Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 45 No. 3 (2024)

Enhancing Object Detection with AI-Driven Deep Learning Techniques

[1]Dhivya Karunya Sampath, [2] Krishna Kumar, [3] Cynthia Anbuselvi Thangaraj

[1]*Department of Electronics & Communication Engineering, S.E.A. College of Engineering & Technology, Bengaluru, Visvesvaraya Technological University, Belagavi - 590018, India.*

[2]*Department of Electronics & Communication Engineering, Gopalan College of Engineering and Management, Bengaluru, Visvesvaraya Technological University, Belagavi - 590018, India.*

[3]*Department of Electronics & Communication Engineering, S.E.A. College of Engineering & Technology, Bengaluru, Visvesvaraya Technological University, Belagavi - 590018, India.*

Abstract: Object detection plays a crucial role in computer vision and is vital for various applications, such as autonomous vehicles. Over the past five decades, object detection techniques have advanced considerably, resulting in many innovative and successful approaches. Today, object recognition methods are broadly categorized into traditional machine learning techniques and deep learning methods. This article provides an overview of object detection techniques, beginning with a summary of traditional machine learning-based methods. It then explores two leading deep learning approaches, R-CNN and YOLO, before concluding with a comparison and discussion of these methods.

Keywords: Machine Learning, Target Detection, Computer Vision, AI, Deep Learning

1. Introduction

Until recently, the development of software and hardware for image processing mainly concentrated on user interface design, involving most of the programmers in each company. The introduction of the Windows operating system shifted this focus, directing developers' attention to image processing challenges [1]. Despite this shift, substantial progress in areas such as face recognition, license plate detection, road sign analysis, and the interpretation of remote and medical images remains limited. These persistent issues often addressed via trial and error by various engineering and scientific teams [2] . Due to the high costs of modern technical solutions, automating the creation of software tools to solve complex problems has become a priority, particularly in international contexts. In image processing, an effective toolkit should facilitate the analysis and recognition of previously unknown image content, enabling ordinary programmers to develop applications efficiently, much like the Windows toolkit supports interface creation for diverse applications [3].

Object recognition encompasses a range of computer vision tasks, including identifying objects in digital images [4]. For instance, image classification involves predicting the category of an object in an image, while object localization determines the location of objects by drawing bounding boxes around them[5] . Object detection combines these tasks by both localizing and classifying objects in an image. The terms "object recognition" and "object detection" are used interchangeably, that can be confusing for newcomers[6]..

2. Methodology

In From past fifty years, object detection methods progressed continuously, leading to numerous innovative approaches and significant advancements. Today, these techniques primarily focused into two main groups: traditional machine learning methods and deep learning methods[7].

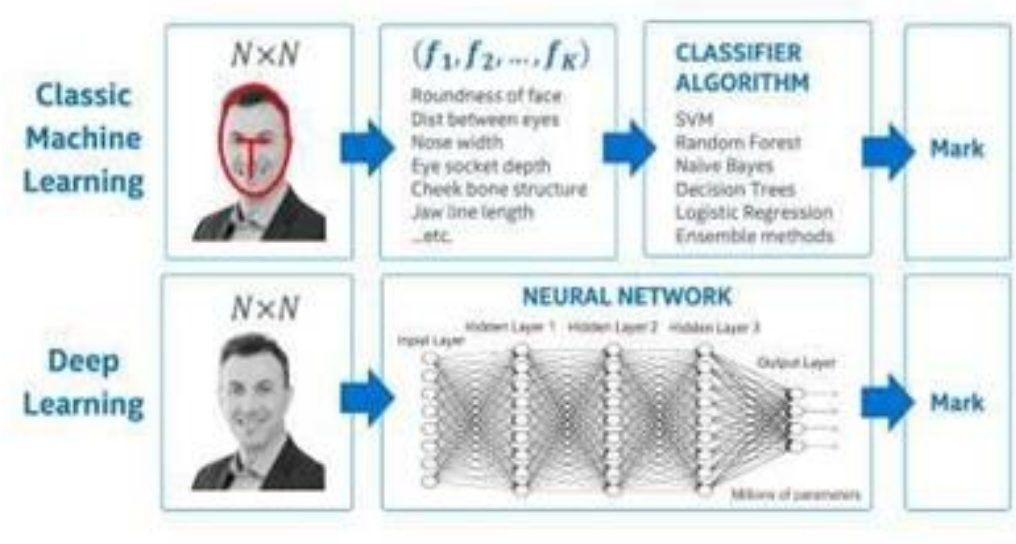

Fig 1: Machine Learning versus Deep Learning

Traditional Machine Learning Methods:

Machine learning involves study of computer algorithms which autonomously improve through experience and use of data. As a part of artificial intelligence, these algorithms construct models based on sample data, known as "training data," to make decisions or predictions without explicit programming. Machine learning techniques usually applied in activity recognition methods for detection purposes [8].

The algorithm consists of four stages:

Computation of Regions: The image is divided into pixels, and each pixel is processed as a binary number based on its intensity level.

Haar-like Feature: Human faces share common properties, which can be matched using Haar Features.

Cascading Classifiers: Based on location and size of detected features, classifiers are applied in a cascade to improve detection accuracy.

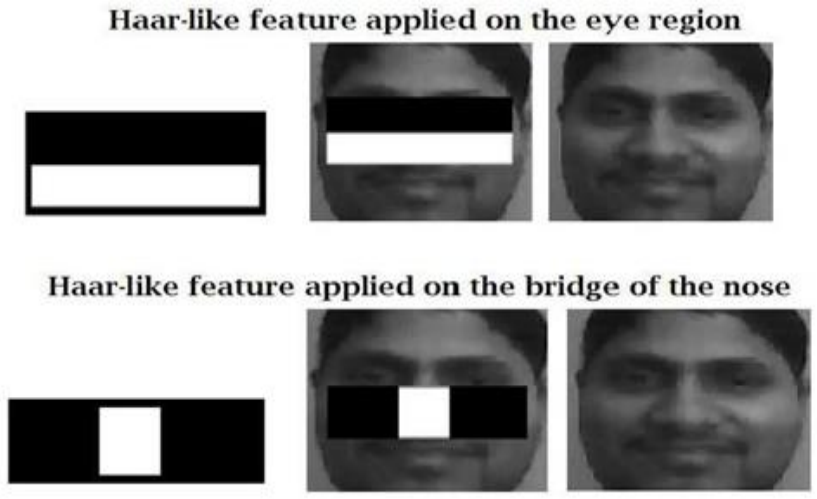

Fig 2: Haar-Like Feature applied on the eye region and bridge of the nose

Object detection: is utilized in numerous applications, like face detection, face recognition, and computer vision, where conventional algorithms may be inadequate.

Main machine learning methods used for object detection include:Viola–Jones Object Detection Framework,

Scale-Invariant Feature Transform (SIFT), Histogram of Oriented Gradients (HOG) Features.

Viola–Jones Object Detection Framework

The Viola–Jones object detection framework, developed by Paul Viola and Michael Jones in 2001, was primarily designed to address problem of face detection, although it can be trained to recognize various object classes. Detecting faces in an image is a complex task for computers, requiring precise instructions and constraints. To simplify the process, Viola–Jones focuses on detecting full view, frontal, upright faces, meaning the face must be directly facing the camera and not tilted to either side [9].

Despite these constraints potentially limiting the algorithm's utility, they are quite acceptable in practice since the detection step is usually followed by a recognition step. The key characteristics that make the Viola–Jones algorithm effective for detection include:

Selection of Haar like features, Creation of Integral Image, Running AdaBoost Training, and Creating Classifier Cascades.

In Figure 2, the algorithm cascades through pixels of the eyes, mouth, and nose, noting their positions and organizing them according to pixel intensity. The recognized features are searched for within face image. Pixel values are then converted from binary to decimal based on their intensity.

