A Survey on Object Detection in Computer Vision: From Early Methods to Deep Learning and Beyond

*Note: Sub-titles are not captured in Xplore and should not be used

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

***Abstract*—Object detection is a cornerstone of computer vision, underpinning diverse applications such as autonomous driving, medical imaging, robotics, and intelligent surveillance. Over the past two decades, the field has evolved from traditional hand-crafted pipelines to deep learning-based detectors and, more recently, transformer-driven and generative paradigms. This survey provides a comprehensive and structured overview of object detection, unifying developments across three major eras: pre-deep learning, convolutional neural network (CNN)–based, and transformer-based approaches. We systematically analyze key architectures including the R-CNN and YOLO families, anchor-free and hybrid designs, and recent advances in vision–language and generative (GAN/Diffusion) detection. In addition, we review benchmark datasets, lightweight and 3D detection, and emerging challenges such as open-world generalization, efficiency on edge devices, and fairness. By integrating historical insights with modern trends, this survey highlights conceptual transitions, unresolved challenges, and future directions toward unified, multimodal, and foundation-level perception models.**

***Index Terms*—object detection , survey , computer vision**

I. Introduction

Object detection stands as one of the most fundamental and enduring problems in computer vision, serving as a cornerstone for visual perception and intelligent decision-making. It aims to simultaneously identify and localize objects within an image or video frame, predicting both the category and spatial coordinates of each instance. Unlike image classification—which assigns a single label to an entire image—or semantic segmentation—which provides dense pixel-level labeling, object detection bridges the gap between recognition and understanding by capturing both what an object is and where it appears. This dual objective of classification and localization makes object detection an essential building block for high-level visual reasoning tasks such as scene parsing, activity recognition, and tracking.

The importance of object detection extends across a wide range of real-world applications. In autonomous driving, detection systems enable vehicles to perceive pedestrians, traffic signs, and obstacles, forming the perceptual basis for safe decision-making. In medical imaging, object detection facilitates the identification of tumors, lesions, and other abnormalities across modalities such as X-ray, CT, and MRI scans, aiding early diagnosis and treatment planning. Robotics and industrial automation rely on detection for grasping, manipulation, and navigation in dynamic environments. In surveillance and security, it supports anomaly detection, crowd monitoring, and event understanding. These diverse use cases collectively demonstrate that object detection is not merely a perception problem—it is the visual intelligence layer that enables actionable decisions in complex systems.

Historically, the development of object detection has progressed through several distinct paradigms, each redefining how visual patterns and spatial relations are modeled. The pre-deep learning era was dominated by hand-crafted feature representations and classical machine learning classifiers. Pioneering works such as the Viola–Jones detector introduced Haar-like features and cascade classifiers for real-time face detection, while the HOG + SVM and Deformable Part Models (DPM) frameworks improved robustness to pose and appearance variation. Despite their success, these approaches suffered from scalability and generalization limitations due to their reliance on manually designed features.

The advent of deep learning revolutionized object detection by enabling end-to-end feature learning directly from raw pixels. Convolutional Neural Networks (CNNs) became the dominant paradigm, with architectures such as AlexNet, VGG, and ResNet providing hierarchical feature representations. Two-stage detectors, including the R-CNN family (R-CNN, Fast, Faster, and Mask R-CNN), achieved unprecedented accuracy by combining region proposals with deep feature extraction. Subsequently, one-stage models such as YOLO, SSD, and RetinaNet prioritized real-time performance through unified detection pipelines. Anchor-free detectors like FCOS and CenterNet further simplified design complexity, improving generalization while reducing hyperparameter dependency. This era marked a paradigm shift from handcrafted features to data-driven representation learning.

The latest wave of innovation has been driven by **transformer-based architectures**, which leverage self-attention mechanisms to model long-range dependencies and

Identify applicable funding agency here. If none, delete this.

global context. DETR (DEtection TRansformer) reframed detection as a direct set prediction problem, eliminating hand-engineered components such as anchors, proposals, and non-maximum suppression. Its successors, including Deformable DETR, Conditional DETR, Sparse R-CNN, and hybrid CNN-transformer frameworks, have addressed challenges of slow convergence and small-object sensitivity. Beyond purely visual modeling, transformer-based detectors have enabled vision–language integration, with models like GLIP, OWL-ViT, and MDETR supporting open-vocabulary and zero-shot detection through large-scale multimodal pretraining.

More recently, a growing line of research has explored generative paradigms for object detection, reinterpreting detection as a synthesis or denoising process. GAN-based approaches have been used for small-object enhancement, domain adaptation, and data augmentation, while diffusion-based frameworks such as DiffusionDet and Pix2Seq-Diffusion reformulate detection as an iterative denoising process, bridging the gap between discriminative and generative learning.

