JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A COMPREHENSIVE ANALYSIS OF ONE-STAGE OBJECT DETECTION MODELS IN COMPUTER VISION

Sahil Thakur, Anshul Kalia and Sumesh Sood
Department Of Computer Science, Himachal Pradesh University, India

Abstract : In the realm of computer vision, object detection encompasses the identification and precise positioning of desired entities within an image or video. With the development of advanced deep learning methodologies, notably in Convolutional Neural Networks (CNNs), Deep learning powered approaches for detection of objects have garnered tremendous acclaim in recent times due to their exceptional capabilities and automatic feature learning capabilities. This study aims to provide a thorough understanding of different One-Stage object detection algorithms, which have shown promising results in various applications. Overall, this study will highlight the strengths and limitations of different single-stage object detection methods, including their architecture, training strategies, and performance on standard benchmarks. By providing a thorough understanding of these methods, this study aims to assist researchers and professionals alike in deciding the appropriate algorithm for their specific application requirements.

Keywords
Computer Vision (CV), Object Detection (OD), Machine Learning (ML), Deep Learning (DL), CNN's

1. INTRODUCTION

Recently, there have been remarkable advancements in computer science, particularly in the area of creating sophisticated intelligent machines and systems that can imitate human intelligence. One of the most captivating and demanding concepts in this field is to equip computer systems with the capacity to perceive and comprehend the environment, just as humans do. This concept forms the basis of computer vision, an interdisciplinary field which focuses on designing systems that can process and analyze visual data to comprehend and identify its content, similar to how humans perceive visual information.

The concept computer vision had been around for a long time, but remarkable advancements have been achieved with the growth of Deep Learning and the utilization of Big Data. With the accessibility of extensive data volumes and robust computational capabilities, researchers have been able to develop sophisticated machine learning algorithms that can learn to recognize patterns and features in visual data.

Detection of individual objects is a fundamental and indispensable task within the field of computer vision that encompasses identifying and locating instances of specific visual objects in images or videos. It represents an important and persistent problem in the domain of computer vision, as it seeks to answer two primary questions: what are the objects present in the image or video and where the object is located exactly in the picture? [33]. Object detection involves training machine learning algorithms to recognize and classify objects based on their features and characteristics. These algorithms are typically trained on large datasets that contain annotated images or videos, where each object of interest is labelled with a bounding box that indicates its location within the image.

It encompasses the two other sub-tasks of computer vision classification and localization. In object detection, the goal is to not only identify the object's class but also to precisely locate and enclose each instance with bounding boxes delineating their boundaries in the image as shown in Fig 1.

2. LITERATURE REVIEW

Redmon et al. in 2015 presented a novel methodology with a brand new object detection model called YOLO (You Only Look Once) [24]. It considers detection as a regression-based task, making predictions for both class probabilities and bounding box coordinates directly from the entirety of the image data. YOLO is highly efficient, achieving speeds of up to 155 FPS, and ImageNet dataset is used for training through the utilization SGD (Stochastic Gradient descent). [24].

Table 1: Difference between Object Detection, Classification, Identification and Localization

Detection	Classification	Identification	Localization
A combination of classification and localization that aims to identify and precisely determine the location of instances of visual objects in the image.	To analyze the given image and classify the content to a defined category.	To analyze the image and find a particular object instance in the image.	To analyze the image and find the location of a given entities within the provided image.

In 2016, Redmon et al. proposed an improved iteration of YOLO, capable of detecting over 9000 object categories. The model showcased a simplified architecture, implemented multi-scale training, incorporated Batch Normalization, leveraged a High-resolution classifier, employed Dimension clusters, harnessed Fine-grained features, and adopted an anchor-based approach. They introduced the DarkNet-19 model, comprising of 19 convolutional and 5 max pooling layers, which are trained on ImageNet dataset using the SGD optimization algorithm. YOLO9000 uses the merged ImageNet and COCO datasets for fast detection and classification of diverse objects [22].

In 2016, Liu et al. introduced Single Shot Detector (SSD). For each feature map point, this model divides the output space of bounding boxes into a collection of default boxes with each default box having a unique aspect ratio and size. At the time of the prediction, it generates scores indicating the presence of items of each category within every default box. Additionally, the network adjusts the parameters to more effectively align with the shape of the item being detected. The network includes predictions from numerous feature maps with different resolutions to efficiently handle objects of variable sizes. This allows for the natural handling of objects with diverse scales and sizes. The Model achieved speed of 59 FPs with 74.3 mAP on the 2007, Pascal VOC test. It is uses an anchor based approach and uses VGG-16 convolutional network as its backbone [18].