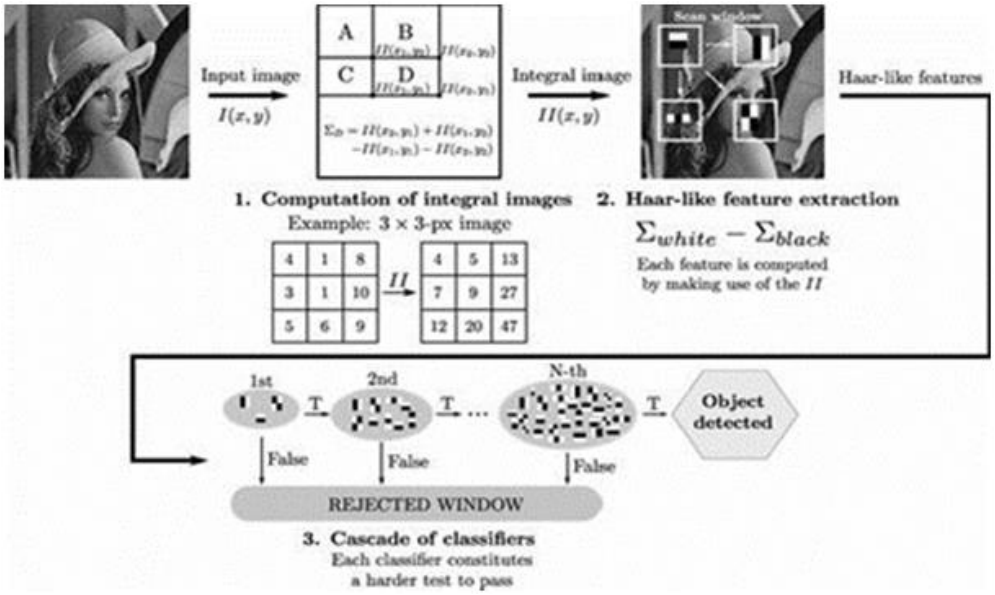

Fig 3: Viola- Jones algorithm

Scale-Invariant Feature Transform (SIFT)

It is a technique in computer vision for identifying and describing local features in images. This method used in 3D modeling by comparing features from a new image with database of features and identifying potential matches based on the Euclidean distance between their feature vectors. The process involves selecting subsets of matches, such as those found using Histogram of Oriented Gradients (HOG).

The histogram of oriented gradients (HOG)

This is a feature descriptor utilized in computer vision and image processing to identify objects. Accuracy of descriptor fit and number of likely false matches are assessed. Matches that meet all criteria are identified as correct with high confidence.

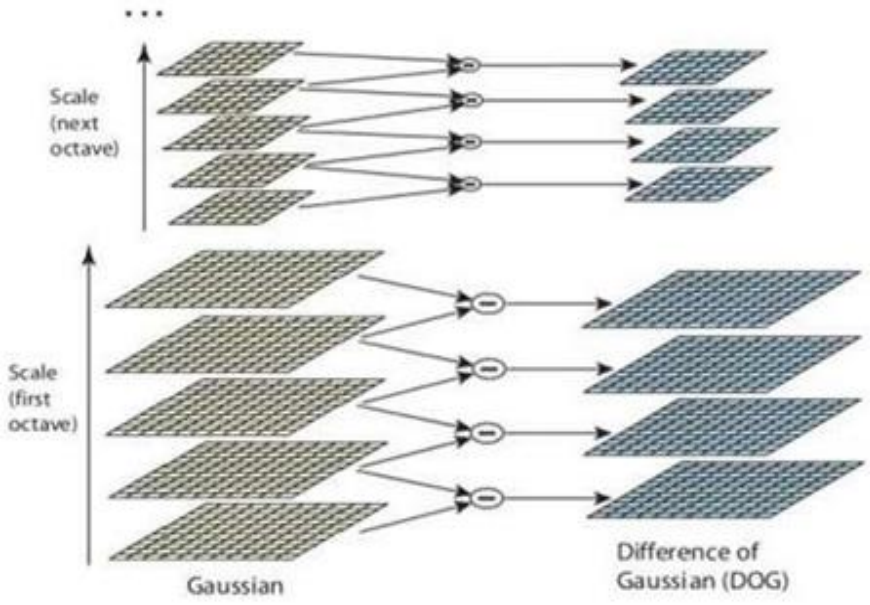

Fig 4:Scale-Invariant Feature Transform (SIFT)

The operation works as follows: Objects are initially extracted from set of reference images, stored in database. To recognize an object in new image, every feature from new image is compared to this database, identifying candidate matches based on Euclidean distance. From the complete set of matches, key points that align with the object's location and orientation in the new image are identified to filter out incorrect matches.

