iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** III **Month of publication:** March 2024

DOI: https://doi.org/10.22214/ijraset.2024.59632

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue III Mar 2024- Available at www.ijraset.com

Analysis of Object Detection Models

Damini[1], Aman Kumar Sharma[2]
Department of Computer Science, Himachal Pradesh University, India

Abstract: ***Object detection plays a crucial role in today's world, and it gives direction in the field of computer vision. Object detection is a fundamental task used to detect the object in the images. Object detection plays a significant role in various applications such as surveillance, medical images, and autonomous vehicles. The primary goal of object detection is to accurately detect the boundaries of objects of interest and classify those objects into predefined categories. This paper analyzes three common object detection models: Single Shot Multibox Detector (SSD), You Only Look Once (YOLOv7), and You Only Look Once (YOLOv8). In this research work, both empirical and theoretical approaches were followed. The empirical study was carried out on a set of experiments using some software tools. For the theoretical approach, a review of both secondary data and data based on results obtained by applying the tools is studied. Secondary data was acquired from the books, e-journals, reports, review papers, published theses, websites, articles, conference proceedings, survey papers, and blogs that are related to this domain. The results that were obtained from this implementation were analyzed for the findings of the research. This paper compares three object detection model's results, which are tested on the same dataset, the same environment, and the same system. This paper found that You Only Look Once (YOLOv8) obtained the best results, followed by You Only Look Once (YOLOv7) and Single Shot Multibox Detector (SSD) at last.***

Keywords: Object Detection, You Only Look Once (YOLOv8), You Only Look Once (YOLOv7), Single Shot Multibox Detector (SSD) and Regional Convolutional Neural Network (RCNN), Precision Time, Recall Time Mean Average Time (MAP).

I. INTRODUCTION

During the last years, there has been a rapid and successful expansion on computer vision research [1]. Object detection is important in computer vision systems. It can be used for many applications like video surveillance, medical imaging, and robot navigation. The newest method used for object detection is called convolutional neural networks (CNN) [2]. With the recent development of deep learning, it boosts the performance of object detection task [3]. It can generally be divided into two groups: two stage region-based detectors, which propose object regions in stage one and extract features from these regions in stage two (e.g. Region CNN and its descendants Fast R-CNN, Faster R-CNN, Region FCN, Mask R-CNN), and one stage grid-based detectors, which skip the region proposal step and instead extract features over a dense, static grid of possible object locations in the image (e.g., SSD, YOLO). One stage detectors are faster and suitable for real-time inference [4]. In this study three object detection will be evaluated to check the accuracy. Three object detection algorithms used for evaluation are You Only Look Once (YOLOv8), You Only Look Once (YOLOv7) and Single Shot Multibox Detector (SSD).

II. TAXONOMY

A. Single Shot Multibox Detector (SSD)

In 2016, the SSD model was proposed by Liu. The model uses the regression idea used in the YOLO algorithm and draws on the concept of the anchor box proposed in the Faster R-CNN detection model [5].The SSD approach is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum suppression step to produce the final detections [6]. The single shot multibox detection (SSD) [4] model takes an entire image as input and passes it through multiple Conv layers with different sizes of filters (10×10, 5×5 and 3×3). Feature maps from conv layers at different position of the network are used to predict the bounding boxes. They are processed by a specific conv layers with 3 × 3 filters, called extra feature layers, to produce a set of bounding boxes [7]. Additionally, the idea of ROI is also applied [8]. The classification is applied to these features and then the locations of boxes are adjusted to match the object boundaries more precise. SSD enables multi-scale object detection by taking into consideration of each output of different scaled convolutional layers. The higher dimensional convolution outputs give the feature of smaller regions. Contrary to this, the lower dimensional convolution outputs yield to bigger region areas [9]. During predictions, the model generates the scores in each default box for every object detected and scales the default box to fit the object shape [10]. SSD maintains the accuracy of Fast-RCNN while maintaining the detection speed of YOLO.

©IJRASET: All Rights are Reserved

This is because SSD put regression idea of YOLO and the anchor mechanism of Fast-RCNN in one model and uses multi-scale regions in different positions of the image for regression. The higher resolution layers in the architecture of SSD are responsible for detecting small objects but such layers have some insignificant features which are not useful and are less informative for object detection. SSD speeds up the process by eliminating the need of the region proposal network. To recover the drop in accuracy, SSD applies a few improvements including multi-scale features and default boxes [11].

B. You Only Look Once (YOLOv7)

In the areas of computer vision and machine intelligence, the object detection technique known as YOLO is widely used. A convolutional neural network (CNN) called YOLO separates an input picture into a grid of cells and forecasts the existence, class, and boundary box coordinates of each item within each cell. This approach allows YOLO to process images quickly and accurately without the need for complex post-processing steps [12].YOLOv7 was released in 2022 by Wang et al [13]. The YOLOv7 architecture, an extension of YOLOv4, has been developed with several architectural reforms, including the Extended Efficient Layer Aggregation Network (EELAN) and free trainable tools [14]. The architectural reforms include E-ELAN (Extended Efficient Layer Aggregation Network) along with Model Scaling for concatenation based Models to enhance learning ability [15]. E-ELAN uses expand, shuffle, and merge cardinality to achieve the ability to continuously enhance the learning ability of the network without destroying the original gradient path. E-ELAN only changes the architecture in computational block, while the architecture of transition layer is completely unchanged. YOLOv7 also has model scaling for concatenation-based models. The main purpose of model scaling is to adjust some attributes of the model and generate models of different scales to meet the needs of different inference speeds [16].Coarse for Auxiliary and Fine for Lead Loss: A YOLO architecture, the head contains the predicted model outputs. YOLOv7 has two heads. The lead head which is responsible for the final output, and the auxiliary head which is used to assist training in the middle layers. Batch normalization and EMA model: The batch normalization and Ensemble Mean Teacher (EMA) are two training techniques were carried over from YOLOv4 architecture and used in YOLOv7. The batch normalization integrate the mean and variance of batch into the bias and weight 4/10 of convolutional layer at the inference stage. Mean Teacher improves test accuracy and enables training with fewer labels [17]**.**The backbone module comprises a series of BConv convolution layers, E-ELAN convolution layers, and max-pooling convolution (MPConv)layers. The input module of the YOLOv7 architecture is tasked with the responsibility of resizing the input image to a uniform pixel dimension in order to adhere to the input size requirements of the backbone network. The BConv approach integrates multiple components, including a convolution layer, a batch normalization (BN) layer, and a LeakyReLU activation function, in order to effectively capture image features across different scales. This process enables the transmission of fundamental knowledge from lower tiers to higher tiers, thereby ensuring the efficient incorporation of characteristics across various tiers [18].

C. You Only Look Once (YOLOv8)

YOLOv8 uses a similar backbone as YOLOv5 [19] with some changes on the CSP Layer, now called the C2f module. The C2f module (cross-stage partial bottleneck with two convolutions) combines high-level features with contextual information to improve detection accuracy [20]. This component consists of a pre-trained CNN that is responsible for extracting low-, medium-, and high-level feature maps from the input images. The C2f structure consists of two convolution blocks (this process reduces the number of channels in the feature map usually by half) and three bottleneck convolution blocks [21]. YOLOv8 uses an anchor-free model with a decoupled head to independently process objectness, classification, and regression tasks. This design allows each branch to focus on its task and improves the model's overall accuracy. In the output layer of YOLOv8, they used the sigmoid function as the activation function for the objectness score, representing the probability that the bounding box contains an object. It uses the softmax function for the class probabilities, representing the objects' probabilities belonging to each possible class. YOLOv8 uses CIoU and DFL loss functions for bounding box loss and binary cross-entropy for classification loss. These losses have improved object detection performance, particularly when dealing with smaller objects. YOLOv8 also provides a semantic segmentation model called YOLOv8-Seg model. The backbone is a CSPDarknet53 feature extractor, followed by a C2f module instead of the traditional YOLO neck architecture. The C2f module is followed by two segmentation heads, which learn to predict the semantic segmentation masks for the input image. The model has similar detection heads to YOLOv8, consisting of five detection modules and a prediction layer. The YOLOv8-Seg model has achieved state-of-the-art results on various object detection and semantic segmentation benchmarks while maintaining high speed and efficiency [20]. YOLOv8 provides different size models of N / S / M / L / X scales [22].

