# Object Detection in Computer Vision

Archana Das[1], Neelam Somai[2], Faizan Sayyad[3] , Kona Sireesha[4] , Jaymeet Mehta[5]

[1]PhD Scholar , Bennett University, Greater Noida, India

[2]*Student, Dept. of computer Engineering, Vivekanand Education Society 's Institute of Technology, Maharashtra,India.*

[3]*Student,Dept. of computer Engineering, Anjuman-I-Islam's M.H.Saboo Siddik College Of Engineering,Maharashtra,India*

[4]*Student, Dept of computer Engineering,RVR&JC College Of Engineering,Andhra Pradesh,India.*

[5]*Student, Dept of Information and Technology,,Chandubhai S Patel Institute of Technology,Gujarat,India.*

***Abstract - Probabilistic Object Detection in computer vision helps to locate and identify objects in video or image. It is widely used in computer vision tasks such as face recognition, face detection , autonomous self driving cars and many more. This paper aims to demonstrate object detection using a single shot multibox detector (SSD) and you look only once(YOLO) . The paper gives the brief idea about these two object detection algorithms , their accuracies and performances. The comparison has been developed between these two algorithms in order to understand which algorithm is faster and accurate. The accuracy obtained for individual objects using these two algorithms has also been described.***

***Keywords - SSD , YOLO , Object detection , CNN , COCO.***

## 1. Introduction

Humans can easily identify and detect the objects present in an image. Humans are very fast and accurate at identifying multiple objects and performing multiple tasks. So,humans can easily train the computers to classify and detect multiple objects within an image with high accuracy by having large amounts of data, faster GPUs, and appropriate algorithms.Humans train the computer for three tasks.They are image classification,object localization and object detection.

Image classification is nothing but assigning class labels (person,animal,things)to image. It classifies the image into different class labels.object localization separates the objects by drawing bounding box.object detection combines the above two tasks.

But it is quite hard to distinguish the difference between object localization and detection.Because all these three come under object recognition.To safely operate in the real world, robots need to evaluate how confident they are about what they see around . A new challenge in computer vision algorithms to not just detect and localize objects, but also report how certain they are. Object detection is often an important part of the perception system of robots or autonomous systems such as driverless cars. It provides crucial information about the robot's surroundings and has significant influence on the performance of the robot in its environment. For example, driverless cars need object detection to be aware of other cars, pedestrians, cyclists and other obstacles on the road. Future domestic service robots and robots in healthcare will have to be able to detect a large range of household objects in order to properly fulfil their tasks.

## 2. Related Work

This section provides the literature survey of the work done in the object detection field in the past few decades. Developers had started working on it since early 1958, but due to very poor processing speed and a deficient amount of storage space for large datasets it takes a long gap until most powerful Viola-Jones Algorithm comes in 2001, which uses Haar-Like features, Cascading and Ada-boost to detect faces[3]. The modern steps of object detection go along with the improvement of Convolutional Nets (CN) which began in 2012 when Alex-Net won the 2012 image net large-scale visual recognition competition

(ILSVRC). CN is utilized as image feature mapping. Alex-net based on old Le-Net along with data augmentation, ReLU, and GPU implementation[4]. Girshick Ross introduced Region Based ConvNet (RCNN) which is a natural combination of heuristic region proposal method and CN feature extractor. The Alex-Net and support vector machine (SVM) model is then trained to classify the object [5]. The overfeat method is introduced by Sermanet Pierre which uses Alex-Net to extract features at
multiple evenly-spaced square windows in the image over multiple scales of an input image [6]. ZFNet is the ILSVRC 2013 winner, which is basically an Alex-Net with minor modification [7]. Spatial pyramid pooling net (SPPNet) is an enhanced version of RCNN by introducing two important concepts: adaptive-sized pooling and computing feature volume only once [8]. For scalable and high-quality object detection Multi-box method is introduced which is not an object recognition but a CN based object proposal method [9]. GoogLeNet (inception) is the winner of ILSVRC 2014 in which instead of using traditional conv and max-pooling layers, it stacks up inception modules [10]. Fast RCNN is SPPNet with trainable feature extraction network and region of interest pooling in replacement of the SPP layer [11].Significant changes take place in the field of object detection when you only look once (YOLO) is introduced which is a direct development of the multi-box method. It turns multi-box from object proposal solution to an object recognition method by adding a soft-max layer [12]. Faster RCNN is the
modification to Fast RCNN in which heuristic region proposal is replaced by the region proposal network (RPN) inspired by
multi-box [13]. Single Shot Multibox Detector was introduced in 2016 which can do everything in a single shot, it just has to look at the image once it does not have to go back to the same image. It does not have to do the object proposal and does not have to run many convolutional layers which reduces the time and computational cost [14].