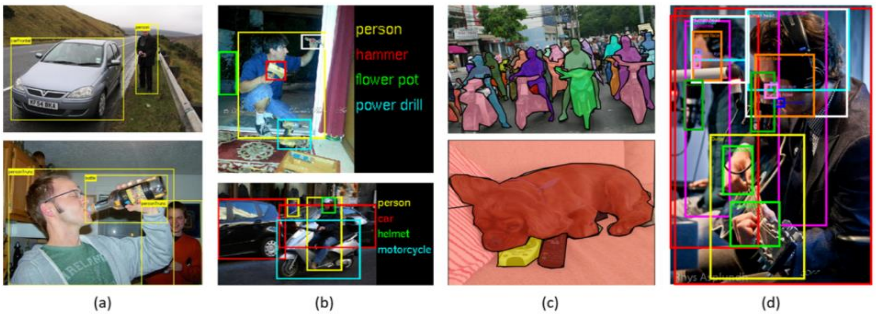

Figure 1: Object Detection with Bounding Boxes [33]

In 2017, Yang-Fu et al. combined a fast detection framework (SSD) with a cutting-edge classifier network (Residual-101) to create DSSD. In order to add more large-scale context to object recognition and increase accuracy, particularly for tiny objects some deconvolution layers are added to the model. The model performs much better than its predecessor SSD on the Pascal VOC test. Here Residual-101 network is used as backbone in place of VGG to enhance the model accuracy. Skip connections are employed to enhance the robustness of features. With regard to tiny items or context-specific objects, the new DSSD model demonstrates superior performance to the SSD framework while maintaining comparable performance to existing detectors [8].

Lin et al. introduced RetinaNet an object detection model proposed at Facebook AI Research (FAIR) in 2017. It addresses the challenge of identifying and locating objects of various sizes and under different lighting conditions in images. RetinaNet uses an innovative function for Focal Loss that is specifically developed to tackle the issue of class imbalance that often arises in object detection tasks, where the majority of image pixels are background or non-object regions. It assigns greater emphasis to challenging examples that are difficult to detect during the training process., which helps it to focus on improving the accuracy of difficult examples. It prevents the overwhelming effect of a large number of simple negative examples at the time of training, which could overshadow the detector's learning process by concentrating training on a small selection of challenging instances. The model also utilizes a Feature Pyramid Network (FPN) integrated with the ResNet architecture that enables efficient detecting of objects of varying sizes by aggregating features from multiple scales. RetinaNet achieves top-performing results on various object detection benchmarks and has been widely adopted in industry and academia for various applications [15].

2018 Redmon et al. Introduced some improvements and design changes to the YOLO V2 model. It introduced an extended variant of the Darknet-19 network known as Darknet-53, featuring 53 convolutional layers for enhanced model capacity and representation power. It makes use of an FPN alongside a SPP net to handle objects of varying sizes and scales. It employs multi-scale training to enhance performance. YOLO v3 demonstrates cutting-edge performance on various object detection benchmarks. On the COCO dataset, YOLO v3 attains mAP (Mean Average Precision) of 57.9, which exceeds different cutting-edge models like Faster R-CNN and SSD [23].

The authors Deng et al. developed a brand-new object detecting system known as CornerNet for precise and timely object detection in 2018. Unlike previous object detection methods that relied on identifying object boundaries or anchor boxes, CornerNet uses a fully convolutional network in order identify object bounding boxes as corner keypoint pairs (top left and bottom right), enabling precise localization and accurate detection even in challenging scenarios. They also introduced Corner Pooling,it consist of a novel pooling layer which brings together feature maps in a way that allows the network to efficiently predict object keypoints and localize corners without losing spatial resolution. The Hourglass network is used as the backbone in this model. The paper proposes a new approach for handling occluded objects in which the network predicts occlusion-aware keypoints that can better handle partial object visibility. In comparison to other techniques of object detection, CornerNet produced outstanding results with faster inference times than most existing methods at the time [14].

Duan et al. in 2019 presented an innovative approach for identifying objects CenterNet on the basis of the CornerNet model. CenterNet identifies every object as a set of three keypoints rather than a pair, which enhances both recall and accuracy. The paper introduces a new "center" heatmap loss function that permits learning on the network to predict the center point and scale of every instance of an item in the given media in a more accurate and robust manner. They also designed two redesigned modules referred as Center Pooling and Cascade Corner Pooling. In cascade corner pooling multiple corner pooling layers are cascaded to aggregate features at different scales and resolutions. In center pooling features are pooled around the centre of each instance of an object to create a representation of features that is invariant to object scale, orientation, and aspect ratio. This enables the model to accurately detect objects of different sizes and aspect ratios using a single scale feature map. Both cascade corner pooling and center pooling are techniques designed to increase accuracy as well as efficiency of object detection models by leveraging more informative and robust representations of objects [4].