III. NEED OF STUDY

Object detection is a versatile and impactful technology that contributes to advancements in numerous fields, improving efficiency, safety, and our ability to understand and interact with the world around us. Object detection is an important task these days. Because this object detection helps the machine see and understand things just like a human. Before object detection, automating tasks was manual and time-consuming. However, nowadays, automating tasks is not done manually with the help of object detection algorithms. Object detection plays a significant role in the security domain. Nowadays, surveillance cameras are able to detect objects, detect suspicious activities, and count and track real-time objects. Object detection is also used in self-driving cars. It heavily relies on object detection. In a self-driving car, object detection detects multiple objects, navigates the road, and avoids collisions. The accuracy of object detection is effective and accurate. In the medical field, where accuracy is crucial, object detection helps to analyze medical images such as MRIs and X-rays with higher speed and accuracy. With the help of this doctor, he can diagnose the patient's diseases and save the patient's life. Now doctors can see tumors in 3D, which helps them operate on the tumor effectively. In the agriculture field, the drone equipment, with the help of an object detection algorithm, frequently identifies ripe crops, detects pests, and even estimates yield. With the help of object detection, farmers can improve harvest efficiency and minimize losses. With the help of object detection algorithms, microscopic images of cells and tissues can be analyzed to identify potential drug targets and accelerate drug discovery. Object detection aids in disaster management by identifying and locating people, buildings, and infrastructure in affected areas. This information is crucial for coordinating rescue and relief efforts. Human-computer interaction scenarios are using object detection, which includes gesture recognition, augmented reality, and virtual reality. It allows devices to understand and respond to the presence and movement of objects, enhancing user experiences. In environmental monitoring, object detection plays a significant role. An object detection algorithm is used to track and analyze changes in ecosystems, wildlife habitats, and natural landscapes. Object detection aids in disaster management by identifying and locating people, buildings, and infrastructure in affected areas. This information is crucial for coordinating rescue and relief efforts during natural disasters. In retail stores and e-commerce platforms, object detection assists in recognizing products, facilitating image-based searches, and providing personalized recommendations to customers. The importance of object detection transcends mere technological advancement. It is becoming a daily part of our lives and enhancing safety and security across various domains. Because of this advancement, object detection is becoming important and widely used in every field.

Object detection algorithms are widely used because of their better accuracy and faster performance. Because now a days we do not compromise with speed and accuracy. Speed and accuracy are the most significant factors in every real-time domain. An object detection algorithm provides a good balance in terms of speed and accuracy. In object detection algorithms, accuracy is measured in terms of mean average precision (mAp). The aim of studying object detection algorithms is to create a meaningful comparison. There are many research papers on two-stage object detection algorithms. However, the research papers on the single-stage object detection algorithm are minimal. By studying object detection algorithms, we will draw a meaning comparison. A comparison study helps for better understanding and correctly understanding the differences between each other.

IV. OBJECTIVES OF STUDY

The specific objectives of the study are listed below:

1) To study object, object detection and computer vision associated with computer vision tool.

2) To empirically identify the three efficient and popular object detection algorithms, which are better for detection of objects in object detection.

3) To evaluate and validate the performance of three object detection algorithms and find out the best object detection algorithms among them.

V. ENVIRONMENT

The analysis is written on Google Colab using python language. Python language version used in the experiment is python 3.12.13. Colab notebook is an application which make us to access the run time GPU. Experiment is conducted using a 64 bit windows 10 pro system, the processor is Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz 2.00 GHz. The dataset [20] is used is COCO dataset from website: https://www.cocodataset.org/ accessed 3/12/2023 at 5:32 PM. The COCO data set containing 100 images. The size of the dataset is 7MB. For implementing object detection Py Torch, Tenser flow is must and some other packages such as Numpy, Matlab etc.

The theoretical approach used to carry out the study included analysis of many research papers based upon object detection algorithms mainly single shot multibox detector (SSD), you only look once (YOLOv7), you only look once (YOLOv8). The analysis of the research papers concluded that all the object detection algorithms have their own set of merits and demerits.

The second approach used is the empirical approach. This approach evaluates and validates the results given by theoretical approach.

For empirical approach experiment was carried out. The experiment was to use for object detection algorithms i.e. Single Shot Multibox Detector (SSD), You Only Look Once (YOLOv7), You Only Look Once (YOLOv8) on the same dataset and using same platform and tools. We used YOLOv8n, YOLOv8 variant "N" i.e. Nano.

A. Dataset

COCO is a large-scale object detection, segmentation, and captioning dataset. The dataset is used is COCO dataset from website: https://www.cocodataset.org/ accessed 3/12/2023 at 5:32 PM. The COCO data set containing 100 images. The size of the dataset is 7MB [23] .

B. Google Colab

Google Colab is a platform which we our used to executed our experiment by using python language. We used python version number 3.12.13 in the experiment. Google Colaboratory, or Colab, is an as-a-service version of Jupyter Notebook that enables you to write and execute Python code through your browser.

Jupyter Notebook is a free, open-source creation from the Jupyter Project. A Jupyter notebook is like an interactive laboratory notebook that includes not just notes and data, but also code that can manipulate the data. The code can be executed within the notebook, which, in turn, can capture the code output [24].

C. Metrics

These are the matrix which we are decide to evaluate the three algorithms:

1) Precision: Precision measures the proportion of accurately categorized positive samples (True Positive) to the total number of positively classified sample (either correctly classified or not, True Positive + False Positive).
2) Recall: The recall value is calculated by taking ration of True Positive to all Positive samples (True Positive + False Negative). It measures how well the model can identify positive samples.
3) Accuracy: Accuracy measures how accurately our model can detect objects.
4) Mean Average Precision (mAP):

The mAP@0.5 calculates a score by comparing the detected box to the ground-truth box bounding box at IOU threshold of 0.5. The model's detections are the more precise, the higher the score [16]. Mean Average Precision (mAP) averages the precision and recall scores for each object class to determine the overall accuracy of the object detector [25].

VI. RESULT ANALYSIS

100 images were tested on Google Colab by the object detection models, i.e. SSD, YOLOv8, and YOLOv7. Metrics used to test the performance of these models are recall time, accuracy, precision time, and map. To present 100 images, as depicted in fig 1. (A), fig 1. (B), and fig 1. (C), each figures consist of 100 images. Fig 1(a) Depicted the recall time of resulted images i.e. 1 to 33 images.

Fig1. (a) Shows the recall time of image_8 by YOLOv8n, YOLOv7, and SSD. The recall time of image_8 by YOLOv8n i.e. 1, recall time of image_8 by YOLOv7 i.e. 0.2 and recall time of image_8 by SSD i.e. 0.7.

Fig1. (a) Shows the recall time of image_17 by YOLOv8n, YOLOv7, and SSD. The recall time of image_17 by YOLOv8n i.e. 1, recall time of image_17 by YOLOv7 i.e. 1 and recall time of image_17 by SSD i.e. 0.5.

Fig1. (a) Shows the recall time of image_23 by YOLOv8n, YOLOv7, and SSD. The recall time of image_23 by YOLOv8n i.e. 1, recall time of image_23 by YOLOv7 i.e. 0.66 and recall time of image_23 by SSD i.e. 0.60.

Fig1. (a) Shows the recall time of image_30 by YOLOv8n, YOLOv7, and SSD. The recall time of image_30 by YOLOv8n i.e. 1, recall time of image_30 by YOLOv7 i.e. 0.55 and recall time of image_30 by SSD i.e. 0.50.

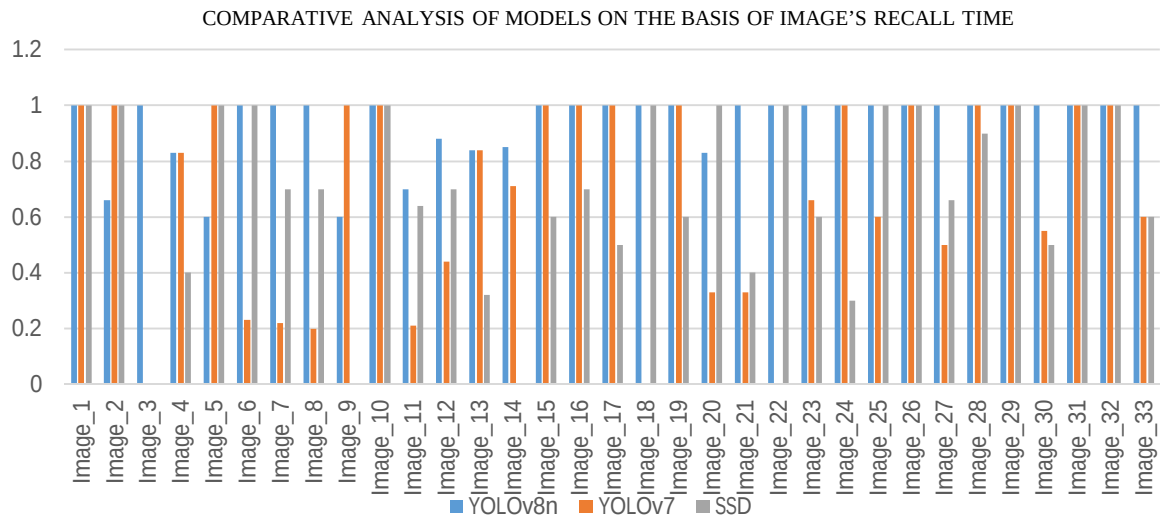


Fig 1(a): Comparative Analysis of Models on the Basis of Image's Recall Time.

Fig 1(a): i.e. "Comparative analysis of models on the basis of image's recall time" shows the recall time of images by YOLOv8n. YOLOv7 and SSD. Recall measures how well our model can find true positives (TP) out of all predictions (TP+FN) [14]. The figure shows the images from 1 to 33. Recall time is calculated for all these (1 to 33) images. Blue lines show the recall time of images by YOLOv8, orange lines show the recall time of images by YOLOv7, and gray lines show the recall time of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

To present 100 images, as depicted in fig 1. (A), fig 1. (B), and fig 1. (C), each figures consist of 100 images. Fig 1 (b) depicted the recall time of resulted images i.e. 37 to 66 images.