The evolution of object detection thus reflects a broader transition in computer vision—from localized feature heuristics toward unified, context-aware, and generative perception systems. Each paradigm has not only improved accuracy and efficiency but also transformed the conceptual understanding of how objects and scenes are represented.

Contributions of this Survey: Given the rapid and multi-dimensional progress of the field, this survey provides a comprehensive and structured overview of object detection methods, offering several unique contributions:

- **Unified Taxonomy**: We propose a three-level taxonomy spanning pre-deep learning, CNN-based, and transformer-based approaches, highlighting conceptual, architectural, and chronological transitions that have shaped the evolution of detection systems.
- **In-depth Analysis of the YOLO Family**: We provide the most extensive survey to date on the YOLO series (v1–v8), examining its architectural innovations, backbone evolution, and deployment efficiency for real-time and edge scenarios.
- **Transformer-based Detection**: We analyze recent transformer-driven models, from vanilla DETR to hybrid and multimodal designs, emphasizing how attention mechanisms and vision–language pretraining redefine detection paradigms.
- **Generative and Diffusion-based Detection**: We cover emerging generative frameworks, including GAN- and diffusion-based approaches, that view object detection as a probabilistic or generative process.
- **Datasets and Evaluation**: We review the major datasets and benchmarks (e.g., VOC, COCO, Open Images, KITTI, nuScenes) and discuss their influence on architectural design, performance evaluation, and generalization.
- **Challenges and Future Trends**: We outline open problems—small object detection, domain shift, efficiency on edge devices, and ethical concerns—and highlight research directions such as self-supervised learning, foundation models, and hardware-aware neural architecture search.

Together, these contributions establish this survey as a comprehensive bridge between classical, deep, and emerging paradigms in object detection. By integrating historical insights with modern trends, it aims to provide a cohesive understanding of the field and inspire future advancements toward unified and intelligent perception systems.

II. Pre-Deep Learning Era

Before the advent of deep learning, object detection relied on manually designed features and conventional machine learning classifiers. A standard detection pipeline followed a sliding window paradigm, in which fixed-size windows were densely scanned across an image at multiple scales and aspect ratios. For each window, hand-crafted descriptors such as Haar, HOG, or SIFT features were extracted and fed into classifiers such as Support Vector Machines (SVMs) or boosted decision trees to determine object presence. This design provided a well-defined computational framework and enabled the first generation of practical detectors. However, it was also computationally demanding—requiring thousands of evaluations per image—and lacked robustness to appearance variation. Despite these limitations, this paradigm dominated object detection research from the early 2000s until the rise of deep learning, laying the conceptual foundation for modern detection architectures.

A. Viola–Jones Framework

The **Viola–Jones framework** (2001) marked one of the first major breakthroughs in real-time object detection and became a foundational model for early computer vision systems. It introduced three core innovations that collectively defined the modern detection pipeline:

1) **Haar-like features**, which represented local intensity patterns using simple rectangular filters;
2) the **integral image**, enabling extremely fast computation of these features at multiple scales; and
3) a cascade of **AdaBoost** classifiers, designed to progressively reject negative windows while retaining promising candidates.

This hierarchical filtering mechanism drastically reduced computational cost, making real-time face detection feasible even on low-power CPUs—an unprecedented achievement at the time. The Viola–Jones detector was widely adopted in surveillance, biometrics, and consumer electronics, serving as the first practical example of algorithmic efficiency in vision-based perception.

However, its performance was tightly coupled to rigid object categories such as frontal human faces and deteriorated sharply under variations in pose, illumination, and background clutter. Moreover, the reliance on manually designed Haar features limited its ability to generalize to complex object classes. Despite these drawbacks, the Viola–Jones framework established several design principles—feature extraction, hierarchical filtering, and computational efficiency—that would

later influence the development of feature pyramids, cascaded CNNs, and lightweight real-time detectors like YOLO.

B. HOG + SVM

The **Histogram of Oriented Gradients (HOG)** descriptor, introduced by Dalal and Triggs (2005), represented a major milestone in traditional object detection. Instead of relying on raw intensity or simple rectangular filters, HOG captured the local structure of an object by encoding the distribution of gradient orientations within spatially localized cells and overlapping blocks. This representation effectively preserved edge and contour information, making it robust to illumination variations, small geometric deformations, and background noise.

Combined with a **linear Support Vector Machine (SVM)** classifier, HOG achieved remarkable performance on the **INRIA Pedestrian Dataset**, setting a new benchmark for pedestrian detection and inspiring a decade of research into gradient-based descriptors. The method's strength lay in its balance between accuracy and interpretability, enabling systematic analysis of feature responses across scales and orientations.