Tian et al. in 2019 introduced FCOS a single-stage, entirely convolutional detector that tackles the challenge of approaching object detection in a per-pixel prediction manner, resembling the methodology used in semantic segmentation. It makes use of ResNet architecture as its backbone. The model divides the feature map into a set of grids and predicts objectness scores, regression offsets, and center-ness scores for each grid cell. The centerness score is a new concept introduced by FCOS that captures how close the object center is to the center of the grid cell. This score helps the algorithm to overcome the limitations of previous approaches that rely on bounding box overlap to assess the detection quality. The entire algorithm is trained from beginning to end and post-processing techniques like non-maximum suppression (NMS) are not used to refine the results, which simplifies the detection pipeline. It is an anchor free model [28].

In 2020, Bochkovskiy et el. proposed a series of improvements to the YOLOv3 model, including a new backbone network, a more advanced data augmentation strategy, and various optimization techniques. These enhancements lead to a model that attains cutting-edge accuracy on various object detection benchmarks, all the while maintaining real-time efficiency on modern graphics processing units (GPUs). Several enhancements have been implemented, which encompass the integration of novel elements such as Cross-Stage-Partial-connections (CSP), Self-adversarial training (SAT), Mosaic data augmentation, Weighted Residual Connections (WRC), DropBlock regularization, Mish activation and CIoU loss. YOLOv4 uses various optimization techniques, that are designed to enhance both training efficiency and the overall model performance. It is an anchor-based model. The results demonstrate that YOLOv4 achieves very good performance, surpassing other similar models [2].

The authors Mingxing et al. in 2020 introduced a family of models that attain cutting-edge performance on the COCO benchmark while also demonstrating computational efficiency. They introduce a novel compound scaling approach which scales the resolution, depth and width uniformly of both the backbone and the detection head.. This allows them to optimize the model for both accuracy and efficiency. The paper introduces a novel weighted Bi-directional Feature Pyramid Network (BiFPN)[27] which enables efficient fusion of features across different scales within the network. This helps enhance the performance of the model at the same time maintaining low computation costs. The EfficientDet model utilizes a modified version of the EfficientNet as its backbone. The EfficientDet models achieves excellent performance on the COCO dataset while utilizing fewer FLOPS compared to other models. EfficientDet-D7 model attains a total mAP of 55.1 on the test-dev dataset of COCO while using 410 billion FLOPS which is less than other models such as the Swin Transformer and the DETR model [27].

3. APPROACHES TO OBJECT DETECTION

3.1 Classical Approach

The classical approach to object detection involves the development of models using traditional and simpler machine learning techniques, which were predominantly used before 2013-2014. During this period, considerable advancements have been made using this approach, with several milestones achieved. Efforts to develop classical object detection models began in the latter part of the 20th century and significant breakthroughs have been made in the early 2000s with the introduction of the Viola-Jones detector. This detector, developed in 2004, has been a breakthrough in terms of real-time object detection [30, 31]. In 2005 a histogram-based approach has been introduced [3]. Other important models include SIFT [19] and DPM [7].

In the classical approach of object detection, a crucial step called feature extraction is performed to identify the distinctive attributes of the object. It entails the utilization of manually designed or handcrafted features., which are manually designed to detect specific object characteristics. These features are then fed into a machine learning algorithm, which learns to classify objects based on these features. While this approach has been successful in many applications, it does have limitations in terms of scalability and adaptability to new environments. A notable limitation of this approach is its reliance on the need of expert judgement to select the optimal features for efficient detection, which is a challenging task. Moreover, fine-tuning a considerable number of parameters related to these features can be a time-consuming and demanding undertaking. This approach places less demand on hardware resources but relies heavily on human effort. The selection of features can greatly impact the accuracy and efficiency of the detection process, and as a result, it requires significant expertise and experience to optimize these features properly. Moreover, the process of parameter tuning requires a lot of manual intervention, which is time-consuming and susceptible to human errors, posing challenges and potential inaccuracies. [20].

3.2 Deep Learning Approach

The advancement in computer technology, particularly the rapid increase in computational power, has led to a general trend towards the utilization of deep learning methodologies for object detection. While the classical approach provided satisfactory results, the deep learning approach offers much better accuracy and performance, albeit at the cost of requiring more powerful hardware [20]. The deep learning approach is grounded in the concept of neural networks., which are designed to learn and extract features from raw data, such as images. With regard to object detection, CNNs emerged as the predominant deep learning method employed in various applications, including object detection due to their capability to learn and capture spatial relationships among pixels within an image.

Since the early 2010s, the deep learning-based approach has gained prominence due to significant advancements in terms of speed, accuracy, and overall capability of object detection. The use of GPU (Graphics Processing Unit) technology has further accelerated the training of large neural networks, making it possible to process huge volumes of data in a relatively short amount of time.

Furthermore, the deep learning-based approach has demonstrated remarkable achievements across multiple applications, including medical diagnosis, autonomous driving, and security. This success can be chalked up to their inherent capability to autonomously learn and extract features from raw data. Despite its reliance on powerful hardware, the deep learning-based approach offers a more efficient and effective solution to object detection and has paved the way for further advancements in this field.

Several notable DL based models include R-CNN, Fast R-CNN, YOLO, and SSD. [11, 10, 24, 18].