Fig1. (b) Shows the recall time of image_36 by YOLOv8n, YOLOv7, and SSD. The recall time of image_36 by YOLOv8n i.e. 1, recall time of image_36 by YOLOv7 i.e. 0.57 and recall time of image_36 by SSD i.e. 0.25.

Fig1. (b) Shows the recall time of image_37 by YOLOv8n, YOLOv7, and SSD. The recall time of image_37 by YOLOv8n i.e. 1, recall time of image_37 by YOLOv7 i.e. 1 and recall time of image_37 by SSD i.e. 0.5.

Fig1. (b) Shows the recall time of image_59 by YOLOv8n, YOLOv7, and SSD. The recall time of image_59 by YOLOv8n i.e. 1, recall time of image_59 by YOLOv7 i.e. 0.83 and recall time of image_59 by SSD i.e. 0.59.

Fig1. (b) Shows the recall time of image_ 66 by YOLOv8n, YOLOv7, and SSD. The recall time of image_ by YOLOv8n i.e. 1, recall time of image_66 by YOLOv7 i.e. 0.60 and recall time of image_ 66 by SSD i.e. 0.40.

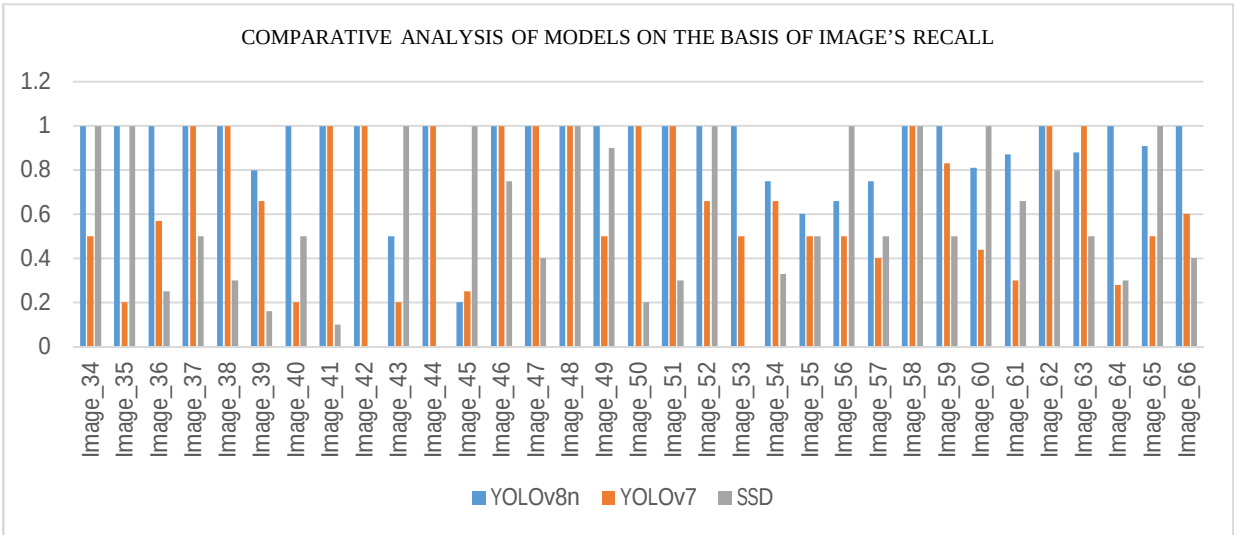


Fig 1(b): Comparative Analysis of Models on the Basis of Image's Recall Time

Fig 1(b): i.e. "Comparative analysis of models on the basis of image's recall time" shows the recall time of images by YOLOv8n. YOLOv7 and SSD. Recall measures how well our model can find true positives (TP) out of all predictions (TP+FN) [16]. The figure shows the images from 34 to 66. Recall time is calculated for all these (34 to 66) images. Blue lines show the recall time of images by YOLOv8, orange lines show the recall time of images by YOLOv7, and grey lines show the recall time of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

To present 100 images, as depicted in fig 1. (A), fig 1. (B), and fig 1. (C), each figures consist of 100 images. Fig 1. (B) depicted the recall time of resulted images i.e. 67 to 100 images.

Fig1. (c) Shows the recall time of image_72 by YOLOv8n, YOLOv7, and SSD. The recall time of image_72 by YOLOv8n i.e. 1, recall time of image_72 by YOLOv7 i.e. 1 and recall time of image_72 by SSD i.e. 0.30.

Fig1. (c) Shows the precision time of image_82 by YOLOv8n, YOLOv7, and SSD. The recall time of image_82 by YOLOv8n i.e. 1, recall time of image_82 by YOLOv7 i.e. 0.40 and recall time of image_82 by SSD i.e. 0.33.

Fig1. (c) Shows the recall time of image_92 by YOLOv8n, YOLOv7, and SSD. The recall time of image_92 by YOLOv8n i.e. 1, recall time of image_92 by YOLOv7 i.e. 1 and recall time of image_92 by SSD i.e. 0.62.

Fig1. (c) Shows the recall time of image_95 by YOLOv8n, YOLOv7, and SSD. The recall time of image_95 by YOLOv8n i.e. 1, recall time of image_95 by YOLOv7 i.e. 0.75 and recall time of image_95 by SSD i.e. 0.60.

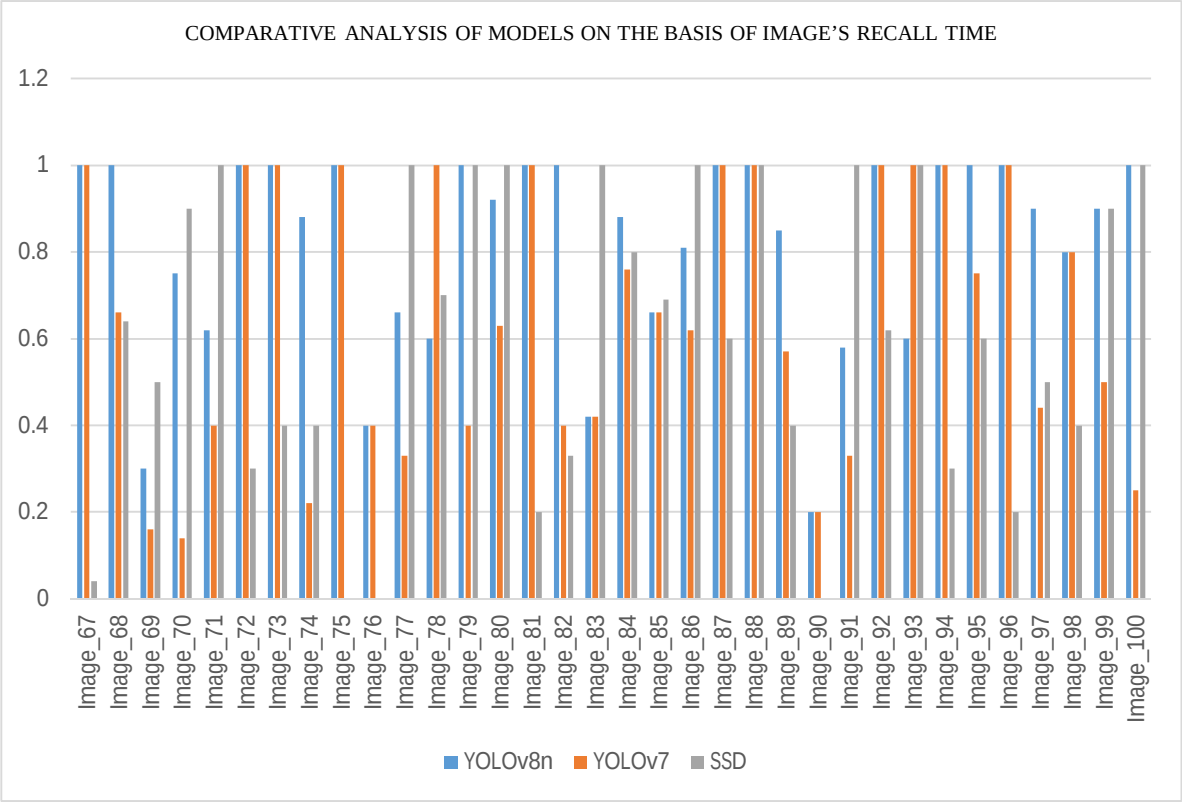


Fig 1(c): Comparative Analysis of Models on the Basis of Image's Recall Time.