However, the approach still required an exhaustive **sliding window** search across multiple scales, which imposed significant computational overhead. Furthermore, it struggled under severe occlusion, large viewpoint changes, and highly deformable object classes. Despite these limitations, the HOG + SVM pipeline established several enduring principles—feature normalization, block-wise representation, and discriminative linear classification—that directly influenced subsequent models such as **Deformable Part Models (DPM)** and, conceptually, early convolutional feature hierarchies in deep learning.

C. Deformable Part Models (DPM)

The **Deformable Part Model (DPM)**, proposed by Felzenszwalb et al. (2008), marked a pivotal shift in traditional object detection by introducing a structured representation of objects as compositions of parts. Each object was modeled using a root filter capturing global appearance and multiple part filters defined over HOG descriptors, arranged in a spatial configuration that allowed limited deformation. The model's formulation, expressed through a latent SVM framework, optimized both appearance and geometric alignment by balancing part match scores with a learned deformation cost.

This design endowed DPMs with remarkable robustness to variations in pose, viewpoint, and articulation—properties that earlier holistic detectors lacked. As a result, DPMs achieved state-of-the-art performance on benchmarks such as **PASCAL VOC (2007–2012)** and became the dominant detection paradigm before the emergence of deep learning.

However, despite their conceptual elegance, DPMs were computationally expensive, relied heavily on hand-crafted features, and required complex parameter tuning to balance deformation and appearance components. Moreover, their reliance on fixed HOG-based descriptors limited scalability to large datasets and complex scenes.

Nevertheless, the hierarchical and compositional principles established by DPM—combining local part evidence into global object hypotheses—directly anticipated the layered feature hierarchies later realized in **Convolutional Neural Networks (CNNs)**. In retrospect, DPM represents the conceptual bridge between classical structured vision models and the modern era of deep, end-to-end trainable detectors.

D. Comparative Analysis and Limitations

As summarized in **Table I**, classical object detection methods—from **Viola–Jones** to **HOG+SVM** and **Deformable Part Models (DPM)**—reflects a steady shift from holistic intensity-based representations toward structured, part-aware models. Each approach introduced key innovations that advanced detection accuracy and efficiency; however, all shared inherent limitations due to their reliance on hand-crafted features and separate optimization stages.

In comparison, **Viola–Jones** prioritized speed through feature cascades, **HOG+SVM** improved robustness by leveraging gradient structure, and **DPM** introduced compositional modeling for deformation handling. Yet, despite these advances, all approaches suffered from common weaknesses:

1) Limited scalability due to exhaustive sliding-window evaluation;
2) Sensitivity to illumination and geometric changes;
3) High computational cost; and
4) Poor generalization across domains.

These shortcomings stemmed from the fragmented design of traditional pipelines, where feature extraction, classification, and post-processing were optimized independently. The inability to learn hierarchical representations end-to-end prevented these systems from scaling to diverse object categories and large datasets.

Collectively, this era established the conceptual building blocks of modern detection—feature representation, discriminative classification, and multi-scale processing—while simultaneously revealing the necessity for integrated, data-driven learning. These limitations ultimately catalyzed the transition toward deep convolutional neural networks (CNNs), which unified feature learning and object localization into a single, end-to-end framework.

III. Deep Learning-based Approaches

The advent of deep learning fundamentally transformed object detection by allowing models to learn hierarchical feature representations directly from raw pixels, eliminating the need for manually engineered descriptors such as Haar or HOG. The watershed moment came with AlexNet (Krizhevsky et al., 2012), which achieved a dramatic improvement in the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) and showcased the effectiveness of **convolutional neural networks (CNNs)** for large-scale image understanding.

Unlike traditional pipelines that separated feature extraction and classification, CNNs unified these processes within a single, trainable architecture, enabling spatial abstraction and semantic compositionality through stacked convolutional

TABLE I
COMPARISON OF CLASSICAL OBJECT DETECTION FRAMEWORKS PRIOR TO DEEP LEARNING

Method	Key Idea	Strengths	Limitations
Viola–Jones (2001)	Haar-like features + cascade of AdaBoost classifiers	First real-time face detector; simple and efficient on CPUs	Limited to rigid objects; sensitive to illumination and pose
HOG + SVM (2005)	Gradient orientation histograms + linear SVM	Robust to small deformations and illumination; interpretable features	Computationally heavy; poor under occlusion and viewpoint changes
DPM (2008)	Part-based model with latent SVM using HOG descriptors	Handles pose variation and partial occlusion; strong accuracy on VOC	Complex training; slow inference; relies on hand-crafted features

layers. Subsequent architectures—**VGGNet** (Simonyan and Zisserman, 2014) and **ResNet** (He et al., 2016)—further deepened the representational hierarchy and introduced innovations such as residual learning, improving both accuracy and generalization.

