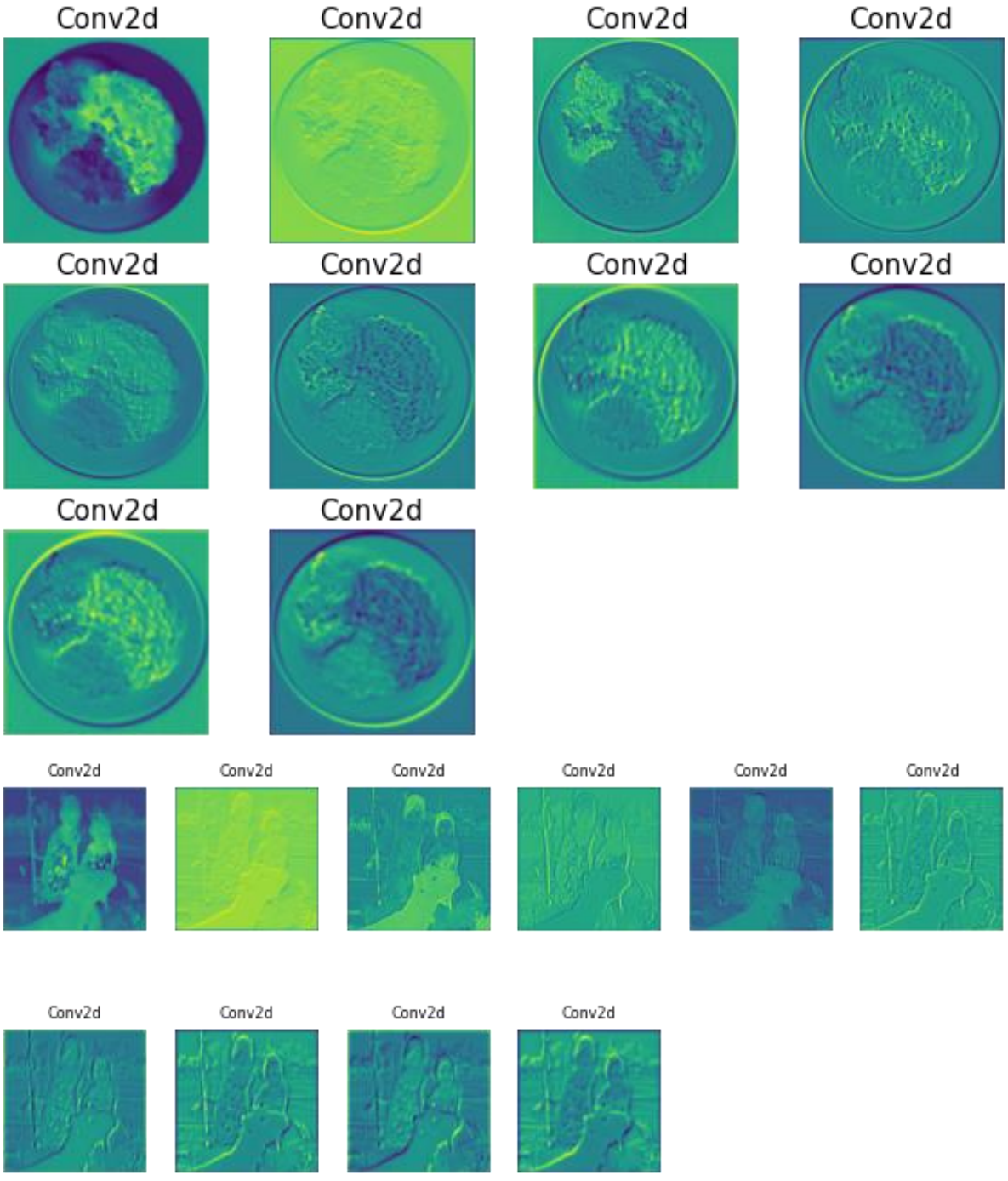




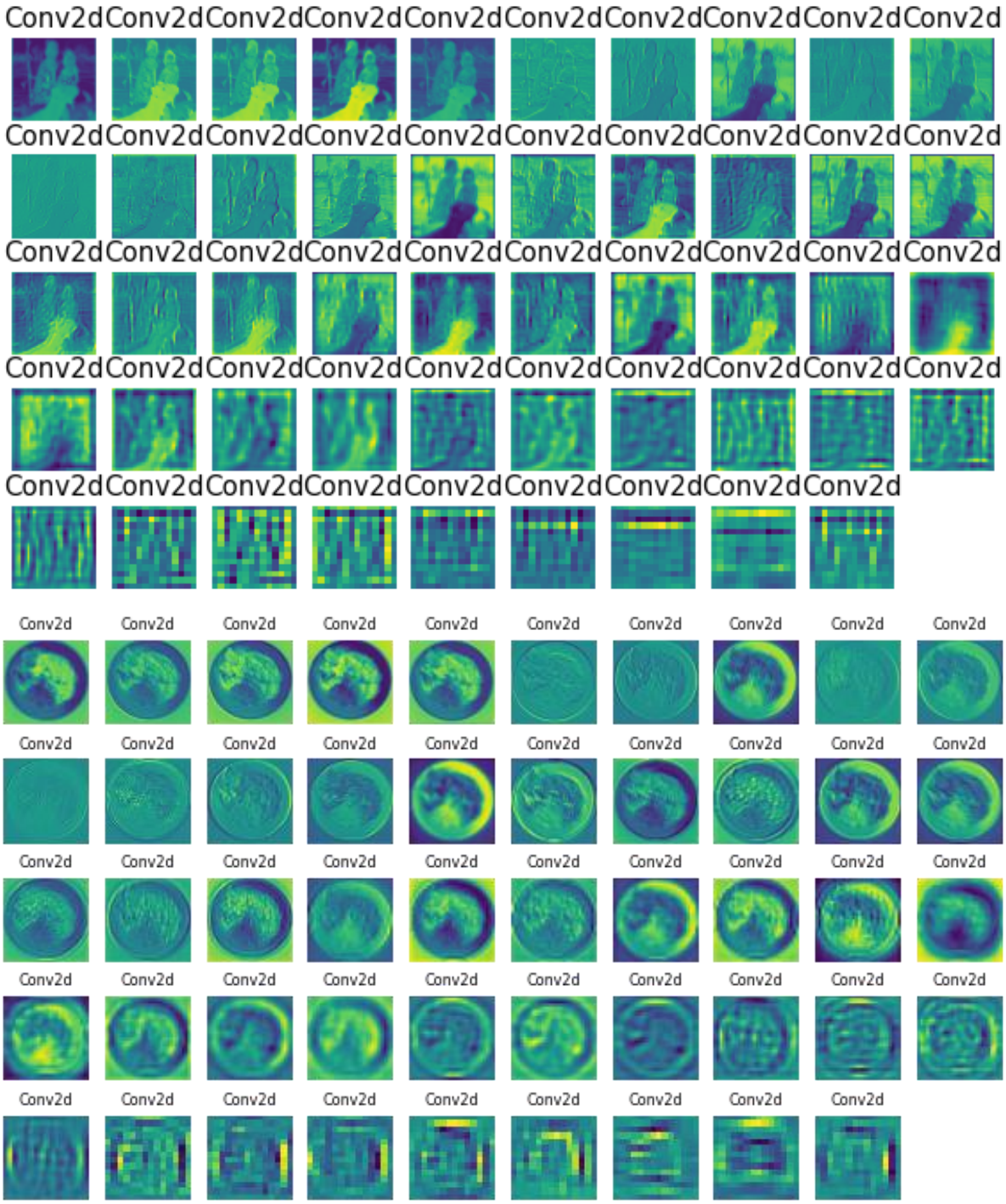


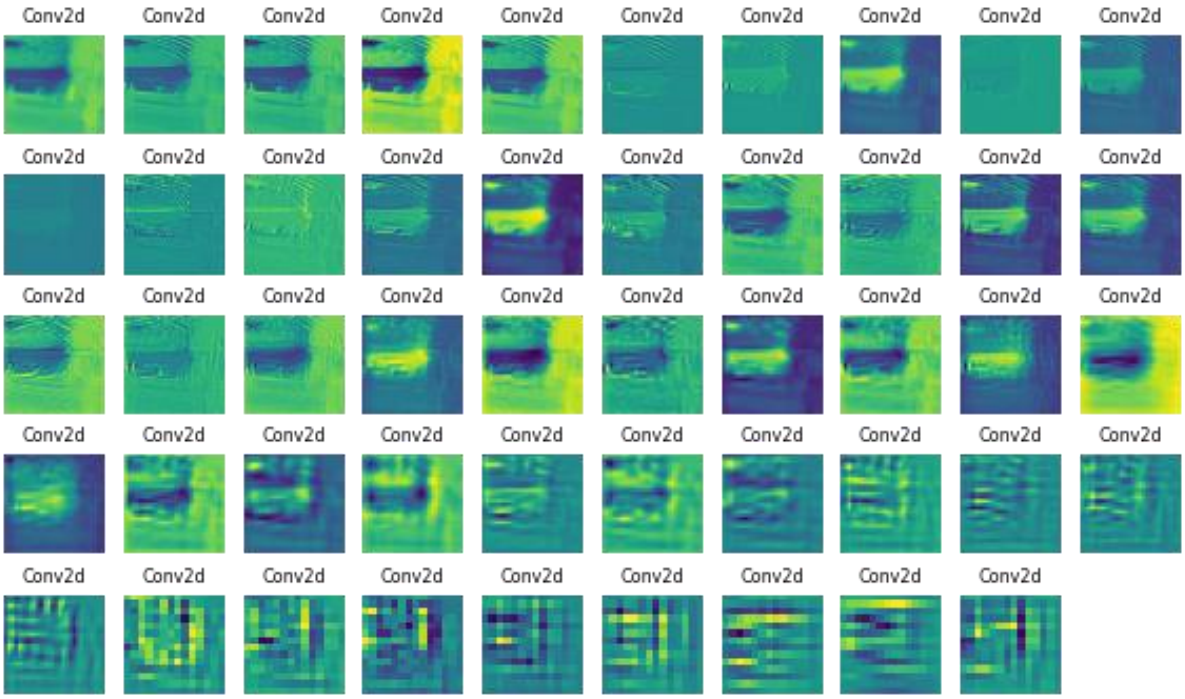