Fig 1(c): i.e. Comparative analysis of models on the basis of image's recall time" shows the recall time of images by YOLOv8n. YOLOv7 and SSD. Recall measures how well our model can find true positives (TP) out of all predictions (TP+FN). The figure shows the images from 34 to 66. Recall time is calculated for all these (67 to 100) images. Blue lines show the recall time of images by YOLOv8, orange lines show the recall time of images by YOLOv7, and grey lines show the recall time of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

Fig 4: Here is the average recall time of these 100 resulted images. "Average recall time" which illustrates the recall time of object detection models. YOLOv8 is archives 0.88 recall time, YOLOv7 is archives 0.66 recall time and SSD is archives 0.62 recall time. YOLOv8 is archives higher recall time followed by YOLOv7 i.e. 0.66, SSD i.e. 0.62 at last.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

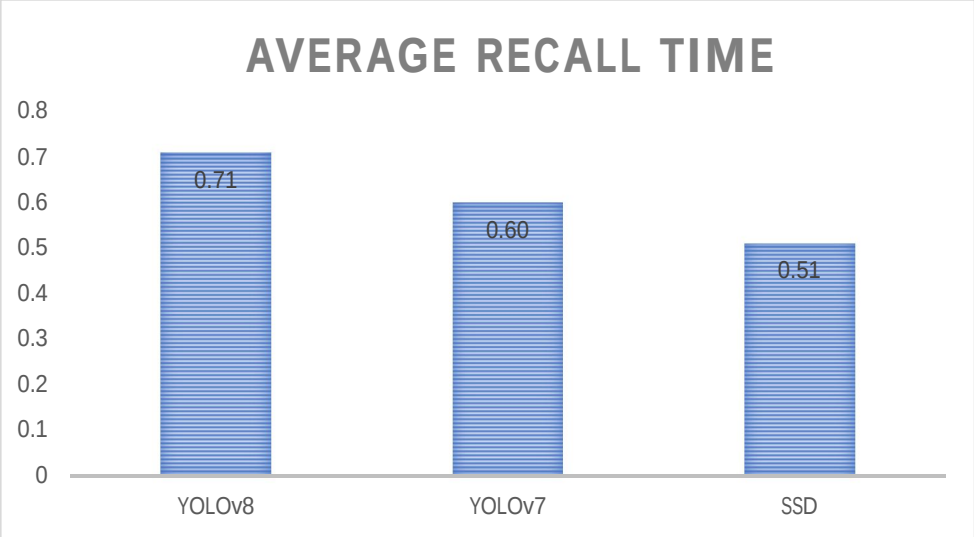


Fig 4: Average Recall Time

Results of Average Recall Time = Average Recall Time of YOLOv8n > Average Recall Time of YOLOv7 > Average Recall Time of SSD. 100 images were tested on Google Colab by the object detection models, i.e. SSD, YOLOv8, and YOLOv7. Metrics used to test the performance of these models are recall time, accuracy, precision time, and map. To present 100 images, as depicted in fig 2. (A), fig 2. (B), and fig 2. (C), each figures consist of 100 images. Fig 2. (a) depicted the precision time of resulted images i.e. 1 to 33 images.

Fig 2. (a) Shows the precision time of image_4 by YOLOv8n, YOLOv7, and SSD. Precision time of image_4 by YOLOv8n i.e. 0.83, precision time of image_4 by YOLOv7 i.e. 0.63 and precision time of image_4 by SSD i.e. 0.33.

Fig 2. (a) Shows the precision time of image_8 by YOLOv8n, YOLOv7, and SSD. Precision time of image_8 by YOLOv8n i.e. 1, precision time of image_8 by YOLOv7 i.e. 0.20 and precision time of image_8 by SSD i.e. 0.70.

Fig 2. (a) Shows the precision time of image_16 by YOLOv8n, YOLOv7, and SSD. Precision time of image_16 by YOLOv8n i.e. 1, precision time of image_16 by YOLOv7 i.e. 1 and precision time of image_16 by SSD i.e. 0.42.

Fig 2. (a) Shows the precision time of image_33 by YOLOv8n, YOLOv7, and SSD. Precision time of image_33 by YOLOv8n i.e. 1, precision time of image_33 by YOLOv7 i.e. 0.60 and precision time of image_33 by SSD i.e. 0.60.

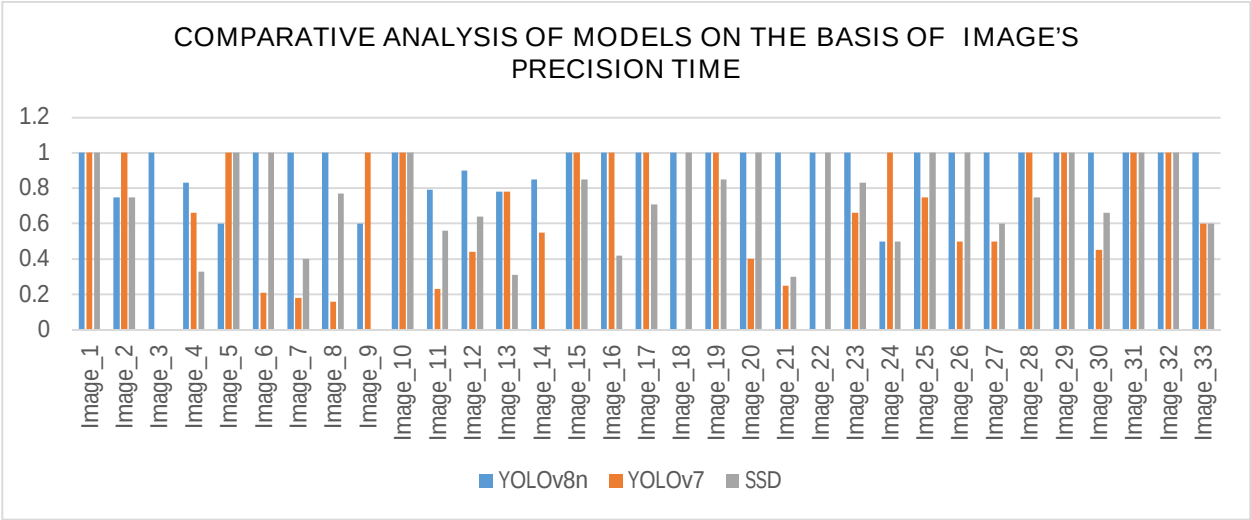


Fig 2 (a): Comparative Analysis of Models on the Basis of Image's Precision Time

Fig 2 (a): i.e. "Comparative analysis of models on the basis of image's precision time" shows the precision time of images by YOLOv8n. YOLOv7 and SSD. Precision measures how well our model can find true positives (TP) out of all positive predictions (TP+FP). The figure shows the images from 1 to 33. Precision time is calculated for all these (1 to 33) images. Blue lines show the precision time of images by YOLOv8, orange lines show the precision time of images by YOLOv7, and grey lines show the precision time of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

100 images were tested on Google Colab by the object detection models, i.e. SSD, YOLOv8, and YOLOv7. Metrics used to test the performance of these models are recall time, accuracy, precision time, and map. To present 100 images, as depicted in fig 2. (A), fig 2. (B), and fig 2. (C), each figures consist of 100 images. Fig 2. (B) Depicted the precision time of resulted images i.e. 34 to 66 images.

Fig 2. (b) Shows the precision time of image_36 by YOLOv8n, YOLOv7, and SSD. Precision time of image_36 by YOLOv8n i.e. 1, precision time of image_36 by YOLOv7 i.e. 0.66 and precision time of image_36 by SSD i.e. 0.25.

Fig 2. (b) Shows the precision time of image_54 by YOLOv8n, YOLOv7, and SSD. Precision time of image_54 by YOLOv8n i.e. 0.66, precision time of image_54 by YOLOv7 i.e. 0.33 and precision time of image_54 by SSD i.e. 0.50.

Fig 2. (b) Shows the precision time of image_57 by YOLOv8n, YOLOv7, and SSD. Precision time of image_57 by YOLOv8n i.e. 0.66, precision time of image_57 by YOLOv7 i.e. 0.66 and precision time of image_57 by SSD i.e. 0.33.

Fig 2. (b) Shows the precision time of image_64 by YOLOv8n, YOLOv7, and SSD. Precision time of image_64 by YOLOv8n i.e. 1, precision time of image_64 by YOLOv7 i.e. 0.33 and precision time of image_64 by SSD i.e. 0.23.