These advances directly addressed the major limitations of pre-deep-learning detectors: scalability through shared convolutional computation, robustness through learned invariances, and generalization via large-scale data-driven optimization. Consequently, the object detection landscape shifted from hand-crafted pipelines to end-to-end differentiable systems, laying the foundation for modern frameworks such as **R-CNN, YOLO, and SSD**.

A. Convolutional Backbones and Attention in CNNs

The evolution of convolutional neural networks (CNNs) provided the structural backbone for modern object detectors. Early architectures such as **AlexNet (2012)** and **VGGNet (2014)** established the concept of hierarchical feature extraction through stacked convolutional layers, while **ResNet (2016)** introduced residual connections to mitigate vanishing gradients and enable deeper, more expressive representations. Subsequent designs like **Inception** and **EfficientNet** emphasized architectural efficiency and parameter optimization, forming the foundation for popular detection backbones including **ResNet-FPN, CSPDarknet, and MobileNet**.

Despite their success, convolutional operations are inherently local, limiting the ability of CNNs to capture long-range dependencies and global contextual relationships. To overcome this, **attention mechanisms** were introduced as lightweight yet effective modules for enhancing feature representation by adaptively reweighting spatial or channel information.

The **Squeeze-and-Excitation Network (SENet)** (Hu et al., 2018) pioneered channel attention, dynamically recalibrating feature responses through a global context descriptor with minimal computational overhead. The **Convolutional Block Attention Module (CBAM)** (Woo et al., 2018) extended this concept by combining both channel and spatial attention, refining feature maps in a context-aware manner. Later, **Efficient Channel Attention (ECA-Net)** (Wang et al., 2020) further simplified this process by removing dimensionality reduction and employing lightweight 1D convolutions to maintain efficiency without sacrificing accuracy.

These attention modules have been seamlessly integrated into detection backbones and necks—such as **ResNet-SE, CSPDarknet-ECA, and PANet with CBAM**—where they enhance representational quality and boost detection accuracy with negligible computational cost. By improving feature selectivity and contextual awareness, attention mechanisms effectively bridge the gap between conventional CNNs and transformer-based architectures, paving the way for more adaptive and globally coherent object detectors.

B. Two-stage Detectors

Two-stage detectors introduced a proposal-based pipeline that fundamentally redefined object detection. Instead of exhaustively scanning the entire image with sliding windows, these methods first generate a limited set of **region proposals** likely to contain objects and then perform classification and bounding-box regression on each proposal. This hierarchical process significantly reduced redundant computation while improving localization accuracy.

Powered by convolutional neural network (CNN) backbones, two-stage frameworks such as the **R-CNN family** achieved unprecedented detection performance and became the dominant paradigm for nearly half a decade. By decoupling proposal generation and object classification, they established the architectural foundation upon which most modern detectors—including one-stage and transformer-based models—were later built.

1) R-CNN (Region-based Convolutional Neural Network): The R-CNN framework, proposed by Girshick et al. (2014), marked the first successful integration of convolutional neural networks (CNNs) into object detection. It followed a two-stage pipeline consisting of:

1) Region proposal generation using Selective Search to identify approximately 2,000 candidate regions per image;
2) Feature extraction by forwarding each region through a pre-trained CNN (e.g., AlexNet); and
3) Classification and bounding-box regression, performed by class-specific **Support Vector Machines (SVMs)** and linear regressors.

By replacing hand-crafted descriptors **(HOG, SIFT)** with deep, learned representations, R-CNN achieved a dramatic improvement in detection accuracy on benchmarks such as **PASCAL VOC 2007/2012**, demonstrating the effectiveness of transfer learning from large-scale classification networks. However, this approach was computationally inefficient, as

each proposal required an independent forward pass through the CNN, leading to thousands of redundant computations per image.

Despite its inefficiency, R-CNN established the conceptual foundation of **region-based deep detection** and inspired subsequent improvements—**Fast R-CNN and Faster R-CNN**—that sought to retain its accuracy while eliminating its computational bottlenecks.

2) Fast R-CNN: To overcome the computational inefficiency of **R-CNN, Fast R-CNN** (Girshick, 2015) introduced a unified and more efficient detection framework. Instead of forwarding each region proposal independently through the network, Fast R-CNN employed a **single shared convolutional feature map** for the entire image. From this shared map, a **Region of Interest (RoI)** Pooling layer extracted fixed-size feature representations for each candidate region.

This design enabled joint classification and bounding-box regression within a single end-to-end trainable network, dramatically reducing redundancy and improving both training and inference efficiency. The shared convolutional computation allowed detection to scale better to large datasets such as **PASCAL VOC and MS COCO**, while maintaining high accuracy.