## 3. Methodology

### 3.1 Dataset

The dataset used for training the models is Microsoft Common Objects in Context ( MS COCO).[1]COCO dataset is an excellent object detection dataset that has 80 different classes, 80,000 training images and 40,000 validation images available.

### 3.2 Algorithms

**Convolutional Neural Network:**

A Convolutional Neural Network (CNN) is a deep learning algorithm that can recognize and classify features in images for computer vision. It is a multiple layers neural network specifically designed to analyze visual inputs and perform tasks such as image classification, segmentation and object detection.
CNNs, just like neural networks, are made up of neurons with learnable weights and biases. Each and every neuron receives several inputs, takes a weighted sum over them , passes it through an activation function like sigmoid, ReLU, etc and responds with an output. The purpose of activation function is to add some kind of non-linearity property to the function.
There are two main parts to a CNN:

- A convolution tool that splits the various features of the image for analysis.

- A fully connected layer that uses the output of the convolution layer to predict the best description for the image.

A CNN is composed of several kinds of layers:

- Convolutional layer- creates a feature map to predict the class probabilities for each feature by applying a filter that scans the whole image, a few pixels at a time.
- Pooling layer (downsampling)- Scales down the amount of information the convolutional layer generates for each feature and maintains the most essential information (the process of the convolutional and pooling layers usually repeats several times).
- Fully connected input layer- “flattens” the outputs generated by previous layers to turn them into a single vector that can be used as an input for the next layer.
- Fully connected layer- It applies weights over the input generated by the feature analysis to predict the best accurate label.
- Fully connected output layer- Generates the final probabilities to determine a class for the image.

### You Look Only Once(YOLOv3)

You only look once (YOLO) is an object detection system targeted for real-time processing.YOLO divides the input image into an **S×S** grid. Each grid cell predicts an object. YOLOv3 is much faster than SSD while achieving very comparable accuracy. Lets see how YOLO detects the objects in a given image.

First, it divides the image into a 13×13 grid of cells. The size of these 169 cells vary depending on the size of the input. For a 416×416 input size that we used in our experiment , the cell size was 32×32. Each cell is then responsible for predicting a number of boxes in the image.

For each bounding box, the network also predicts the confidence that the bounding box actually encloses an object, and the probability of the enclosed object being a particular class.Most of these bounding boxes are eliminated because their confidence is low or because they are enclosing the same object as another bounding box with very high confidence score. This technique is called non-maximum suppression.[2] The authors of YOLOv3, Joseph Redmon and Ali Farhadi, have made YOLOv3 faster and more accurate than their previous work YOLOv2. YOLOv3 handles multiple scales better. They have also improved the network by making it bigger and taking it towards residual networks by adding shortcut connections.

### Single Shot Detector(SSD)

SSD is designed for real-time object detection . SSD is faster than Faster R-CNN
due to some improvement which includes multi-scale features and default boxes.SSD uses low resolution images which helps to increase the speed further.

SSD object detection is made up of two components:
1. Extracting features maps
2.Convolution filters to detect objects

SSD uses VGG16 to extract feature maps. The conv4-3 detects objects.For example, we draw the Conv4-3 to be 8 × 8 spatially (it should be 38 × 38). For each cell (also called location), it makes 4 object predictions.Each prediction consists of a boundary box and 21 scores for each class and we pick the one with the highest score and assign the class to the bounding box.