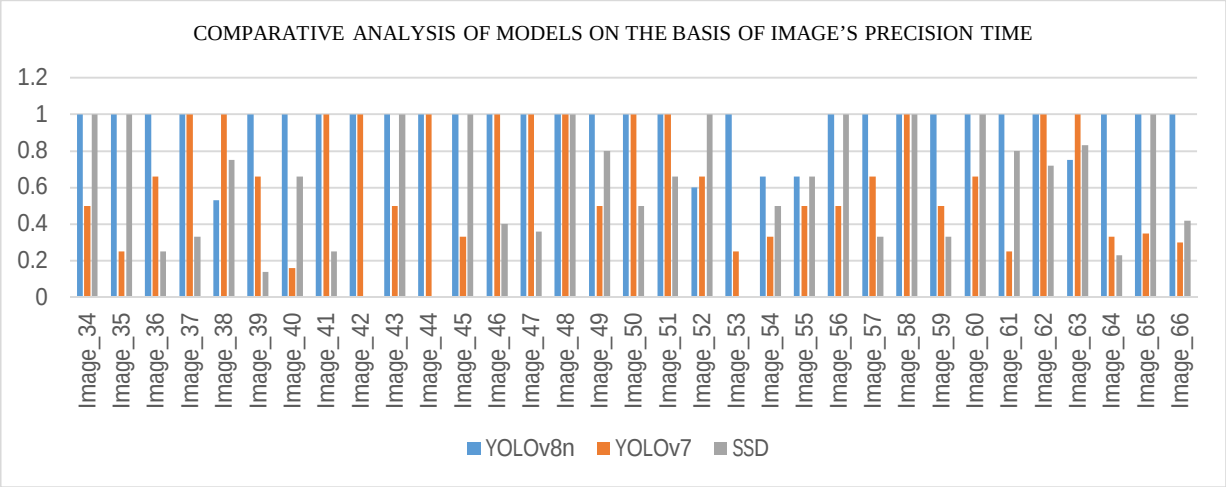


Fig 2(b): Comparative Analysis of Models on the Basis of Image's Precision Time

Fig 2(b): i.e. "Comparative analysis of models on the basis of image's precision time" shows the precision time of images by YOLOv8n. YOLOv7 and SSD. Precision measures how well our model can find true positives (TP) out of all positive predictions (TP+FP). The figure shows the images from 34 to 66. Precision time is calculated for all these (34 to 66) images. Blue lines show the precision time of images by YOLOv8, orange lines show the precision time of images by YOLOv7, and grey lines show the precision time of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

100 images were tested on Google Colab by the object detection models, i.e. SSD, YOLOv8, and YOLOv7. Metrics used to test the performance of these models are recall time, accuracy, precision time, and map. To present 100 images, as depicted in fig 2. (A), fig 2. (B), and fig 2. (C), each figures consist of 100 images. Fig 2. (C) Depicted the precision time of resulted images i.e. 67 to 100 images.

Fig 2. (c) Shows the precision time of image_67 by YOLOv8n, YOLOv7, and SSD. Precision time of image_67 by YOLOv8n i.e. 1, precision time of image_67 by YOLOv7 i.e. 1 and precision time of image_67 by SSD i.e. 0.66.

Fig 2. (c) Shows the precision time of image_74 by YOLOv8n, YOLOv7, and SSD. Precision time of image_74 by YOLOv8n i.e. 1, precision time of image_74 by YOLOv7 i.e. 0.20 and precision time of image_74 by SSD i.e. 0.50.

Fig 2. (c) Shows the precision time of image_82 by YOLOv8n, YOLOv7, and SSD. Precision time of image_82 by YOLOv8n i.e. 1, precision time of image_82 by YOLOv7 i.e. 0.11 and precision time of image_82 by SSD i.e. 0.50.

Fig 2. (c) Shows the precision time of image_97 by YOLOv8n, YOLOv7, and SSD. Precision time of image_97 by YOLOv8n i.e. 1, precision time of image_97 by YOLOv7 i.e. 0.55 and precision time of image_97 by SSD i.e. 0.40.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

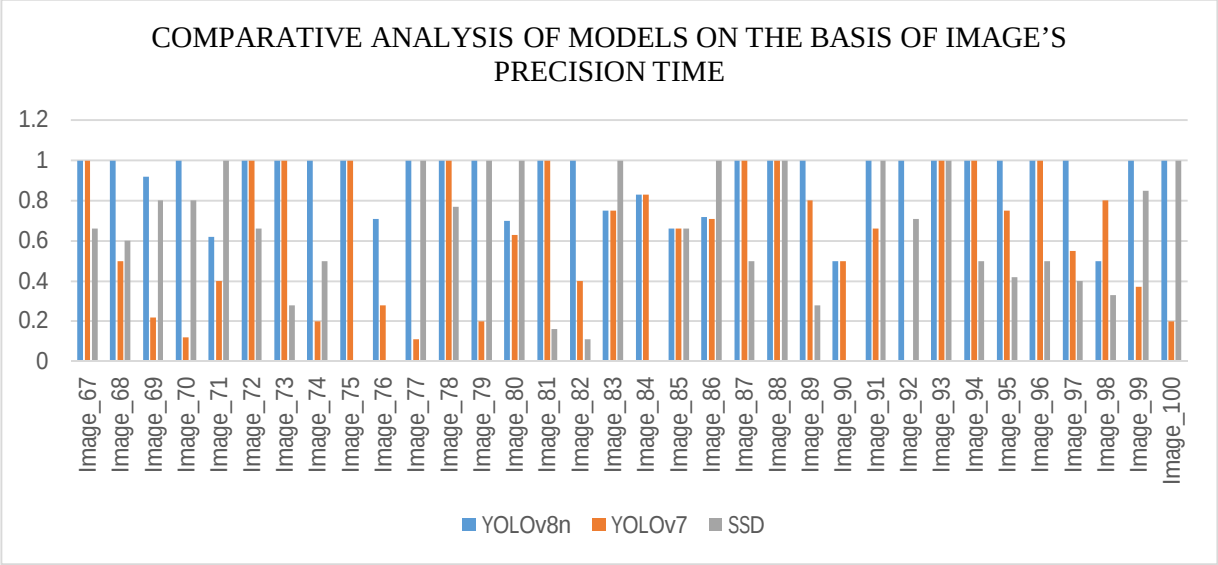


Fig 2 (c): Comparative Analysis of Models on the Basis of Image's Precision Time

Fig 2 (c): i.e. "Comparative analysis of models on the basis of image's precision time" shows the precision time of images by YOLOv8n. YOLOv7 and SSD. Precision measures how well our model can find true positives (TP) out of all positive predictions (TP+FP). The figure shows the images from 67 to 100. Precision time is calculated for all these (67 to 100) images. Blue lines show the precision time of images by YOLOv8, orange lines show the precision time of images by YOLOv7, and grey lines show the precision time of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

Fig 5: Here is the average precision time of these 100 resulted images. "Average Precision Time" which illustrates the precision values of object detection models. YOLOv8 is archives 0.92 precision time, YOLOv7 is archives 0.64 and SSD is archives 0.63 precision time. YOLOv8 is archives higher precision value followed by YOLOv7 i.e. 0.64, SSD i.e. 0.63 at last

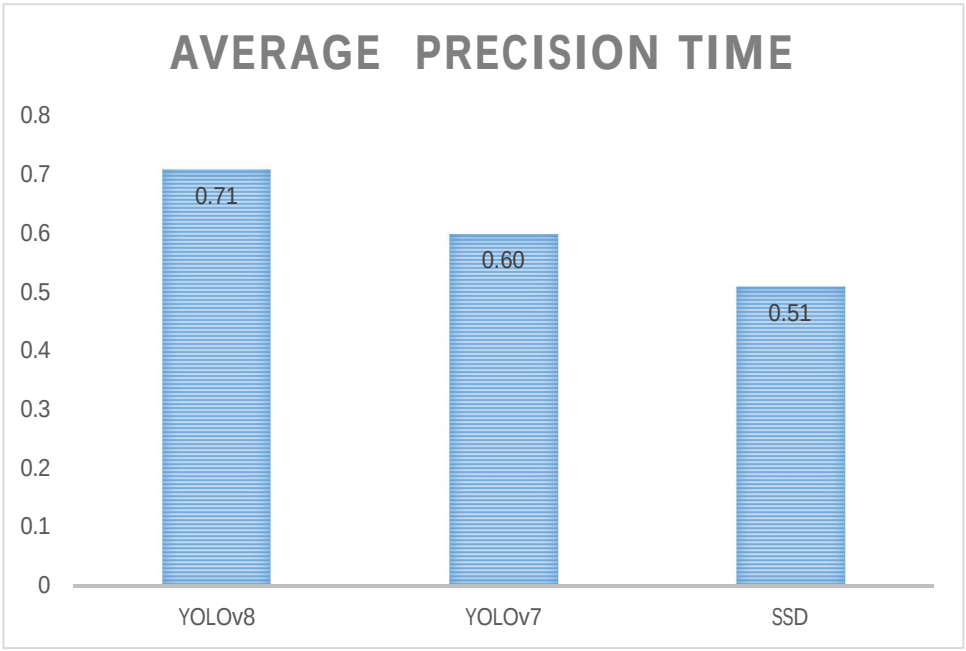


Fig 5: Average Precision Time

Results of Average Precision Time = Average Precision Time of YOLOv8n > Average Precision Time of YOLOv7 > Average Precision Time of SSD.

100 images were tested on Google Colab by the object detection models, i.e. SSD, YOLOv8, and YOLOv7. Metrics used to test the performance of these models are recall time, accuracy, precision time, and map. To present 100 images, as depicted in fig 3. (A), fig 3. (B), and fig 3. (C), each figures consist of 100 images. Fig 2(c) depicted the accuracy of resulted images i.e. 1 to 33 images.

Fig 3(c) depicted the accuracy of image_21 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_21 by YOLOv8n i.e. 100%, accuracy of image_ 21 by YOLOv7 i.e. 33% and accuracy of image_ 21 by SSD i.e. 30%.

Fig 3(c) depicted the accuracy of image_ 24 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_24 by YOLOv8n i.e. 100%, accuracy of image_24 by YOLOv7 i.e. 100% and accuracy of image_ 24 by SSD i.e. 50%.