ssd_model.backbone



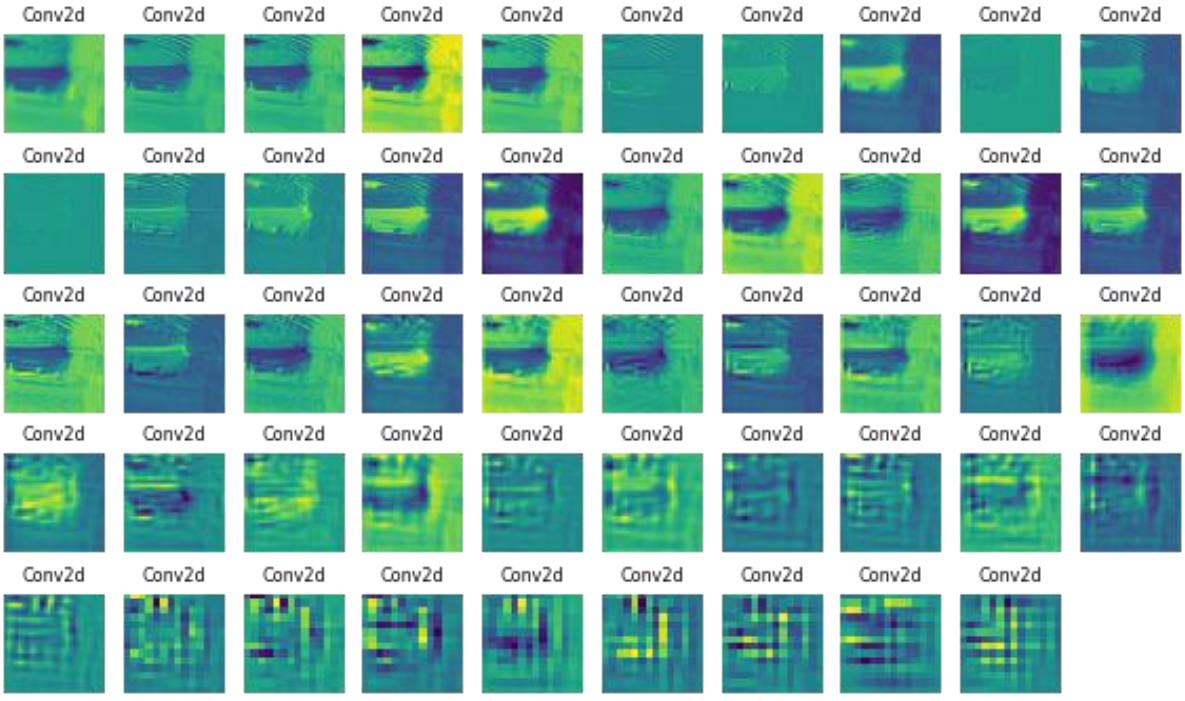

Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d