There are 38*38*4 predictions that is 4 predictions per cell regardless of the depth of the feature maps. As expected, many predictions contain no object. ‘0’ is reserved to denote no object by SSD.
Class and location are both computed using small convolution filters in SSD.After extracting the feature maps, SSD applies 3 × 3 convolution filters for each cell to make predictions.The output of each filter are 25 channel i.e 21 scores for each class and one for bounding box.

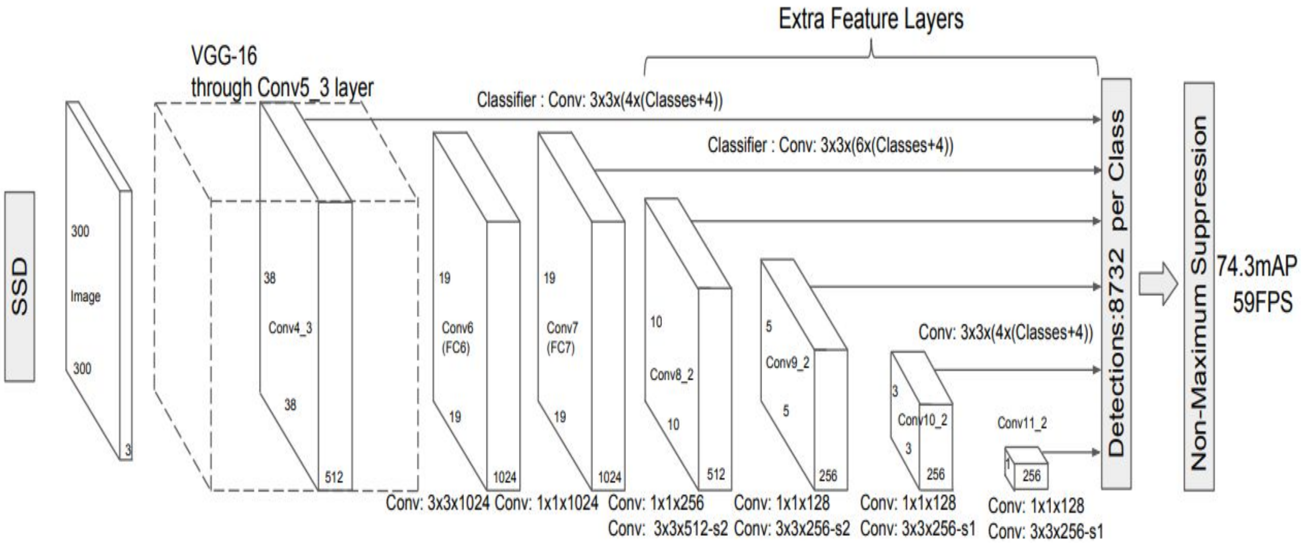
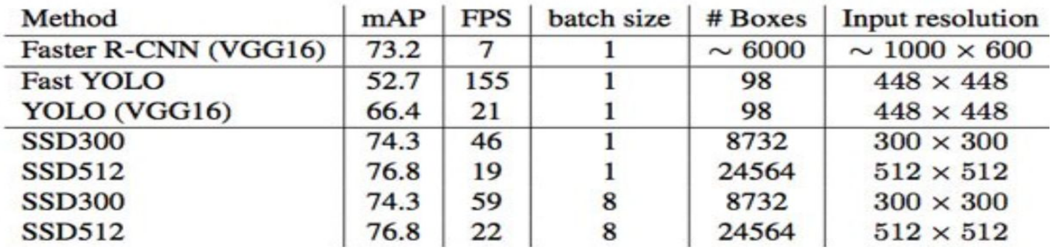