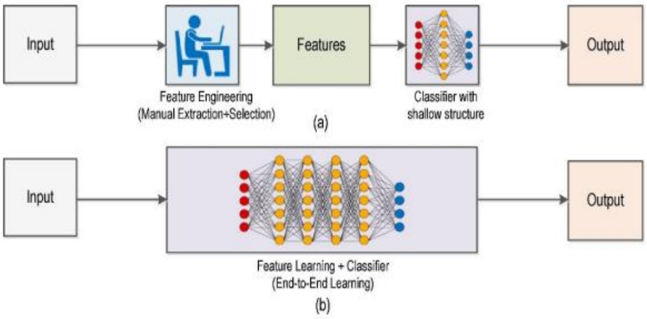

Figure 2: Classical vs. Deep Learning Workflow [20] (a) Classical approach (b) Deep Learning based approach

4. DEEP LEARNING BASED MODELS IN OBJECT DETECTION

DL based models for detection are categorized into two main approaches: One-Stage detectors and Two-Stage detectors [2]. Both approaches have been briefly explained but the one-stage detectors have been discussed primarily.

4.1 Two Stage Detection

In this approach, the detection is divided into two distinct stages. The initial stage involves proposal of regions, where a Region Proposal Network (RPN) evaluates an image and creates proposals for regions which have more probability to contain objects of interest. This helps reduce computational demands by focusing only on relevant regions. The second stage involves the detection of object instances within the proposed regions, one at a time. The two-stage detection process has been described as a “coarse to fine process” [33].

Two-stage detectors typically have better levels of perfomance compared to one-stage detectors, because they first generate region proposals and subsequently use a separate network to classify these regions. This two-step process helps refine the detection and classification accuracy.. This approach allows for more accurate localization of objects [25].

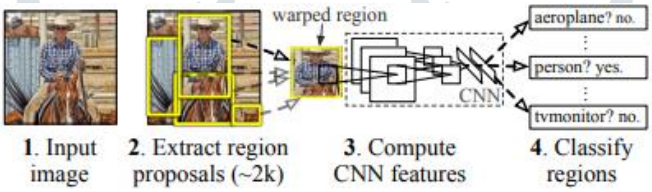

Figure 3: Two stage detection with R-CNN [11]

However, the trade-off for achieving higher accuracy with two-stage detectors is that they are generally slower in terms of computational speed compared to one-stage detectors. This is because they require two passes through the network (one for region proposal and one for classification), which can be computationally expensive [24].
Some famous two-stage detectors include R-CNN [11], Fast R-CNN [10], Faster R-CNN [25], etc.

4.2 One Stage Detection

One-stage object detection is a technique where an entire image is processed at once, without the need for region proposals. This is accomplished by utilizing only a single CNN for the whole object detection task that simultaneously detects the presence of objects, predicts their associated classes and bounding boxes. To achieve this, the image is initially separated into a collection of grids or cells, and then the CNN is applied to each of these grids. The output of the CNN for each cell includes a set of bounding box predictions and class probabilities [24].

One-stage models can employ either anchor based or anchor free bounding boxes. In the anchor-based approach, convolutional neural network (CNN) employs previously defined anchor boxes with different sizes and aspect ratios to estimate the position and dimensions of objects within each grid cell.. The anchor-free approach, on the other hand, does not use pre defined anchor boxes and instead directly predicts the object's bounding box coordinates [13].

The one-stage detection process is described as to “complete in one step” [33]. One-stage object detectors provide a balance between accuracy and speed. Some recent one-stage detectors, such as EfficientDet [27], have outperformed many two-stage detectors.

One loss function is commonly used to train one-stage detectors from beginning to end, which has simplified the training process and reduced the risk of error propagation between stages. One-stage detectors fit real-time object detection applications well, such as autonomous driving, robotics, and surveillance. These applications frequently call for quick and precise object recognition in a variety of environments and lighting conditions [24]. As a result, the interest for one-stage detectors has been rising. The study focuses on the different models that have been proposed in this area.

Some of the most popular one-stage models include YOLO (You Only Look Once) [24], RetinaNet [15], SSD (Single Shot Detector) [18], DSSD [8] and EfficientDet [27]. The choice of model relies on the particular requirements of the application, as each of these models has strengths and drawbacks of its own.

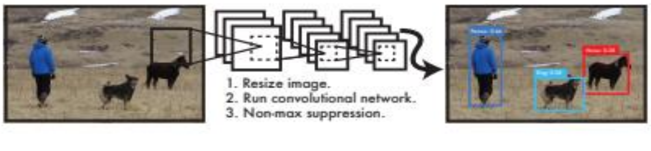

Figure 4: One-Stage Detection with YOLO V1 [24]

5. CONVOLUTIONAL NEURAL NETWORKS

Convolutional neural networks (CNNs) constitute the foundation for each and every model described in this paper. These are a particular class of Artificial Neural Network (ANN) employed for image identification applications. Comparable to conventional ANNs, CNNs consist of neurons that learn and improve their performance through a self-optimization process [21]. CNNs comprises of the following components.