Conv2d

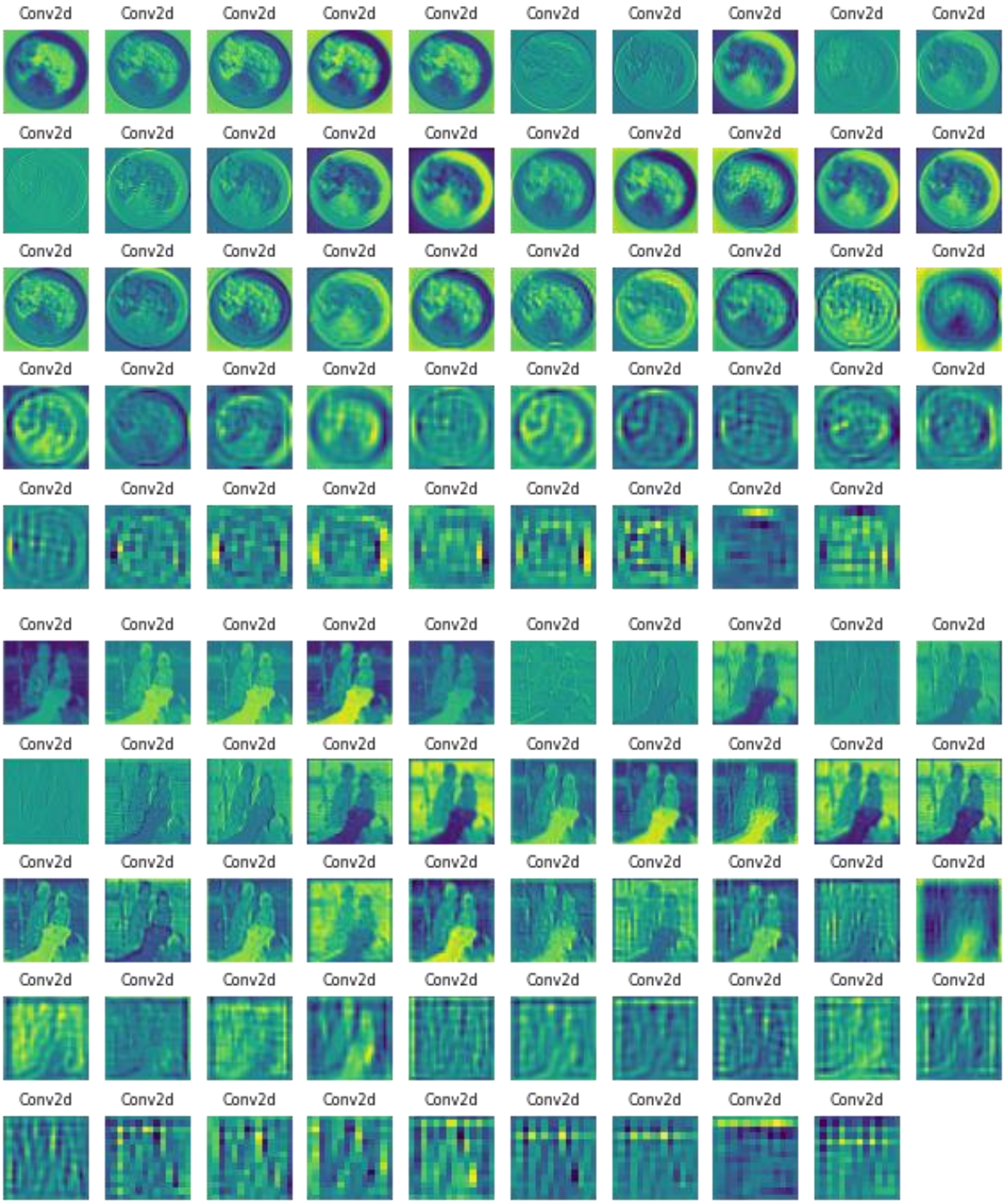
rcnn_model.backbone





retina_model.backbone





Notebook:

https://colab.research.google.com/drive/1i6pggryGTpJXUmwYYYh4mE3upPdQFdox?usp=sharing#scrollTo=LVv2vIuXfaN3