This clustering of matches is efficiently performed using a hash table implementation of the generalized Hough transform. The likelihood that a set of features represents an object will be calculated based on fit accuracy and number of potential false matches. This technique involves counting occurrences of gradient orientation in localized portions of image, similar to edge orientation histograms, scale-invariant feature transform descriptors, and shape contexts, but it operates on dense grid of uniformly spaced cells and engages overlapping local contrast normalization for enhanced accuracy.

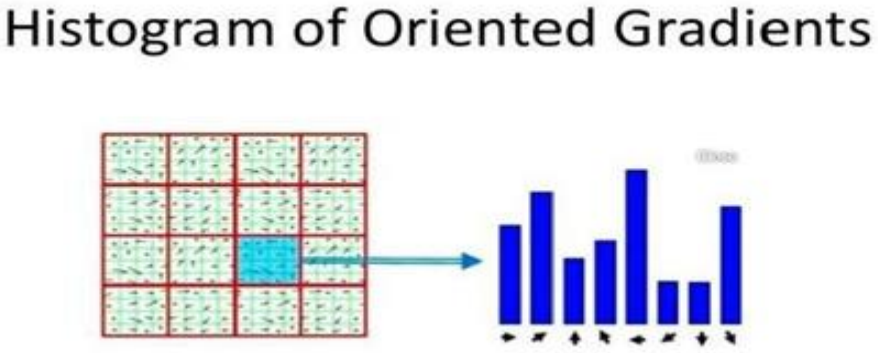

Fig 5: Histogram of Oriented Gradients (HOG)

The algorithm consists of four stages:

Gradient Computation: first step involves normalizing color and gamma values to ensure consistency.

Orientation Binning: The second step creates cell histograms. Every pixel within a cell contributes weighted vote to the histogram. The cell is rectangular, with histogram channels distributed evenly across 0 to 180 degrees or 0 to 360 degrees.

Descriptor Blocks: To account for variations in illumination and contrast, gradient strengths that will be locally normalized by grouping cells into larger, spatially connected blocks.

Object Recognition: Histogram descriptors used for object recognition by providing them into a machine learning algorithm.

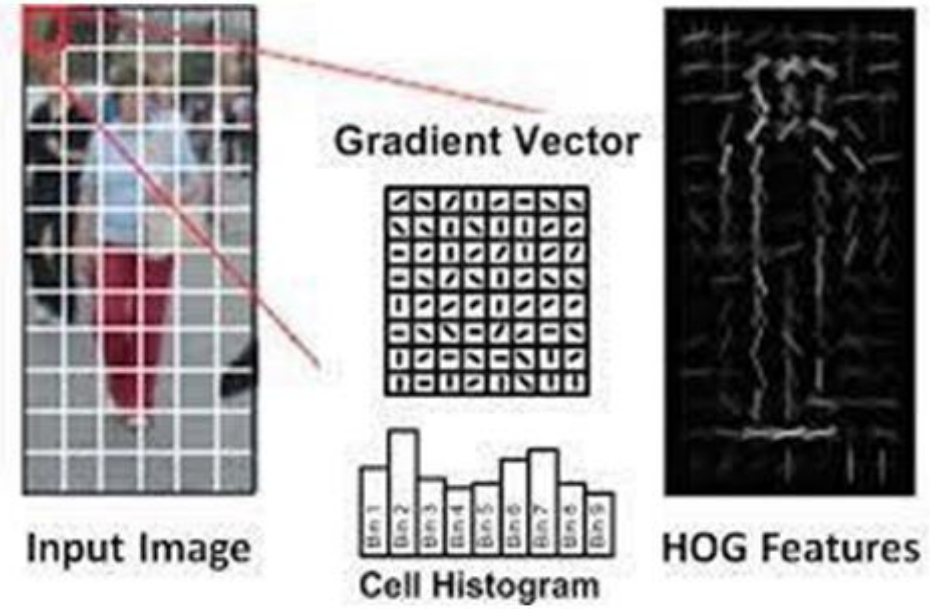

Fig6: Histogram of Oriented Gradients (HOG) algorithm

Deep learning Methods:

It's a part of machine learning techniques that relys on artificial neural networks with portrayal learning. This type can be seen as supervised Learning, semi-supervised Learning, or unsupervised Learning [11].