**Fig. 1 Single Shot Detector Architecture(source:https://arxiv.org/pdf/1512.02325.pdf)**

.Multi-scale feature maps for detection

SSD uses multiple layers (multi-scale feature maps) to detect objects independently. When CNN reduces the spatial dimension, the resolution of the feature maps also decreases.SSD uses layers of low resolution to detect large scale objects. For example, the 4× 4 feature maps are used for larger scale objects.SSD adds more auxiliary convolution layers(six) after the VGG16. Out of the six, five are added for object detection.We make 6 predictions rather than 4 from three of the layers.SSD makes about 8732 predictions using six layers. Multi-scale feature maps improve accuracy significantly.The predictions made by SSD are classified as positive and negative matches.From above option SSD consider only positive matches for calculating the cost of localization. If the intersection over union (IoU) is greater than 0.5 corresponding to the default boundary box with ground truth then the match is positive or else it is negative.The total prediction made by the SSD is 8732 for some better coverage of location,scale and aspect ratio ,more than many other detection methods.Some predictions contain no object. So any prediction which has class confidence score less the 0.01 will be discarded.

**Table 1. Accuracies with different object detection models**



| Method | mAP | FPS | batch size | # Boxes | Input resolution |
|---|---|---|---|---|---|
| Faster R-CNN (VGG16) | 73.2 | 7 | 1 | $\sim 6000$ | $\sim 1000 \times 600$ |
| Fast YOLO | 52.7 | 155 | 1 | 98 | $448 \times 448$ |
| YOLO (VGG16) | 66.4 | 21 | 1 | 98 | $448 \times 448$ |
| SSD300 | 74.3 | 46 | 1 | 8732 | $300 \times 300$ |
| SSD512 | 76.8 | 19 | 1 | 24564 | $512 \times 512$ |
| SSD300 | 74.3 | 59 | 8 | 8732 | $300 \times 300$ |
| SSD512 | 76.8 | 22 | 8 | 24564 | $512 \times 512$ |

## 4. Experimental Results

The results obtained with yolov3 and SSD are shown below:
YOLOv3 has better results for our test images than SSD. The certainty of objects in the image obtained with YOLOv3 are higher. In terms of speed also YOLOv3 performed better than SSD. It took less than 12 seconds to detect 12 objects in the image shown below. The model with single shot detector could detect 6 persons in image with the highest certainty of 91% whereas model trained using Yolov3 algorithm can detect 8 people in an image with certainty of 100%. Similarly, for Kite as an object SSD model could detect 4 kites with highest certainty of 82% and Yolo model can detect 3 kites with 100% certainty.