Fig 3(c) depicted the accuracy of image_30 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_30 by YOLOv8n i.e. 100%, accuracy of image_30 by YOLOv7 i.e. 55% and accuracy of image_30 by SSD i.e. 67%.

Fig 3(c) depicted the accuracy of image_33 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_33 by YOLOv8n i.e. 100%, accuracy of image_33 by YOLOv7 i.e. 60% and accuracy of image_33 by SSD i.e. 60%.

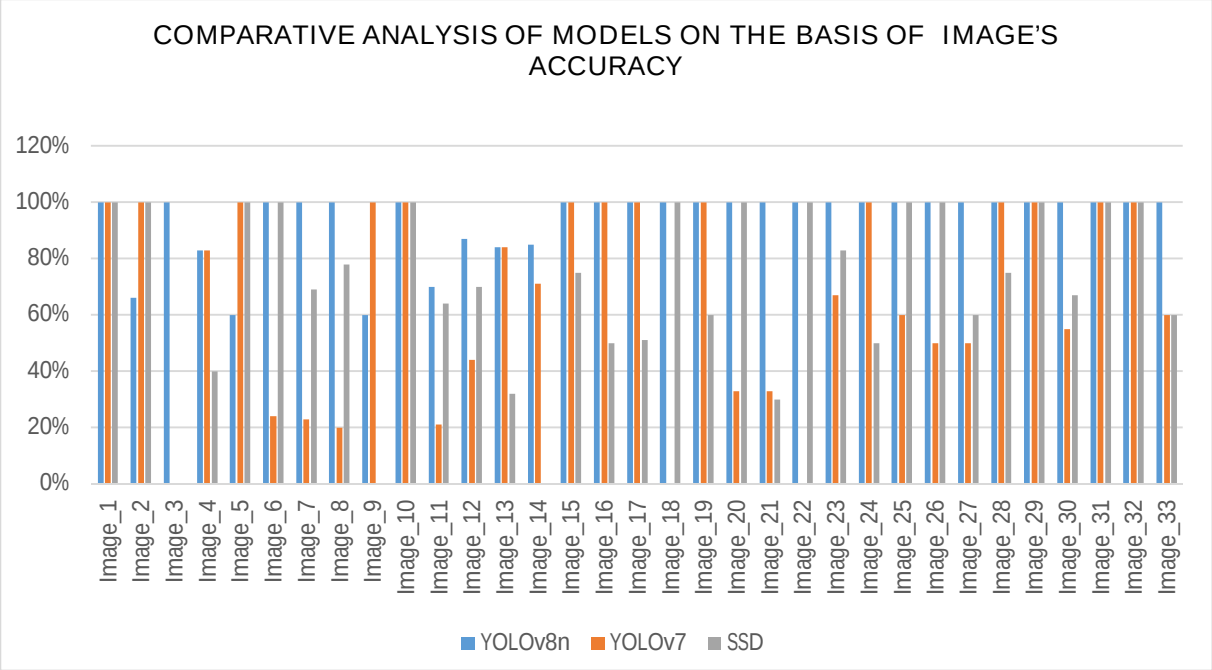


Fig 3 (a): Comparative Analysis of Models on the Basis of Image's Accuracy

Fig 3(a): i.e. "Comparative analysis of models on the basis of image's accuracy" shows the accuracy percentage of images by YOLOv8n. YOLOv7 and SSD. Accuracy measures how accurately our model can detect objects. The figure shows the images from 1 to 33. Accuracy percentage is calculated for all these (1 to 33) images. Blue lines show the accuracy percentage of images by YOLOv8, orange lines show the accuracy percentage of images by YOLOv7, and grey lines show the accuracy percentage of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

100 images were tested on Google Colab by the object detection models, i.e. SSD, YOLOv8, and YOLOv7. Metrics used to test the performance of these models are recall time, accuracy, precision time, and map. To present 100 images, as depicted in fig 3 (A), fig 3. (B), and fig 3. (C), each figures consist of 100 images. Fig 3(c) depicted the accuracy of resulted images i.e. 34 to 66 images.

Fig 3(c) depicted the accuracy of image_36 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_36 by YOLOv8n i.e. 100%, accuracy of image_36 by YOLOv7 i.e. 57% and accuracy of image_ 36 by SSD i.e. 25%.

Fig 3(c) depicted the accuracy of image_ 39 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_39 by YOLOv8n i.e. 100%, accuracy of image_39 by YOLOv7 i.e. 66% and accuracy of image_ 39 by SSD i.e. 16%.

Fig 3(c) depicted the accuracy of image_57 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_57 by YOLOv8n i.e. 67%, accuracy of image_57 by YOLOv7 i.e. 55% and accuracy of image_57 by SSD i.e. 67%.

Fig 3(c) depicted the accuracy of image_66 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_66 by YOLOv8n i.e. 100%, accuracy of image_66 by YOLOv7 i.e. 60% and accuracy of image_66 by SSD i.e. 42%.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

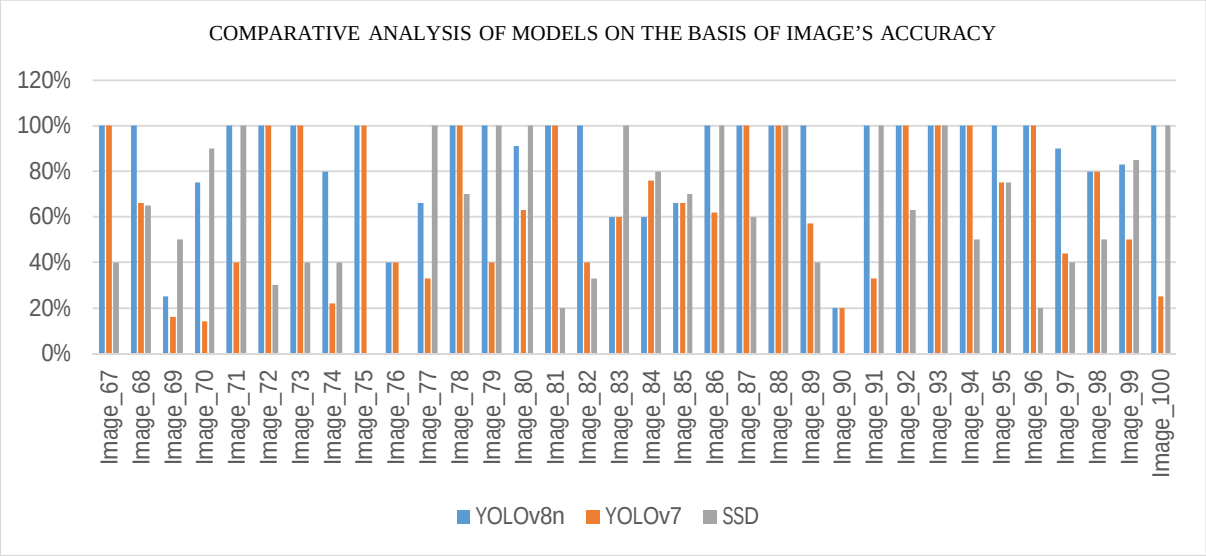


Fig 3 (b): Comparative Analysis of Models on the Basis of Image's Accuracy

Fig 3(b): i.e. "Comparative analysis of models on the basis of image's accuracy" shows the accuracy percentage of images by YOLOv8n. YOLOv7 and SSD. Accuracy measures how accurately our model can detect objects. The figure shows the images from 34 to 66. Accuracy percentage is calculated for all these (34 to 66) images. Blue lines show the accuracy percentage of images by YOLOv8, orange lines show the accuracy percentage of images by YOLOv7, and grey lines show the accuracy percentage of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

100 images were tested on Google Colab by the object detection models, i.e. SSD, YOLOv8, and YOLOv7. Metrics used to test the performance of these models are recall time, accuracy, precision time, and map. To present 100 images, as depicted in fig 3 (A), fig 3. (B), and fig 3. (C), each figures consist of 100 images. Fig 3. (C) depicted the accuracy of resulted images i.e. 67 to 100 images.

Fig 3(c) depicted the accuracy of image_67 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_67 by YOLOv8n i.e. 100%, accuracy of image_67 by YOLOv7 i.e. 100% and accuracy of image_ 67 by SSD i.e. 40%.

Fig 3(c) depicted the accuracy of image_ 74 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_74 by YOLOv8n i.e. 80%, accuracy of image_74 by YOLOv7 i.e. 22% and accuracy of image_ 74 by SSD i.e. 40%.

Fig 3(c) depicted) the accuracy of image_89 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_89 by YOLOv8n i.e. 100%, accuracy of image_89 by YOLOv7 i.e. 57% and accuracy of image_89 by SSD i.e. 40%.

Fig 3(c) depicted the accuracy of image_97 by YOLOv8n, YOLOv7, and SSD. Accuracy of image_97 by YOLOv8n i.e. 90%, accuracy of image_97 by YOLOv7 i.e. 44% and accuracy of image_97 by SSD i.e. 40%.