5.1 Convolutional Layer

The input data is subjected to a series of teachable filters in this layer. Each filter applies a convolution operation on the input picture to create an output feature map. The feature map indicates if a specific feature is present at different locations in the image. The next layer receives the feature maps produced by the filters once they have learned to extract key characteristics from the input data, such as edges or corners.

5.2 Activation Function

This layer introduces non-linearity into the CNN by using a mathematical function on each neuron's output. Rectified Linear Unit (ReLU), the most often used activation function, which zeroes off any negative values while maintaining positive values. Other functions include Sigmoid, Tanh, Softplus etc.

5.3 Pooling Layer

The produced feature maps are downscaled using this layer. The feature maps' spatial dimensions are decreased by the pooling procedure while the key characteristics are kept. The most popular pooling method, known as max pooling, uses the highest value found in each local area of the feature map.

5.4 Fully Connected Layer

Also known as the Dense layer. In this layer, every neuron is coupled to every other neuron in the layer that follows it. Its function is to carry out the classification operation by computing each class's output probabilities. In other words, it integrates the high-level information that the convolutional layers have collected to determine the final classification of the input picture.

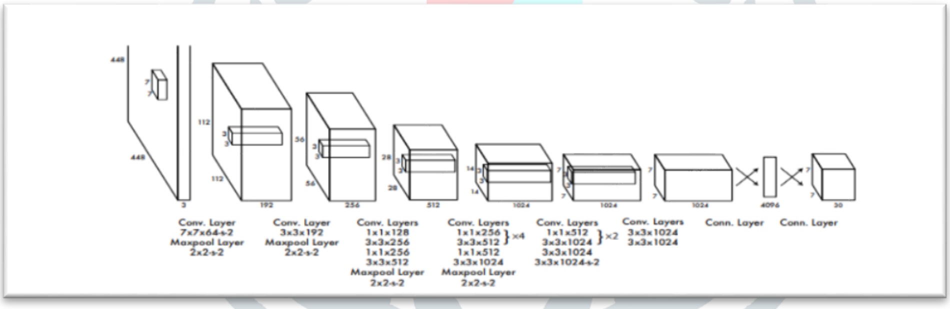

Figure 5: Darknet Network Architecture [24]

6. COMPOSITION OF OBJECT DETECTION MODEL

Most contemporary object detectors are built using convolutional neural networks. There are four primary parts of these detectors [2]. Each of these components has a specific role in the object detection process, and they work together to identify and locate objects in an image.

6.1 The Input

The input component of an object detection system refers to the visual information that the model receives and processes. This visual data can take various forms, such as image files, video files, collections of images, or image pyramids [2]. For an object detection system to identify items accurately and effectively, the input component is essential. Making sure that the supplied data is of excellent quality is crucial, with sufficient resolution and contrast to enable the detector to identify objects with high accuracy. Additionally, the input data must be pre-processed to ensure that it is in a format that the object detection model can process. The images may need to be resized or normalized to ensure that they are of a consistent size and color range [12], [25], [2].

6.2 The BackBone

The Backbone is a crucial component in modern object detection models. It is a pre-trained network that is accountable for filtering features from the supplied data. The backbone network takes in the raw input data and processes it to extract high-level features that are then employed to determine what items are in the picture. In order to create an effective object identification model that fulfils the task's criteria, choosing the right backbone network is crucial. [32].

Several backbone networks have been developed and used in object detection models. Some of the most popular backbone networks are DarkNet [23], ResNet [12], and VGG [26].

6.3 The Neck

In object detection, the neck refers to the intermediate layers that sit between the backbone and the head of the network [2]. The neck's function is to gather and improve feature maps from various backbone network stages and combine them into a multi-scale representation of the input image. There are several variants of neck architectures used in one-stage object detection, including FPN (Feature Pyramid Network) [29], PAN (Path Aggregation Network) [17], and NAS-FPN (Neural Architecture Search based FPN) [9].

6.4 The Head

In single-stage object detection, the model's head which is where the actual classification and bounding box prediction tasks are carried out, is an essential part. The head takes the aggregated features from the neck and uses them to generate predictions for the objects in the input image. There are two categories of heads: those with sparse predictions and those with dense predictions as shown in Fig 6.

Sparse Prediction : R-CNN[11], Fast-RCNN[10].

Dense Prediction : YOLO[24], SSD[18], CornerNet[13].

Each type of head has its own advantages and disadvantages. Sparse prediction methods have better accuracy but are computationally expensive, while dense prediction methods are faster but have lower accuracy.