Latex Report

```
\documentclass{article}

\begin{document}

\begin{titlepage}
   \begin{centre}
      \huge{CV Assignment 4}\\
      [2cm]
      \text{\LARGE Name 1: Abdelrahman Salem Mohamed}\\
      [0.5cm]
      \text{\LARGE ID 1: 6309}\\
      [0.5cm]
      \text{\LARGE Name 2: Reem Abdelhalim}\\
      [0.5cm]
      \text{\LARGE ID 2: 6114 }\\
      [0.5cm]
   \end{centre}
\end{titlepage}


\section{Object Detection}
\text In this assignment, you will work on Cocco dataset which is a large-scale object detection,
\text segmentation, and captioning dataset. You are required to run 3 different object detectors on
\text this dataset. We will learn to use and differentiate between the architectures.
\text Either PyTorch or TensorFlow are allowed to run your models.

\noindent
\section{Dataset}
\text We test models using coco 2017 validation set which consists of 5000 different images with their meta data from segmentation points and object bounding boxes.

\section{Models}
\text We used three different models:
\begin{enumerate}
\item  Faster R-CNN (Two stage Model)
\item  RetinaNet (One stage Model)
\item  Single shot detector (One stage Model)
\end{enumerate}

\subsection{ Faster R-CNN }
\paragraph{Discussion}
\text Two stage based model where, first the model learn how to extract Region of proposals, then for every Region of proposal fe perform the normal Fast R-CNN network
```

from ROI pooling then from fully connected layer compute the two losses one for object box (regression), and one for class type (classification).
\text Each network has its own loss functions and the whole system weights consider the both losses path, and during the test the network needs first to identify the ROI and apply Object detection technique, so Faster R-CNN consider slower than one stage models but have better mAP.
\text The backbone network is usually a dense convolutional network like ResNet or VGG16

\subsection{ RetinaNet }
\paragraph{Discussion}
\text RetinaNet is one of the best one-stage object detection models that has proven to work well with \textbf{dense and small scale objects}. For this reason, it has become a popular object detection model to be used with aerial and satellite imagery.

\text There are four major components of a RetinaNet model architecture :
\text a) Bottom-up Pathway - \textbf{The backbone network (e.g. ResNet)} which calculates the feature maps at different scales, irrespective of the input image size or the backbone.
\text b) Top-down pathway and Lateral connections - The top down pathway upsamples the spatially coarser feature maps from higher pyramid levels, and the lateral connections merge the top-down layers and the bottom-up layers with the same spatial size.
\text c) Classification subnetwork - It predicts the probability of an object being present at each spatial location for each anchor box and object class.
\text d) Regression subnetwork - It regresses the offset for the bounding boxes from the anchor boxes for each ground-truth object.

\subsection{ Single Shot Detector}
\paragraph{Discussion }
\text SSD has two components: a backbone model and SSD head.
\text \textit{Backbone} model usually is a pre-trained image classification network as a feature extractor. This is typically a network like \textbf{ResNet trained on ImageNet} from which the final fully connected classification layer has been removed. We are thus left with a deep neural network that is able to extract semantic meaning from the input image
\text The \textit{SSD head} is just one or more convolutional layers added to this backbone and the outputs are interpreted as the \textbf{bounding boxes and classes of objects in the spatial location of the final layers activations}.

\section{Test Models on Random Images}

\text Note: Faster R-CNN gets better confidence and good localization, the SSD has some overlapping but detected all the objects with accepted confidence, Retina has good localization but low confidence.
\text Note: Overall Faster R-CNN is more accurate while SSD and RetinaNet are faster which gives more convenient in real-time detection.
\text But for localization and precision Faster R-CNN gives better results.

\section{Comparison}

\begin{tabular}{|p{0.6in}|p{0.6in}|p{0.6in}|p{0.6in}|p{0.6in}|p{0.6in}|p{0.6in}|} \hline

```
 & Coco-RCNN & COCO-SSD & COCO-Retina & VOC-RCNN & VOC-SSD & VOC-Retina \\
\hline
mAP & 0.344 & 0.182 & 0.297 & 0.348 & 0.273 & 0.351 \\ \hline
IOU & 0.406 & 0.221 & 0.368 & 0.252 & 0.323 & 0.361 \\ \hline
\end{tabular}

\end{document}
```

Link for the zip file:

https://drive.google.com/drive/folders/19Lwy4htajEA9VhwZxqQvD6hG5cpIjrHK?usp=share_link