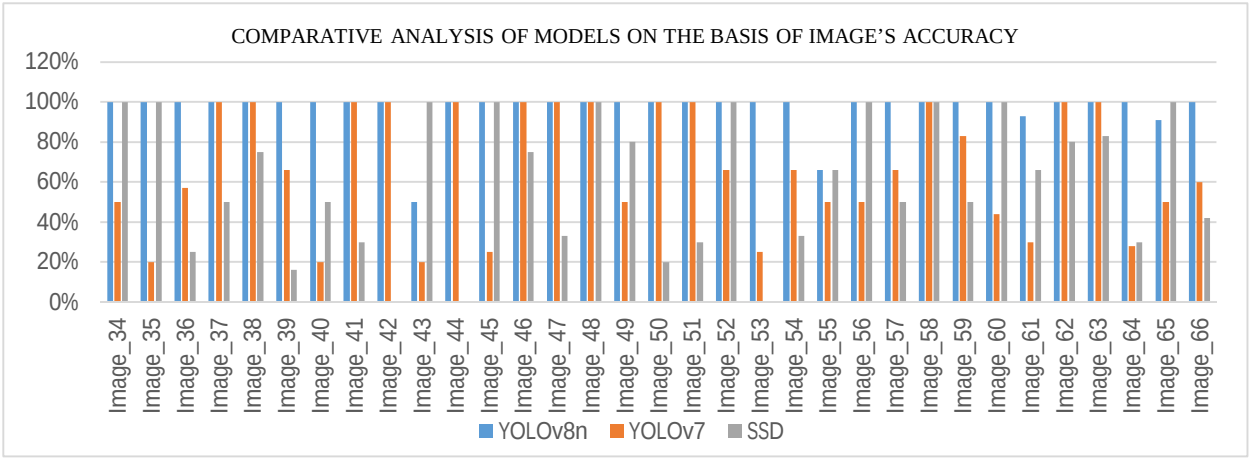


Fig 3 (c): Comparative Analysis of Models on the Basis of Image's Accuracy

Fig 3(c): i.e. "Comparative analysis of models on the basis of image's accuracy" shows the accuracy percentage of images by YOLOv8n. YOLOv7 and SSD. Accuracy measures how accurately our model can detect objects. The figure shows the images from 34 to 66. Accuracy percentage is calculated for all these (67 to 100) images. Blue lines show the accuracy percentage of images by YOLOv8, orange lines show the accuracy percentage of images by YOLOv7, and grey lines show the accuracy percentage of images by SSD. Here, YOLOv8 performs better as compared to YOLOv7 and SSD.

Fig 6: Here is the average precision time of these 100 resulted images. "Average Accuracy" which illustrates the accuracy values of object detection models. YOLOv8 is archives 92% accuracy, YOLOv7 is archives 66% accuracy and SSD is archives 64% accuracy. YOLOv8 is archives higher accuracy followed by YOLOv7 i.e. 66% and SSD i.e. 64% at last

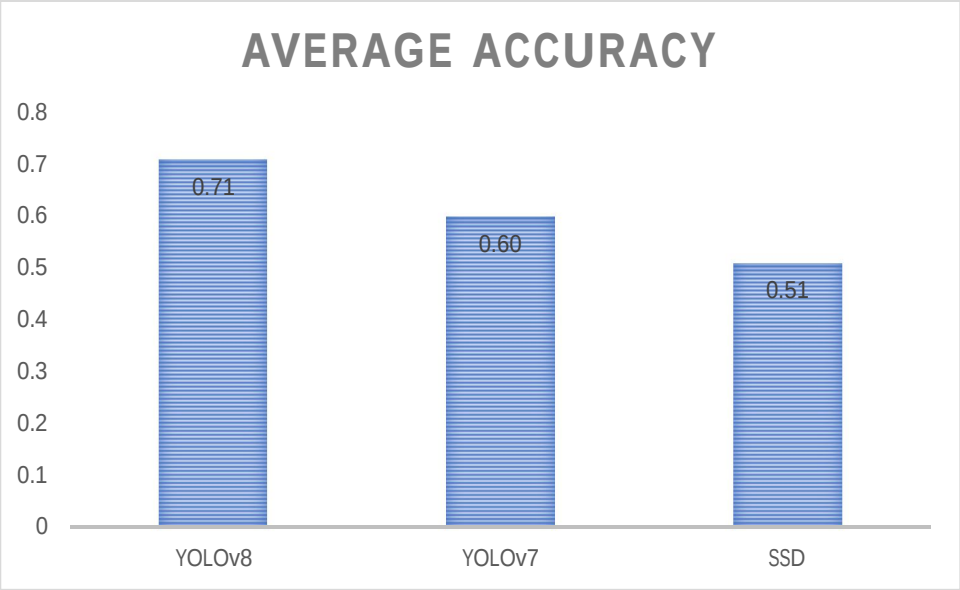


Fig 6: Average Accuracy

Results of Average Accuracy = Average Accuracy of YOLOv8n > Average Accuracy of YOLOv7 > Average Accuracy of SSD.

TABLE 1
Comparative Analysis of Models on the Basis of Mean Average Precision

MODELS	YOLOv8	YOLOv7	SSD
ACCURACY	70.78%	60.8%	51.54%

Table 1 i.e. "Comparative analysis of models on the basis of Mean Average Precision" which shows the accuracy of all the object detection algorithms used for conducting the experiment. The accuracies are as follows: 70.74% for YOLOv8, 60.8% accuracy for YOLOv7 and 51.54% accuracy for SSD.

Fig7. Here is a figure of mean average precision. "Mean Average Precision" which shows the accuracy of all the object detection algorithms used for conducting the experiment. The accuracies are as follows: 70.78% for YOLOv8, 60.8% accuracy for YOLOv7 and 51.54% accuracy.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