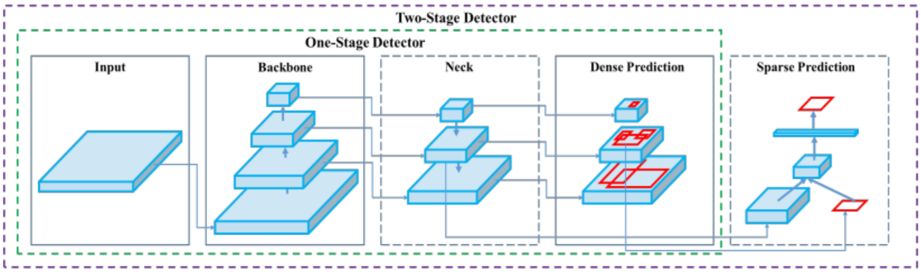

Figure 6: Composition of object detection models [2]

7. DATASETS

For developing and assessing the detection models, datasets are essential. The availability of large, diverse, and unbiased datasets is essential for developing accurate and robust models [33]. There are many popular datasets available for object detection, nevertheless it's crucial to choose datasets wisely that are suitable for the task at hand. The study focus on evaluating various single-stage object detection algorithms using the following datasets:

7.1 Pascal Voc

The Pascal Visual Objects Classes (VOC) challenge was one of the first and most important competitions in the field of computer vision, which were organized from 2005 to 2012 [6, 5]. Thousands of images of items from 20 different daily life categories are included in the Pascal VOC dataset. VOC 2007 and VOC 2012, are two prominent versions of the dataset which are frequently employed to assess object detection models. Further details are provided in Table 2.

Table 2: Statistics from the mentioned object detection datasets [33]

Dataset	Train		Validation		Total	
	Images	Objects	Images	Objects	Images	Objects
VOC-2007	2501	6301	2510	6307	5011	12608
VOC-2012	5717	13609	5823	13841	11540	27450
MS-COCO-2015	82783	604907	40504	291875	123287	896782
MS-COCO-2017	118287	860001	5000	36781	123287	896782

Additional information on the dataset is provided in Table 1.2, including the number of images and object instances in the training and validation datasets.

7.2 MS COCO

In the realm of object detection, microsoft common objects in context (ms-coco) is considered one of the biggest and widely used dataset [16]. It consists of over 100,000 distinct images with objects from 80 different categories, making it significantly larger than the pascal voc dataset. For training and assessing object detection models, ms-coco has grown to be one of the most significant datasets. Nearly all cutting-edge models in this field are tested and evaluated using this dataset to gauge their effectiveness. Table 2 provides detailed statistics of the dataset

8. COMPARISON AND ANALYSIS

By examining the specific features, parameters, and results of each algorithm, we aim to gain a deeper understanding of their strengths and limitations. By thoroughly examining these models, we have developed a more effective understanding of their capabilities. This allows us to make informed decisions about which models are best suited to a given problem.

Table 3: Comparison of the components of object detection models.

Name	Year	BackBone	Neck	Anchor Boxes	Training Method	Activation Function
YOLO	2015	DarkNet	-	Anchor Free	SGD	Leaky ReLu
YOLO v2	2017	Darknet-19	-	Anchor Based	SGD	Leaky ReLu
SSD	2015	VGG-16	-	Anchor Based	SGD	ReLu
DSSD	2017	ResNet	Deconvolutional module	Anchor Based	SGD	ReLu
RetinaNet	2017	ResNet-FPN	FPN	Anchor Based	SGD	ReLu
YOLO v3	2018	Darknet-53	FPN	Anchor Based	SGD	Leaky Relu
CornerNet	2018	Hourglass-104	Hourglass	Anchor Free	SGD	ReLu
CenterNet	2019	Hourglass-104	DLA	Anchor Free	SGD	ReLu
FCOS	2019	ResNet-FPN	FPN	Anchor Free	SGD	-
YOLO v4	2020	CSP-Darknet-53	SPP , PAN	Anchor Free	ADAM	Leaky ReLu, Mish, Swish
EfficientDet	2020	EfficientNet	Bi-FPN	Anchor Based	SGD	Swish-1

Table 4: Analysis of Strengths and Limitation with Remarks

Name	Strengths	Limitations	Remarks
Yolo V1	• YOLOv1 achieves precise localization accuracy through direct bounding box prediction.	• Struggles with detection of small objects. • Struggles to perform well for complex	Introduced a novel approach to Object detection that provides good results