**SSD:**

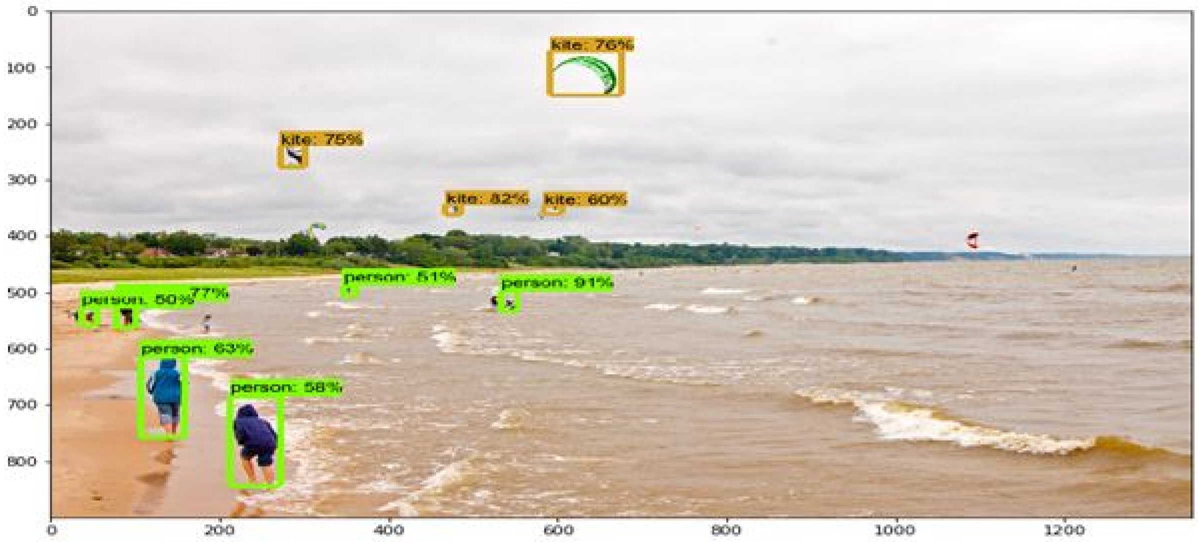


**Fig. 2 Object detection using SSD**

**YOLOv3:**

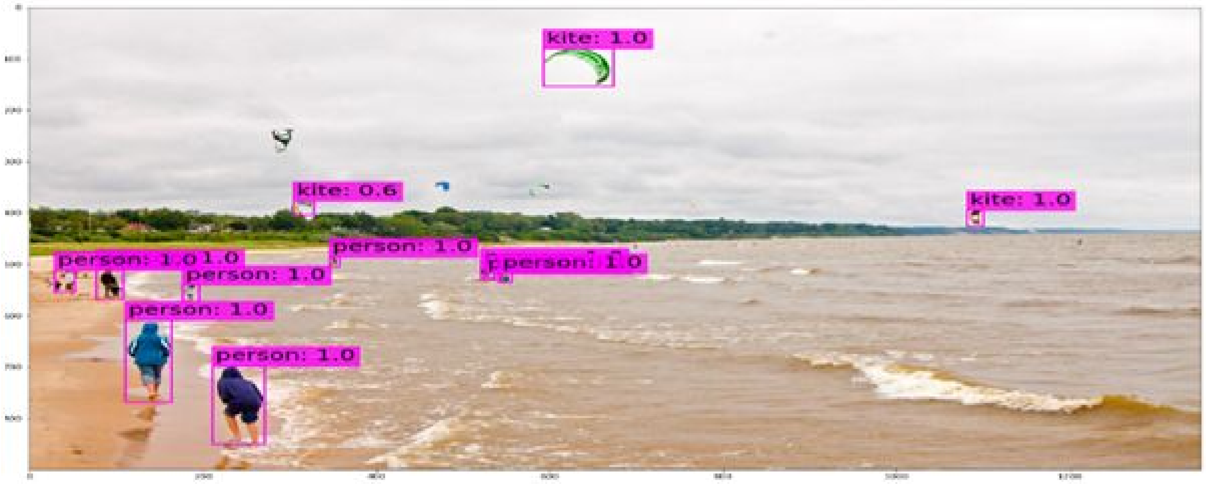
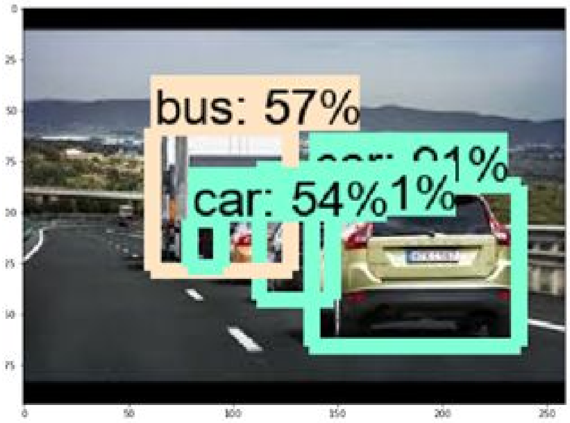
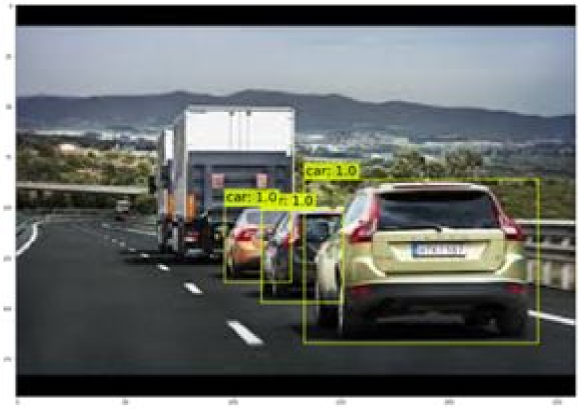