However, Fast R-CNN still depended on **external region proposal methods**, typically **Selective Search**, which required several seconds per image and limited real-time performance. This remaining bottleneck motivated the next major milestone—**Faster R-CNN**, which replaced hand-crafted region proposals with a learnable **Region Proposal Network (RPN)**.

3) Faster R-CNN: **Faster R-CNN** (Ren et al., 2016) represented a major breakthrough in object detection by introducing the **Region Proposal Network (RPN)**, which enabled the generation of region proposals directly within the CNN architecture in an end-to-end trainable manner. The RPN operated on shared convolutional feature maps, using a set of **anchor boxes** at each spatial location to predict objectness scores and bounding-box coordinates.

This innovation eliminated the dependency on external proposal algorithms such as Selective Search, effectively removing the key computational bottleneck of earlier two-stage detectors. As a result, Faster R-CNN achieved significant gains in both accuracy and efficiency, setting a new benchmark on datasets such as **PASCAL VOC, MS COCO, and ImageNet DET**.

The integration of RPN and detection head into a unified framework marked the first fully trainable deep detection pipeline, laying the conceptual groundwork for numerous successors, including **Mask R-CNN , Cascade R-CNN, and Feature Pyramid Networks (FPN)**.

Nevertheless, Faster R-CNN remained computationally demanding, with inference times typically unsuitable for real-time deployment, especially on resource-constrained devices. This trade-off between precision and speed motivated the exploration of lightweight, single-stage alternatives and further architectural refinements.

4) Mask R-CNN: **Mask R-CNN** (He et al., 2017) extended the Faster R-CNN architecture by introducing a parallel mask prediction branch for pixel-level instance segmentation, thereby transforming the detector into a unified framework for multiple vision tasks. This additional branch operated on Region of Interest (RoI) features and predicted binary segmentation masks for each detected object, enabling precise instance-level understanding beyond bounding boxes.

A critical refinement in Mask R-CNN was the introduction of **RoIAlign**, which replaced the quantized pooling operation of RoI Pooling with bilinear interpolation, eliminating misalignment between the feature map and original image coordinates. This improved spatial accuracy was especially beneficial for fine-grained localization tasks such as segmentation and keypoint detection.

Mask R-CNN became the de facto baseline for a wide range of tasks—object detection, instance segmentation, and human pose estimation—and inspired numerous derivatives such as **PANet, Hybrid Task Cascade, and Detectron/Detectron2** frameworks.

However, despite its versatility and high accuracy, Mask R-CNN remained computationally intensive, with high memory and latency costs that limited deployment on edge or real-time systems. Nevertheless, it marked the culmination of the **two-stage detection paradigm**, demonstrating the power of multi-task learning and paving the way for unified perception models that integrate detection and segmentation.

5) Discussion and Comparative Analysis: The evolution of the R-CNN family illustrates the gradual transition from manually designed, multi-stage pipelines to fully integrated, end-to-end trainable detection frameworks. As summarized in Table X, each generation progressively reduced computational redundancy while improving accuracy and architectural elegance.

R-CNN introduced deep feature representations but suffered from high computational cost due to per-region processing. Fast R-CNN alleviated this by sharing convolutional computation across the entire image, while Faster R-CNN achieved full integration by introducing a learnable Region Proposal Network (RPN). Finally, Mask R-CNN unified detection and segmentation within a single multi-task model, extending the scope of object detection toward holistic visual understanding.

Despite their outstanding accuracy, these two-stage detectors remain computationally expensive, making them less suitable for real-time or edge deployment. This limitation motivated the development of one-stage detectors such as YOLO and SSD, which aimed to simplify the detection pipeline and achieve higher inference speed without significant loss of accuracy.

Table II summarizes the main members of the R-CNN family, highlighting their impact, representative metrics, primary design choices, and limitations.

C. One-stage Detectors

In contrast to the region proposal–based paradigm of two-stage detectors, one-stage methods were introduced to perform

TABLE II
COMPARATIVE SUMMARY OF THE R-CNN FAMILY (REPRESENTATIVE METRICS AND DESIGN CHOICES)