	• Generalizes well to new domains.	scenes with object overlap or occlusion etc.	while generalizing well to new domains.
Yolo V2	• Uses Anchor Boxes to improve localization of objects. • Introduced many useful new features such as High resolution classifier, Fine grade features, Batch Normalization and more.	• Due to the Anchor box approach it has difficulties in handling aspect ratio variations. • Struggles with contextual information.	YOLOv2 provides improved accuracy and handles objects of different scales, but struggles with extreme aspect ratios and small object detection.
SSD	• Uses the idea of outputs from multi-scale convolutional bounding boxes, which are connected to a number of feature maps placed at the top of the network. • It can handle objects of various sizes very well by combining predictions from several feature maps with various resolutions.	• Lower accuracy in case of small objects and Complex scenarios. • Increased computational demand due to the multi scale architecture and multiple feature maps.	SSD offers efficient multi-scale detection but struggles with very small objects and densely packed scenes. Contextual understanding is limited.
DSSD	• Utilizes Deconvolutional layers and multi-scale feature maps leading to improved accuracy. • Deconvolutional layer helps in precise localization of objects. • Better results in case of small and context specific objects.	• Deconvolutional layer causes increase in the computational requirements. • Has difficulty in handling densely packed objects like previous models.	DSSD introduces Deconvolution into the model and achieves accurate multi-scale object detection but has increased computational complexity and limitations in handling fine-grained objects.
RetinaNet	• Introduces a novel Focal Loss function and utilizes a FPN to improve accuracy. • Effectively takes care of objects with various sizes and aspect ratios. • Gives better results in handling object occlusion.	• Relatively high computational requirements. • Has limited contextual understanding. • Retina net is sensitive to hyper parameter tuning.	RetinaNet's focal loss function tackles class imbalance, enabling high accuracy, but small objects and imbalanced datasets remain challenging.
YOLO v3	• Introduces a new and improved Darknet-53 Backbone network. • Thereal time performance is much better than previous models. • Increased performance for detection of small objects.	• Increased computational requirements due to a deeper network. • Difficulty in handling overlapping objects.	YOLOv3 achieves high accuracy and real-time performance, but struggles comparatively with medium to large sized objects and overlapping instances.

CornerNet	• CornerNet excels in detecting objects with diverse rotations, enhancing its suitability for scenarios with varied orientations. • Achieves fast inference times due to its lightweight architecture and the absence of anchor boxes or complex post-processing steps. • Introduces corner pooling which improves performance of the mdoel.	• CornerNet's corner detection is prone to errors in cluttered or occluded scenes due to challenging corner localization. • Lacks contextual understanding which limits performance in complex scenes.	CornerNet's unique approach to object detection using corner pairs shows promise and effectiveness in detecting objects with arbitrary orientation but can struggle with precise corner localization in cluttered or occluded scenes.
CenterNet	• CenterNet directly predicts object centers, leading to precise localization and a reduction in false positive detections. • Increases the accuracy and recall of object recognition by detecting each item as a triplet of keypoints. • CenterNet can detect objects with arbitrary shapes and orientations, making it suitable for diverse object detection tasks.	• CenterNet's reliance on center points may hinder accurate object detection and localization in cluttered or occluded scenes. • Lack of contextual understanding may impact performance. • Sensitive to densely packed or overlapping objects.	CenterNet's direct object center prediction enables accurate localization and reduced false positives, making it a promising approach for efficient and precise object detection tasks.
FCOS	• FCOS can detect objects of various sizes without relying on predefined anchor boxes, making it robust to scale variations. • FCOS follows a fully convolutional architecture, which simplifies the training process and enables efficient inference. • Can work as an effective region proposal network in two stage models.	• FCOS can be sensitive to densely packed objects, potentially leading to lower detection accuracy	FCOS offers scale-invariant detection and accurate localization but may face challenges with small objects and dense scenes making it a promising approach for robust object detection tasks.
YOLO v4	• Useful new features are introduced such as WRC, CSP, Mosaic data augmentation etc. • Yolo v4 is very versatile that makes it suitable for diverse use cases like pedestrian detection to industrial automation. • A very fast model that can give speeds up to 150+ Frames per second.	• Struggles with detection of smaller objects. • Struggles with detection of objects that are densely packed together.	Has very fast real-time object detection, exceptional accuracy, versatile for various applications and efficient backbone network for optimal performance.

EfficientDet	• Out of all the models that are examined, it achieves the best accuracy. • Works with considerably smaller number of FLOPS and parameters than other models. • Provides a family of EfficientDet models for different usage requirements.	• Sensitivity to input resolution. • Struggles with complex scenes.	Outstanding object detection models that offer high accuracy, efficient computation, scalability, and versatility with some limitations.