Deep learning architectures, like graph neural networks, deep belief networks, convolutional neural networks, deep neural networks, and recurrent neural networks, have been applied to various fields [12]. These include medical image analysis, audio recognition, machine vision, computer vision, speech recognition, material inspection, natural language processing, social network filtering, machine translation, drug design, board game programs and bioinformatics often achieving or surpassing human expert performance [13].

Object Detection Using Deep Learning

The primary approaches include:

i) Region-Based Convolutional Neural Network (R-CNN)

ii) You Only Look Once (YOLO)

iii) Deformable Convolution Networks

Region-Based Convolutional Neural Networks (R-CNN):

The original objective of R-CNN was to take an input image and produce bounding boxes to each object in image, along with category (e.g., car or pedestrian) of each object. R-CNN operates as follows:

Extracting Regions of Interest (ROI): Using Selective Search, R-CNN extracts rectangular ROIs from the input image that might contain objects.

Feature Generation: Each ROI is fed into a neural network, which generates feature outputs.

Object Classification: Support-vector machine classifiers use these features to recognize type of object within each ROI.

Steps for R-CNN to Detect Objects:

Pre-trained Convolutional Neural Network: Start with a pre-trained CNN and retrain it for the specific task.

Training the Last Layer: Train the network's last layer based on number of classes to be discovered.

Determining ROIs: Identify Region of Interest for each image.

Restructuring Regions: Resize these regions in order to match the CNN input size.

Classifying Objects and Background: Train SVMs to differentiate objects from the background.

Bounding Box Refinement: Train linear model to produce dense bounding boxes for each identified object.

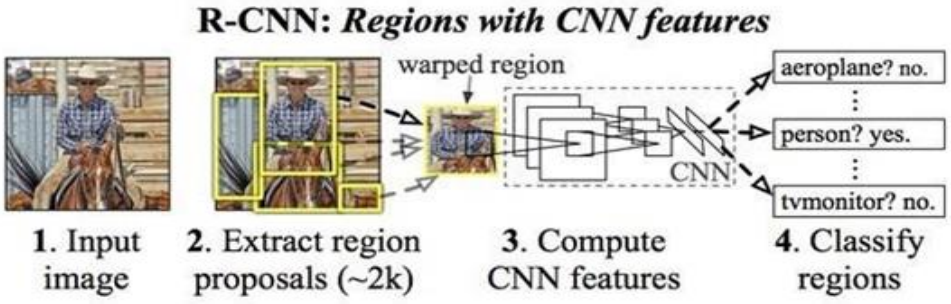

Fig 7: Region Based Convolution Neural Networks(R-CNN) applied to an image

You Only Look Once (YOLO)

It employs a unique approach to object detection. YOLO is an innovative convolutional neural network designed for object detection in real-time [14]. The algorithm processes entire image with single neural network, dividing image into regions and predicting bounding boxes and probabilities for each region. These anticipated probabilities are used to weight the bounding boxes [15].

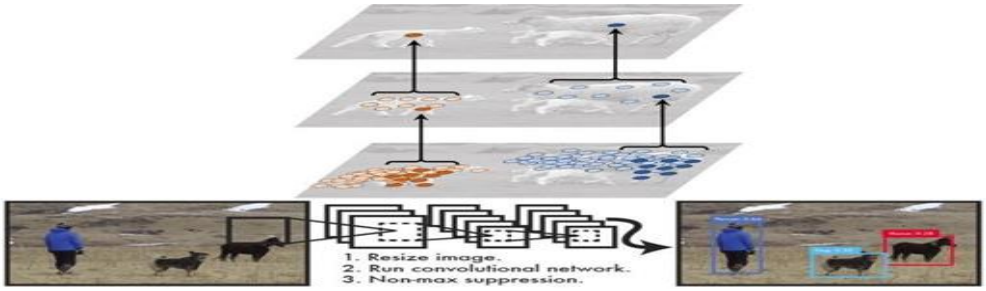

Fig 8: You Only Look Once (YOLO) applied to an image

YOLO develops generalizable representations of objects, allowing it to outperform other leading detection methods when trained on natural images and tested on artwork [16].