**Fig.3 Object detection using YOLOv3**

When it comes to objects like vehicles Yolov3 can easily detect cars in an image but missed bus in an image whereas SSD trained model could detect both but with less probability of that object.

| SSD | YOLOv3 |
|---|---|
| <br> bus: 57%<br>car: 91%<br>car: 54%1%<br>0<br>50<br>100<br>150<br>200<br>250<br>**Fig.4 Objection detection For Cars and Buses using SSD** | <br> <br>**Fig.4 Objection detection For Cars and Buses using YOLOv3** |

Trying it with different images and videos following are the analysis of success rate and miss rate of different objects with YOLOv3 and SSD.

**Table 2. Accuracy Analysis**

| Result Analysis for YOLOv3 and SSD | | | | |
|---|---|---|---|---|
| **Object Class** | **Success rate of YOLOv3** | **Miss Rate of YOLOv3** | **Success rate of SSD** | **Miss Rate of SSD** |
| Person | 97% | 3% | 95% | 5% |
| Kites | 43% | 57% | 57% | 43% |
| Cars | 98% | 2% | 96% | 4% |

Similarly, analysis of accuracies with different objects can be done . Thus, summarizing YOLOv3 performed better in terms of speed and accuracy for different test images.

## 5. Conclusions

An efficient and accurate object detection system has been developed which achieves comparable measures with the existing state-of-the-art system. This project is developed by using recent techniques

in the field of deep learning and computer vision.And the evaluation was consistent.This is useful in real-time applications which require detection of the object for pre-processing in their pipeline.
we will need object detection systems for robots (which will explore some areas that have not been seen by humans). And it will be used to see depth areas of the sea or other planets.

## 6. Acknowledgement

We would like to thank Bennett University for providing us this wonderful opportunity to work on a project Probabilistic Object Detection, also we would like to thank Ms. Archana Das our project mentor for guiding us throughout this project.

***References***

[1] T. Lin, M. Maire, S. Belongie,L. Bourdev, R. Girshick, J. Hays, D. Ramanan, C. Zitnick and P. Dollar,"Microsoft COCO: Common Objects in Context",*arXiv:1405.0312v3* , Feb 2015.
[2] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement", *arXiv:1804.02767v1*, Apr 2018.
[3] P. Viola & M. Jones. "Rapid Object Detection using a Boosted Cascade of Simple Features," *Conference on Computer Vision and pattern recognition*, 2001.
[4] Krizhevsky, Alex, I. Sutskever, and G. Hinton. "Imagenet classification with deep convolutional neural networks",*NIPS Conference ,2012.*
[5] R. Girshick, J. Donahue, T. Darrell, J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation,"*The IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), pp. 580-587,2014.
[6] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. LeCun, "OverFeat: Integrated Recognition, Localization, and Detection using Convolutional Networks," *Cornell University Library*, 21 Dec 2013.
[7] Zeiler M.D., Fergus R. (2014)" Visualizingand Understanding Convolutional Networks". In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) *Computer Vision – ECCV* 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8689. Springer, Cham.
[8] K. He, X. Zhang, S. Ren, J. Sun," Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition,*"Cornell University Library*,18 June 2014.
[9] C. Szegedy, S. Reed, D. Erhan, "Scalable High-Quality Object Detection," *Cornell University Library,* 9 dec 2015.
[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, " Going Deeper with Convolutions," *Cornell University library*, 17 Sep 2014.
[11] R. Girshick, "Fast R-CNN," *Cornell University Library*, 30 Apr 2015.
[12] J. Redmon, S. Divvala, R. Girshick, A. Farhadi," YouOnly Look Once: Unified, Real-Time Object Detection," *Cornell University Library*, 8 June 2015.
[13] S. Ren, K. He, R. Girshick, J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,"
*Cornell University Library*, 4 June 2015.
[14] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in *European Conference on Computer Vision(ECCV)*, 2016.