Table 5: Results on Pascal VOC and MS-COCO datasets

		Pascal VOC		MS-COCO					
Model	**BackBone**	**mAP 2007**	**mAP 2012**	**AP (.5:0.95)**	**AP 0.5**	**AP 0.75**	**AP$_S$**	**AP$_M$**	**AP$_L$**
YOLO	DarkNet	63.4	57.9	-	-	-	-	-	-
YOLO v2 (544*544)	Darknet-19	78.6	73.4	21.6	44	19.2	5	22.4	35.5
SSD (512)	VGG-16	81.6	80	26.8	46.5	27.8	9	28.9	41.9
SSD (513)	Residual-101	80.6	79.4	31.2	50.4	33.3	10.2	34.5	49.8
DSSD (513)	ResNet-101-DSSD	81.5	80	33.2	53.3	35.2	13	35.4	51.1
YOLO v3	Darknet-53	-	-	33	57.9	34.4	18.3	35.4	41.9
RetinaNet	ResNet-101-FPN	-	-	39.1	59.1	42.3	21.8	42.7	50.2
RetinaNet	ResNet-Xt-101-FPN	-	-	40.8	61.1	44.1	24.1	44.2	51.2
CornerNet (Single Scale)	Hourglass-104	-	-	40.5	56.5	43.1	19.4	42.7	53.9
CornerNet (Multi Scale)	Hourglass-104	-	-	42.1	57.8	45.3	20.8	44.8	56.7
CenterNet (Single Scale)	Hourglass-104	-	-	44.9	62.4	48.1	25.6	47.4	57.4
CenterNet (Multi Scale)	Hourglass-104	-	-	47	64.5	50.7	28.9	49.9	58.9
FCOS	ResNeXt-64x4d-101-FPN	-	-	44.7	64.1	48.4	27.6	47.5	55.6
FCOS	ResNet-101-FPN	-	-	41.5	60.7	45	24.4	44.8	51.6
YOLO v4	CSP-Darknet-53	-	-	43	64.9	46.5	24.3	46.1	55.2
EfficientDet (D4)	Efficient Net	-	81.74	49.7	68.4	53.9	-	-	-

EfficientDet (D7x)	Efficient Net	-	-	55.1	74.3	59.9	-	-	-

Comparison of models on the basis of the quantitative metric of accuracy, shows the results achieved by the models on the VOC and COCO datasets. The results shown are taken from respective original papers. Here AP = average precision, mAP = mean average precision, APs = small object precision , APm = for medium sized objects, APL= for large sized objects.

9. CONCLUSION

In the modern world, where there is a vast amount of picture and video data, object detection is extremely important and has numerous applications. Being able to automatically recognise and find items in images and videos can aid in safe navigation, threat detection, medical diagnosis, task automation, and enhanced user experiences. As a result, the creation of precise and effective object detection models has elevated in importance in computer vision research, with notable advancements made in recent years.

In this study various one-stage object detection models have been reviewed, each with their own strengths and weaknesses. It is seen that most models struggle with the detection of smaller objects, cases of densely packed or overlapping objects, processing contextual information properly and complex scenarios. It is observed that newer models tend to outperform their predecessors in terms of detection accuracy and speed but more often have higher requirements of computational resources.

Multiple models have been studied and the results show that YOLO family of models provide the best performance in terms of speed but it is found that the EfficientDet model outperformed all other models in this study. It achieves high accuracy even in detection of small objects with considerably fewer floating-point operations (FLOPs) and parameters within a good inference time.

Overall, the study highlights the rapid progress and innovation in relation to one-stage detection, and the importance of evaluating different models to determine which is the most productive and efficient for a certain application. It is felt that the findings will be useful for academics and professionals involved in computer vision and object detection.

REFERENCES

[1] Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)* (pp. 1-6). Ieee.

[2] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

[3] Dalal, N., & Triggs, B. (2005, June). Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)* (Vol. 1, pp. 886-893). Ieee.

[4] Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., & Tian, Q. (2019). Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6569-6578).

[5] Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, *111*, 98-136.

[6] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, *88*, 303-338.

[7] Felzenszwalb, P., McAllester, D., & Ramanan, D. (2008, June). A discriminatively trained, multiscale, deformable part model. In *2008 IEEE conference on computer vision and pattern recognition* (pp. 1-8). Ieee.

[8] Fu, C. Y., Liu, W., Ranga, A., Tyagi, A., & Berg, A. C. (2017). Dssd: Deconvolutional single shot detector. *arXiv preprint arXiv:1701.06659*.

[9] Ghiasi, G., Lin, T. Y., & Le, Q. V. (2019). Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 7036-7045).

[10] Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440-1448).

[11] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).

[12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[13] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84-90.

[14] Law, H., & Deng, J. (2018). Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 734-750).

[15] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980-2988).

[16] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13* (pp. 740-755). Springer International Publishing.

[17] Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8759-8768).

[18] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14* (pp. 21-37). Springer International Publishing.

[19] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, *60*, 91-110.

[20] O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., ... & Walsh, J. (2020). Deep learning vs. traditional computer vision. In *Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), Volume 1 1* (pp. 128-144). Springer International Publishing.

[21] O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

[22] Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263-7271).

[23]Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

[24] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

[25] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, *28*.

[26] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[27] Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10781-10790).

[28] Tian, Z., Shen, C., Chen, H., & He, T. (2019). Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9627-9636).

[29] Tsung-Yi, L., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117-2125).

[30] Viola, P., & Jones, M. (2001, December). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001* (Vol. 1, pp. I-I). Ieee.

[31] Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *International journal of computer vision*, *57*, 137-154.

[32] Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, *30*(11), 3212-3232.

[33] Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. *Proceedings of the IEEE*.