Deformable Convolution Networks:

Deformable Convolution Networks arises from its ability to adapt to the geometric variations of objects. Through an examination of its adaptive behavior, we observe that while the spatial support for its neural features conforms more closely than regular convolution to object structure, this support may nevertheless extend well beyond the region of interest, causing features to be influenced by irrelevant image content. To look into this problem, here we present formulation of Deformable Convolution that improves its potential to concentrate on pertinent image regions, by increased modeling power and powerful training. The modeling power is enhanced through a more comprehensive integration of deformable convolution within the network, and by introducing a modulation mechanism that expands the scope of deformation modeling. To effectively harness this enriched modeling capability, we guide network training via a proposed feature mimicking scheme that helps the network to learn features that reflect the object focus and classification power of R-CNN features. This improved version of Deformable ConvNets achieves considerable performance time. The approach takes only one forward

propagation run through the neural network to make predictions, so it "only looks once" at the image.

Fig9: Deformable Convolution Networks applied to an image

3. Evaluation

We first introduce the object detection technique using traditional machine learning and deep learning. However, it is hard to compare which one is better because the answer is based on selection of specific scenarios. Unconventional approaches has been employed to provide solution on the needs for accurate object detection models. The recent popularization of GPU-accelerated deep-learning frameworks, convolution neural networks and object- detection algorithms are initiated from a new perspective. CNNs like R-CNN, Fast R-CNN, Faster R-CNN, R-FCN and YOLO have highly increased the performance standards on the field.

Once if we train the first object detector, then next step is to evaluate its performance. While it may seem the model is successfully identifies all objects in images provided, a more thorough assessment is necessary. Unlike classification, which simply evaluates likelihood of an object's presence in the image, object detection involves localizing the object with bounding box and assigning a confidence score to indicate the conclusiveness of the detection. Object detection benchmarks are pivotal in evaluating model performance for object detection assignments. They permit unbiased comparisons between different detection systems or against a benchmark. In competitions, metrics such as average precision and its variations are used to evaluate and rank detections.

To thoroughly analyze model performance, it is recommended to use both validation set for tuning hyper parameters and test set for assessing the execution of a fully-trained model. Object detection is fetching importance across various industries, from personal security to workplace productivity. It is adapted in numerous areas of computer vision, like image retrieval, security, surveillance, automated vehicle systems, and machine inspection. Despite its advancements, object recognition still faces significant challenges. The potential for future applications of object detection is vast. Here are some current and future applications in detail:

Optical Character Recognition:

Optical character recognition, or OCR, is the mechanical or electronic conversion of images of typed, hand written, or printed text into machine-encoded text, whether from a scanned document, a photograph of a document, or a scene photograph (for example the text on signs and billboards in a landscape photo) or from subtitle text superimposed on an image, wear extracting characters from the image or video.

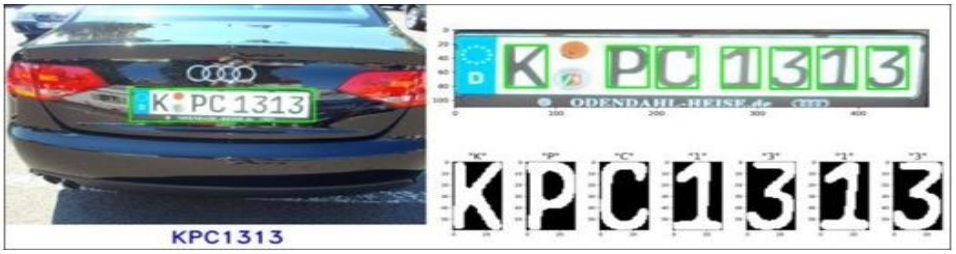

Fig 10: Optical Character Recognition used to detect license plate number

Self Driving Cars

For autonomous driving, one of the finest examples of why you need object detection is in order for an automobile to decide whether to accelerate, brake, or turn in the following step, it must first know where all of the objects around it and what they are. This necessitates object detection, and we'd effectively train the car to recognize a known collection of items such as cars, pedestrians, and other vehicles.