Model	Year	Legacy / Impact	Key Limitations	mAP (VOC / COCO)	Speed (FPS)	Training Paradigm	Feature Extraction
R-CNN	2014	First deep-learning–based detector; foundation for region-based methods	Extremely slow; redundant computation	(VOC07) ∼58.5	∼0.02–	Multi-stage (CNN + SVM + bbox regressor)	Per-region CNN (e.g., AlexNet)
Fast R-CNN	2015	End-to-end training; reduced redundancy	Still relies on external proposals	(VOC07) ∼70+	∼0.5–	Single-stage (joint classification & regression)	Single CNN for whole image + RoI pooling
Faster R-CNN	2016	Set new standard; base for most two-stage detectors	Computationally heavy	(VOC07) ∼76; (COCO) higher	∼5–7	Fully end-to-end	Shared convolutional backbone
Mask R-CNN	2017	Unified framework for detection, segmentation, keypoints	High latency & memory cost	(COCO) ∼39 (mask mAP)	∼2–5	Multi-task (detection + segmentation)	Shared backbone + FPN; RoIAlign

object classification and bounding-box regression directly from dense feature maps, effectively reframing detection as a single end-to-end regression task. By eliminating the separate region proposal stage, these approaches achieved remarkable gains in computational efficiency and enabled real-time inference on commodity GPUs.

The one-stage design follows a fully convolutional architecture, where predictions are generated simultaneously across all spatial locations and scales. While this simplification substantially improved speed and deployment efficiency, it often came at the cost of slightly lower localization accuracy compared to two-stage counterparts.

Notable representatives of this family include **YOLO (You Only Look Once), SSD (Single Shot MultiBox Detector), and RetinaNet**, each contributing unique innovations to balance the long-standing trade-off between speed and precision in object detection.

1) YOLO (You Only Look Once): The original **YOLOv1** model (Redmon et al., 2016) represented a paradigm shift in object detection by formulating detection as a single regression problem rather than a multi-stage pipeline. The input image was divided into an S×S grid (typically 7×7), with each grid cell directly predicting bounding-box coordinates, objectness scores, and class probabilities in a single forward pass. This unified formulation enabled YOLO to perform real-time detection, achieving speeds exceeding 45 frames per second on standard GPUs.

Unlike proposal-based methods, YOLO processed the entire image globally, allowing the network to leverage contextual information and achieve remarkable inference speed. However, the coarse grid structure and fixed number of detections per cell limited its ability to accurately localize small or overlapping objects, leading to lower mean Average Precision (mAP) compared to region-based counterparts such as Faster R-CNN.

Despite these limitations, YOLOv1 established the foundation for a long line of one-stage detectors, inspiring subsequent versions (YOLOv2, YOLOv3, YOLOv4, etc.) that progressively enhanced accuracy through improved backbones, anchor-based prediction, and multi-scale feature fusion.

The results summarized in Table III highlight the key contribution of YOLOv1 in redefining object detection as a single-stage regression problem. By replacing multi-stage processing with a unified, end-to-end architecture, YOLO achieved unprecedented real-time performance while maintaining competitive accuracy. However, its grid-based design introduced spatial limitations that hindered precise localization of small or overlapping objects.

TABLE III

Feature	YOLOv1	Two-Stage Detectors
Architecture	Fully convolutional (grid-based)	Proposal + classifier
Detection type	Single regression (end-to-end)	Multi-stage
Speed (FPS)	∼45	>10
Strength	Real-time, simple pipeline	High accuracy
Weakness	Poor small-object performance	Slow inference

2) SSD (Single Shot MultiBox Detector): The **SSD** framework (Liu et al., 2016) sought to bridge the gap between the speed of YOLO and the accuracy of two-stage detectors by leveraging multi-scale feature representations within a single convolutional network. Unlike YOLO's coarse grid structure, SSD performed predictions from multiple feature maps at different resolutions, allowing the detection of objects of varying sizes — shallow layers captured small objects, while deeper layers handled larger ones.

Each spatial location on these feature maps was associated with a set of default anchor boxes of various aspect ratios and scales, enabling the network to regress both bounding-box coordinates and class probabilities in a single forward

pass. This design produced a strong trade-off between accuracy and real-time efficiency, achieving higher precision on small and medium-sized objects compared to YOLOv1 while maintaining competitive inference speed.

Although SSD's performance remained below that of two-stage detectors on challenging datasets such as **MS COCO**, it demonstrated that one-stage detectors could achieve high accuracy without sacrificing speed, establishing a new balance between computational efficiency and detection quality.

3) RetinaNet: RetinaNet [**?**] addressed one of the fundamental challenges in dense one-stage object detection — the extreme class imbalance between a vast number of background (negative) anchors and relatively few positive samples. In conventional training, this imbalance caused the loss function to be dominated by easy negatives, leading to suboptimal convergence and degraded accuracy.

To overcome this, RetinaNet introduced the *Focal Loss*, a dynamically scaled cross-entropy loss defined as $(1 - p_t)^\gamma \log(p_t)$, which down-weights well-classified examples and forces the model to focus on harder, misclassified samples. This simple yet powerful modification enabled effective training of dense detectors without the need for heuristic sampling or hard example mining.