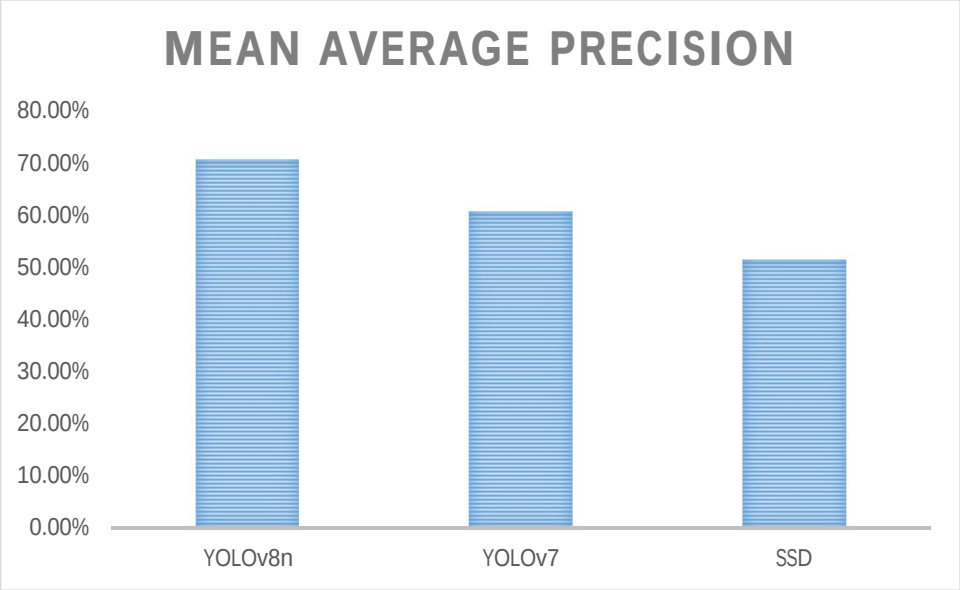


Fig 7: Mean Average Precision

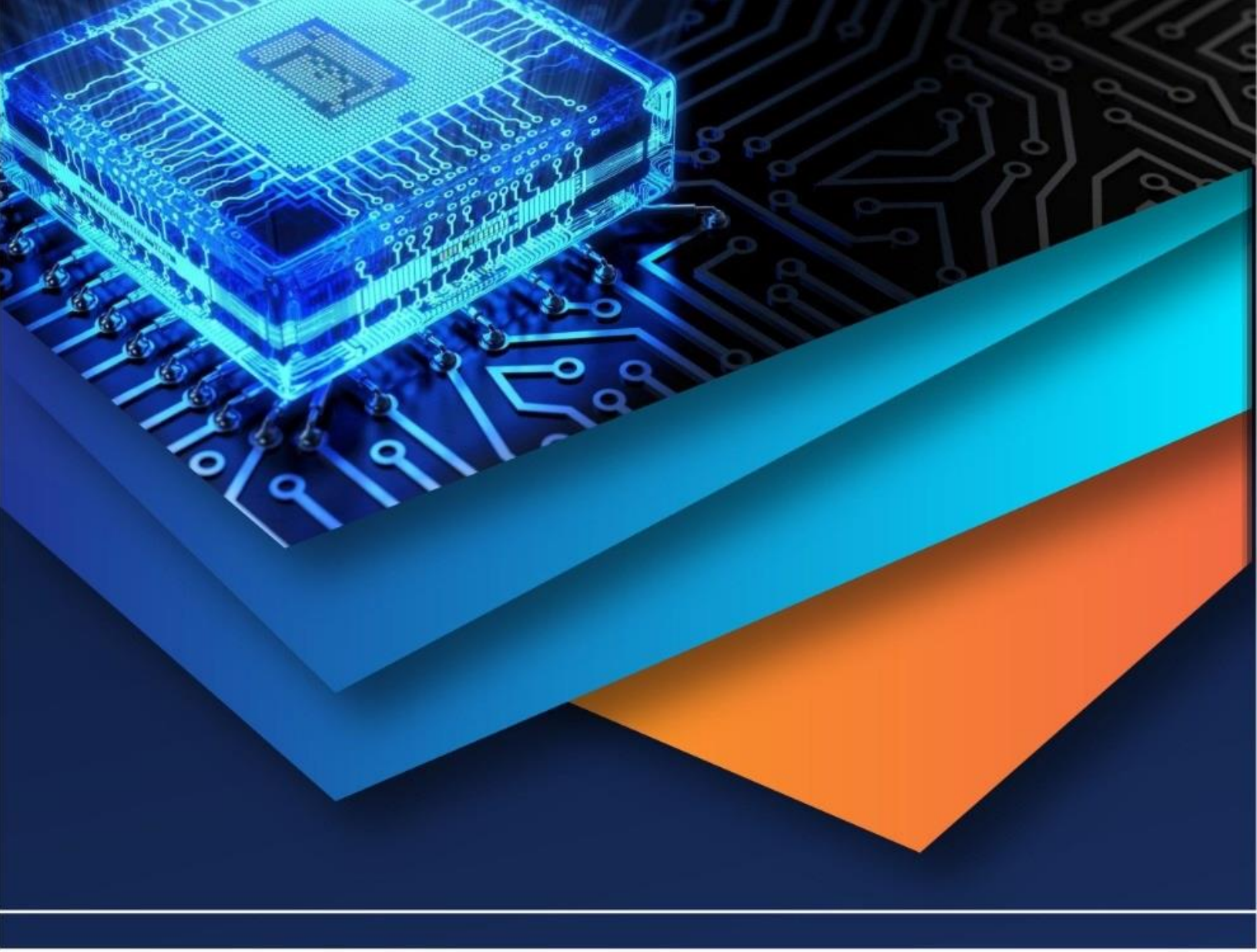
VII. CONCLUSION

The research paper is briefly highlights the object detection and how it is important in today's world. The paper highlights three object detection models and provide brief introduction about them. The three object detection models are You Only Look Once (YOLOv8), You Only Look Once (YOLOv7) and Single Shot MultiBox Detector (SSD). The results which are given by these three object detection models which we are implemented on same dataset, same platform and same system are compared, evaluated and also analyzed. The results of these three object detection models are illustrated by graphs and tables. With the help of results conclusion is drawn out that YOLOv8 yields best accuracy i.e. 70.78% followed by YOLOv7 i.e. 60.8% and SSD has obtained 51.54% accuracy. YOLOv8 yields best precision time i.e. 0.69 followed by YOLOv7 i.e. 0.60 and SSD has obtained precision time 0.51. YOLOv8 yields best recall time i.e. 0.67 followed by YOLOv7 i.e. 0.56 and SSD has obtained 0.52 recall time.

REFERENCES

[1] R. Verschae and J. Ruiz-del-Solar, "Object Detection: Current and Future Directions," Front. Robot. AI, vol. 2, Nov. 2015, doi: 10.3389/frobt.2015.00029.
[2] R. L. Galvez, A. A. Bandala, E. P. Dadios, R. R. P. Vicerra, and J. M. Z. Maningo, "Object Detection Using Convolutional Neural Networks," in TENCON 2018 - 2018 IEEE Region 10 Conference, Jeju, Korea (South): IEEE, Oct. 2018, pp. 2023–2027. Doi: 10.1109/TENCON.2018.8650517.
[3] Y. Li and F. Ren, "Light-Weight RetinaNet for Object Detection." arXiv, May 23, 2019. Accessed: Mar. 18, 2024. [Online]. Available: http://arxiv.org/abs/1905.10011
[4] K. de Langis, M. Fulton, and J. Sattar, "An Analysis of Deep Object Detectors For Diver Detection." arXiv, Nov. 24, 2020. Accessed: Mar. 18, 2024. [Online]. Available: http://arxiv.org/abs/2012.05701
[5] J. Deng, X. Xuan, W. Wang, Z. Li, H. Yao, and Z. Wang, "A review of research on object detection based on deep learning," J. Phys. Conf. Ser., vol. 1684, no. 1, p. 012028, Nov. 2020, doi: 10.1088/1742-6596/1684/1/012028.
[6] W. Liu et al., "SSD: Single Shot MultiBox Detector," vol. 9905, 2016, pp. 21–37. Doi: 10.1007/978-3-319-46448-0_2.
[7] F. Sultana, A. Sufian, and P. Dutta, "A Review of Object Detection Models based on Convolutional Neural Network," vol. 1157, 2020, pp. 1–16. Doi: 10.1007/978-981-15-4288-6_1.
[8] W. Xing, M. R. Sultan Mohd, J. Johari, and F. Ahmat Ruslan, "A Review on Object Detection Algorithms based Deep Learning Methods," J. Electr. Electron. Syst. Res., pp. 1–13, Oct. 2023, doi: 10.24191/jeesr.v23i1.001.
[9] M. Peker, "Comparison of Tensorflow Object Detection Networks for Licence Plate Localization," in 2019 1st Global Power, Energy and Communication Conference (GPECOM), Nevsehir, Turkey: IEEE, Jun. 2019, pp. 101–105. Doi: 10.1109/GPECOM.2019.8778602.
[10] N. Yadav and U. Binay, "Comparative Study of Object Detection Algorithms," vol. 04, no. 11.
[11] D. Aman Kumar Sharma, "COMPARATIVE ANALYSIS OF YOLO AND SSD," Int. Res. J. Mod. Eng. Technol. Sci., vol. 05, no. 12/December-2023, doi: 10.56726/IRJMETS47534.
[12] Eshaan Goyal, "COMPARATIVE STUDY OF VARIOUS OBJECT DETECTION MODELS," Int. Res. J. Mod. Eng. Technol. Sci., doi: 10.56726/IRJMETS35226.
[13] O. K. T. Alsultan and M. T. Mohammad, "A Deep Learning-Based Assistive System for the Visually Impaired Using YOLO-V7," Rev. Intell. Artif. vol. 37, no. 4, pp. 901–906, Aug. 2023, doi: 10.18280/ria.370409.
[14] E. Kahya and Y. Aslan, "Comparative Analysis of Deep Learning Models for Olive Detection on the Branch," WSEAS Trans. Comput., vol. 22, pp. 338–351, Feb. 2024, doi: 10.37394/23205.2023.22.39.

[15] I. S. Gillani et al., "Yolov5, Yolo-x, Yolo-r, Yolov7 Performance Comparison: A Survey," in Artificial Intelligence and Fuzzy Logic System, Academy and Industry Research Collaboration Center (AIRCC), Sep. 2022, pp. 17–28. Doi: 10.5121/csit.2022.121602.
[16] O. E. Olorunshola, M. E. Irhebhude, and A. E. Evwiekpaefe, "A Comparative Study of YOLOv5 and YOLOv7 Object Detection Algorithms," J. Comput. Soc. Inform., vol. 2, no. 1, pp. 1–12, Feb. 2023, doi: 10.33736/jcsi.5070.2023.
[17] G. Al-refai, H. Elmoaqet, M. Ryalat, and M. Al-refai, "Object Detection in Low-Light Environment Using YOLOv7," In Review, preprint, Sep. 2023. Doi: 10.21203/rs.3.rs-3365905/v1.
[18] L. Nkuzo, M. Sibiya, and E. D. Markus, "A Comprehensive Analysis of Real-Time Car Safety Belt Detection Using the YOLOv7 Algorithm," Algorithms, vol. 16, no. 9, p. 400, Aug. 2023, doi: 10.3390/a16090400.
[19] C. Wan, Y. Pang, and S. LAN, "Overview of YOLO Object Detection Algorithm," Int. J. Comput. Inf. Technol., vol. 2, no. 1, p. 11, Aug. 2022, doi: 10.56028/ijcit.1.2.11.
[20] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO: From YOLOv1 and Beyond." arXiv, Oct. 07, 2023. Accessed: Dec. 16, 2023. [Online]. Available: http://arxiv.org/abs/2304.00501
[21] H. M. Zayani et al., "Deep Learning for Tomato Disease Detection with YOLOv8," vol. 14, no. 2, 2024.
[22] X. Wang, H. Li, X. Yue, and L. Meng, "A comprehensive survey on object detection YOLO".
[23] "COCO - Common Objects in Context." Accessed: Mar. 24, 2024. [Online]. Available: https://cocodataset.org/#home
[24] "Why and how to use Google Colab | TechTarget," Enterprise AI. Accessed: Dec. 17, 2023. [Online]. Available: https://www.techtarget.com/searchenterpriseai/tutorial/Why-and-how-to-use-Google-Cola
[25] "Mean Average Precision in Object Detection." Accessed: Mar. 13, 2024. [Online]. Available: https://encord.com/blog/mean-average-precision-object-detection/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)