Fig 11: Self Driving Cars

4. Results and Discussion

Following were some limitations that were observed in the three models

SSD

For smaller objects, SSD's performance is significantly inferior to that of Faster R-CNN. This is primarily because SSD relies on higher resolution layers for detecting small objects, but these layers mainly capture lower-level features like color patches or edges, which diminishes SSD's classification accuracy. Additionally, the complexity of SSD's data augmentation highlights another limitation: it demands a substantial amount of training data, making the process costly and time-consuming depending on the application.

Faster R-CNN

The accuracy of this algorithm comes at the expense of time complexity, making it notably slower compared to models like YOLO. Although it offers improvements over RCNN and Fast RCNN, it still requires multiple passes over a single image, unlike YOLO. FRCNN consists of several components—the convolutional network, Regions of Interest (ROI) pooling layer, and Region Proposal Network (RPN)—any of which can become a bottleneck for the others.

YOLO

YOLOv3 was one of the most significant advancements in object detection since the introduction of Darknet 53, receiving praise from critics and industry professionals alike. However, it had its limitations. Despite being considered a strong performer, complexity analysis revealed flaws, particularly in its loss function, which lacked optimal solutions. These issues were later addressed in an optimized version, leading to enhanced functionality. A new software version often highlights the shortcomings of its predecessor. Upon reviewing the YOLOv4 paper, it's clear that YOLOv3 struggled with images featuring multiple elements that weren't the main focus, resulting in accuracy issues, especially with smaller images, where the accuracy was only about 16%, as our data showed. Moreover, YOLOv4 introduced CSPDarknet-53, an improvement over Darknet-53, which used 66% fewer parameters while delivering better speed and accuracy. Precision-recall curves plotted using the COCO metric and API provided insights into the performance of these models in object detection. The orange-shaded areas on the graphs represent precision-recall curves without errors, violet areas show falsely detected objects, blue areas indicate localization errors (Loc), white areas under the curve denote an IoU value greater than 0.75, and grey areas indicate an IoU value greater than 0.5.

it is evident that region-based detectors like F R-CNN and SSD exhibit lower accuracy, as seen in their larger violet areas. However, F R-CNN is more accurate than SSD, while SSD is more efficient for real-time processing due to its higher mAP values. YOLO stands out as the most efficient model, with minimal violet regions, indicating its superior performance in object detection.

Table 1 COCO metrics

Average precision (AP):		
AP	% AP at IoU	.50:.05:.95 (primary challenge metric)
AP(IoU = .50)	% AP at IoU	.50 (PASCAL VOC metric)
AP(IoU = .75)	% AP at IoU	.75 (strict metric)
AP across scales		
AP (small)	% AP for small objects	Area < 322
AP (medium)	% AP for medium objects	322 < area < 962
AP (large)	% AP for large objects	Area > 962

Average Recall (AR)		
AR (max=1)	% AR given 1 detection per image	
AR (max=10)	% AR given 10 detections per image	
AR (max=100)	% AR given 100 detections per image	
AR across scales		
AR (small)	% AR for small objects	Area < 322
AR (medium)	% AR for medium objects	322 < area < 962
AR (large)	% A for large objects	Area > 962

Nº	IoU	Area	Max Dets	Average precision			Average recall		
				SSD	YOLO	FRCNN	SSD	YOLO	FRCNN
1	0.50:0.95	All	100	0.247	0.337	0.716	0.232	0.279	0.782
2	0.50	All	100	0.424	0.568	0.873	0.341	0.432	0.754
3	0.75	All	100	0.253	0.350	0.851	0.362	0.460	0.792
4	0.50:0.95	Small	100/1*	0.059	0.152	0.331	0.102	0.257	0.567

Nº	IoU	Area	Max Dets	Average precision			Average recall		
				SSD	YOLO	FRCNN	SSD	YOLO	FRCNN
5	0.50:0.95	Med	100	0.264	0.359	0.586	0.401	0.494	0.653
6	0.50:0.95	Large	100	0.414	0.496	0.846	0.577	0.623	0.893