Built upon a **Feature Pyramid Network (FPN)** backbone, RetinaNet achieved detection accuracy comparable to Faster R-CNN while retaining the computational efficiency of one-stage models. This balance between precision and speed established RetinaNet as a widely adopted baseline for single-shot detection and inspired subsequent research into loss reweighting, anchor-free prediction, and balanced training paradigms.

D. Anchor-free Detectors

Anchor-free methods emerged to simplify detection pipelines by removing the need for pre-defined anchor boxes, which required extensive hyperparameter tuning and increased complexity.

1) FCOS: Fully Convolutional One-Stage Object Detector (FCOS) formulated detection as per-pixel regression of bounding box coordinates and classification, eliminating anchors entirely.

2) CornerNet: CornerNet detected objects by predicting paired keypoints (top-left and bottom-right corners), using an embedding mechanism to group corners belonging to the same object.

3) CenterNet: CenterNet further simplified detection by predicting object centers and regressing to object size and offsets, making the pipeline intuitive and efficient. These methods demonstrated that anchor-free designs could achieve accuracy comparable to anchor-based detectors while reducing design complexity.

E. Comparison and Trade-offs

Deep learning-based detectors reveal a fundamental trade-off between speed and accuracy. Two-stage methods such as Faster R-CNN and Mask R-CNN dominate in accuracy but are computationally intensive, limiting deployment in real-time or edge scenarios. One-stage methods like YOLO and SSD prioritize speed, enabling applications in autonomous driving and robotics, but initially sacrificed accuracy. RetinaNet and anchor-free approaches bridged this gap, achieving high accuracy with streamlined architectures. Overall, the evolution of deep learning-based detection reflects a continuous effort to balance accuracy, speed, and generalization, motivating subsequent research into transformers and generative approaches.

IV. YOLO Family in Detail

Among deep learning-based detectors, the You Only Look Once (YOLO) family has been one of the most influential and widely deployed. Unlike traditional region proposal-based approaches, YOLO reframed object detection as a single regression task, enabling real-time detection without sacrificing too much accuracy. Its simplicity, efficiency, and versatility made it the preferred choice in domains requiring real-time perception, including autonomous driving, UAVs, mobile devices, and embedded vision systems. The continuous evolution of YOLO versions, from v1 to v8, demonstrates the progression of architectural innovations toward achieving an optimal balance between speed and accuracy.

A. YOLOv1

YOLOv1 (2016) introduced the concept of end-to-end detection using a single convolutional neural network. The image was divided into a $S \times S$ grid, and each grid cell directly predicted bounding box coordinates, confidence scores, and class probabilities. YOLOv1 achieved unprecedented speed (up to 45 FPS), making real-time detection possible for the first time. However, it struggled with small objects and crowded scenes due to coarse grid-based localization.

B. YOLOv2 (YOLO9000)

YOLOv2 (2017), also known as YOLO9000, addressed key limitations of v1 by introducing anchor boxes, batch normalization, high-resolution classifiers, and a Darknet-19 backbone. It could detect over 9000 object categories when jointly trained on ImageNet and COCO, enabling open-domain detection. YOLOv2 improved accuracy significantly while maintaining real-time inference speeds.

C. YOLOv3

YOLOv3 (2018) incorporated a deeper Darknet-53 backbone with residual connections, improving feature representation. It also introduced multi-scale predictions using feature pyramid networks (FPN), enhancing detection of small objects. YOLOv3 balanced speed and accuracy better than its predecessors and became one of the most widely adopted detection models.

D. YOLOv4

YOLOv4 (2020) emphasized accessibility by enabling high performance on consumer-grade GPUs. It integrated several training and architectural improvements such as CSPDarknet53 backbone, Cross-Stage Partial connections, weighted

residual connections (WRC), and the use of advanced data augmentation techniques (Mosaic, Self-Adversarial Training). YOLOv4 demonstrated state-of-the-art accuracy while preserving real-time capability.

E. YOLOv5

YOLOv5 (2020, by Ultralytics) marked a significant shift, being implemented entirely in PyTorch rather than Darknet. It introduced modular designs, ease of deployment, and automatic mixed precision training. YOLOv5 offered multiple model sizes (s, m, l, x) to balance trade-offs between accuracy and speed, making it practical for a wide range of hardware from edge devices to cloud servers.

F. YOLOv6

YOLOv6 (2022) focused on industrial deployment, particularly for edge and real-time scenarios. It introduced efficient training strategies, improved architectures, and quantization-friendly designs, achieving faster inference with minimal accuracy loss. YOLOv6 was widely adopted in scenarios like robotics and surveillance where resource constraints are critical.