5. Conclusion

This paper provides an overview of key methods in object detection, including both traditional machine learning approaches and recent advancements in deep learning. Object detection is crucial for various computer and robotic vision systems, such as face detection for auto-focus in smart phones and integration into driver assistance technologies. This review article compared the latest and most advanced CNN-based object detection algorithms, highlighting their critical role in analyzing the vast number of images uploaded to the internet daily [42]. Object detection is also essential for technologies like self-driving vehicles that rely on real-time analysis. All the networks were trained using the open-source COCO dataset by Microsoft to maintain a consistent baseline. The findings revealed that YOLOv3 is the fastest, with SSD close behind, while Faster R-CNN is the slowest. However, the choice of algorithm depends on the specific use case: for smaller datasets without the need for real-time results, Faster R-CNN is the best option; for live video feed analysis, YOLOv3 is ideal; and SSD offers a good balance between speed and accuracy. Additionally, YOLOv3, being the most recent of the three, benefits from ongoing contributions by the open-source community. Therefore, YOLOv3 demonstrates the best overall performance among the three object detection CNNs analyzed, aligning with previous findings.

References

[1] AR. Young “A review of spiking neuromorphic hardware communication systems”, IEEE2019.

[2] JKhun “High-performance spiking neural network simulator” , IEEE2019.

[3] G M aranhão, “Integrate and fire neuron implementation using cmos predictive technology model for 32nm”, IEEE 2019.

[4] CD Schuman “A survey of neuromorphic computing and neural networks in hardware”, IEEE 2017.

[5] A Review of Object Detection Techniques, XinruiZou, Southwest Jiao Tong University, Sichuan 611756, China, IEEE 2019

[6] An Innovative Machine Learning Approach for Object Detection and Recognition, AishwaryaSarkale, Kaiwant shah, Anandji Chaudhary, Prof. Tatwadarshi, IEEE2019

[7] Object Detection with Deep Learning: A Review,Zhong-QiZhao, Member, IEEE, Peng Zheng, Shou-Tao Xu, and Xindong Wu, Fellow, IEEE 2019

[8] Small Object Detection Based on Deep Learning, Wei Wei Beijing Key Laboratory of Information Service Engineering College of Robotics Beijing Union University Beijing, China, IEEE 2020

[9] An Overview of Deep Learning Based Object Detection Techniques, Bhagya C and Prof. ShynaA, IEEE2020

[10] 3D Object Detection Based on LiDAR Data, RaminSahba, Amin Sahba, MoJamshidi, PaulRad, IEEE2019.

[11] DhivyaKarunya S, Krishna kumar “Human Activity Recognition Methods” in International Journal of Engineering and Advanced Technology (IJEAT), ISSN:2249-8958 , Volume -9 Issue-5, June 2020, Page No. 1024-1028.

[12] DhivyaKarunya S and Krishna kumar, “Suspicious Action Detection and Recognition in Remote Areas

Using AI and Machine Learning Techniques" in IJO- International Journal of Computer Science and Engineering, Volume - 4, Issue-12, December 2021.

[13] DhivyaKarunya S and Krishna kumar, "Development of Detection and Recognition of Abnormal Human Crowd Behavior Using Supervised Entropy Technique " in Gradiva Review Journal, ISSN:0363-8057 (Online), Volume -8 Issue-5, May 2022.

[14] DhivyaKarunya S and Krishna kumar "Abnormal Crowd Behaviour Detection in Surveillance Videos Using Spatiotemporal Inter-Fused Autoencoder" in International Journal of Intelligent Engineering and Systems, Vol.16, No.6, 2023 DOI: 10.22266/ijies2023.1231.39

[15] DhivyaKarunya S, Krishna kumar "Development of detection and recognition of human activity in sports using GMM and CNN algorithms" in International Journal of System of Systems Engineering, DOI:10.1504/IJSSE.2024.10058138

[16] DhivyaKarunya S, Krishna kumar, "A GSM-Based System for Vehicle Collision Detection and Alert" on the IEEE conference held in East Point College of Engineering & Technology.