G. YOLOv7

YOLOv7 (2022) further optimized performance through architectural innovations such as Extended Efficient Layer Aggregation Network (E-ELAN) and model re-parameterization. It achieved state-of-the-art accuracy and speed trade-offs on COCO, outperforming both YOLOv4 and YOLOv5 in multiple configurations. YOLOv7 became a benchmark for real-time object detection at scale.

H. YOLOv8

YOLOv8 (2023, by Ultralytics) represents the latest generation of YOLO models. It simplified the architecture, introduced anchor-free detection, and improved generalization across domains. YOLOv8 supports classification, detection, segmentation, and pose estimation in a unified framework, emphasizing practical deployment. It is highly optimized for ONNX, TensorRT, and mobile inference, making it the most versatile YOLO release to date.

I. Practical Deployment on Edge Devices

A major reason for YOLO's widespread popularity is its suitability for deployment on resource-constrained devices. Variants such as YOLOv5s, YOLOv6n, and YOLOv8n are lightweight yet effective, enabling deployment on drones, mobile phones, and IoT cameras. Techniques such as quantization, pruning, and hardware-aware neural architecture design further enhance efficiency, making YOLO central to real-world applications that demand both speed and accuracy.

J. Comparison of YOLO Versions

Table IV provides a summary comparison of YOLO versions in terms of backbone, key innovations, typical inference speed, and benchmark accuracy on COCO.

TABLE IV
COMPARISON OF YOLO VERSIONS (V1–V8)

Version	Backbone	FPS (approx.)	mAP (COCO)
YOLOv1	Custom CNN	45	63 (VOC)
YOLOv2	Darknet-19	40–45	76 (VOC), 21.6 (COCO)
YOLOv3	Darknet-53	30–35	33.0
YOLOv4	CSPDarknet53	30+	43.5
YOLOv5	CSPDarknet (PyTorch)	30–140	37–50
YOLOv6	Efficient-optimized	35–150	43–50
YOLOv7	E-ELAN backbone	30–160	51.4
YOLOv8	Anchor-free, lightweight	30–200+	53+

K. Summary

The YOLO family has transformed object detection by achieving real-time performance without sacrificing accuracy, and its continuous evolution reflects a balance between research innovation and practical deployment needs. From YOLOv1's pioneering real-time detection to YOLOv8's anchor-free, multi-task framework, YOLO remains one of the most impactful and widely adopted object detection families to date.

V. TRANSFORMER-BASED DETECTORS

The introduction of transformers into computer vision has significantly reshaped the landscape of object detection. Unlike convolutional neural networks (CNNs), which primarily capture local receptive fields, transformers rely on the *self-attention mechanism* to model long-range dependencies across an entire image. This ability to capture global context is particularly valuable in object detection, where understanding relationships between objects and their surroundings is critical. Inspired by the success of transformers in natural language processing (e.g., BERT, GPT), researchers adapted these architectures to visual tasks, leading to the development of transformer-based detectors.

A. DETR: End-to-End Detection

DETR (Detection Transformer) was the first major work to apply transformers directly to object detection. It reformulated detection as a direct set prediction problem, eliminating the need for hand-crafted components such as anchor boxes, region proposals, or non-maximum suppression (NMS). DETR consists of a CNN backbone for feature extraction, followed by a transformer encoder-decoder architecture. The encoder captures global relationships among image patches, while the decoder outputs a fixed-size set of object queries representing bounding boxes and class predictions. DETR demonstrated remarkable simplicity and elegance, but suffered from slow convergence and difficulty handling small objects due to its global attention design.

B. Deformable DETR

To address the limitations of DETR, Deformable DETR introduced a multi-scale deformable attention mechanism. Instead of attending to all positions in the feature map, deformable attention focuses on a sparse set of relevant key

points around each query. This significantly reduces computational cost and accelerates convergence, enabling training in fewer epochs. Moreover, by incorporating multi-scale features, Deformable DETR improved performance on small objects, making transformer-based detection more practical for real-world applications.

C. Sparse R-CNN

Sparse R-CNN proposed an alternative paradigm by directly learning a small set of dynamic object proposals, which are iteratively refined through a cascade of transformer-based heads. Unlike dense anchor-based methods, Sparse R-CNN maintains a fixed set of proposals, drastically reducing computational redundancy. This sparse design improved efficiency and demonstrated that detection can be achieved without relying on dense anchors or sliding windows.

D. Conditional DETR

Conditional DETR further improved convergence by conditioning each query on spatial reference points. This spatial prior reduced the burden of learning positional relationships from scratch, making the model more sample-efficient and robust to initialization.

E. Hybrid CNN-Transformer Models

While pure transformer-based detectors demonstrate strong potential, CNN backbones remain highly effective at capturing low-level visual patterns. Hybrid models combine the strengths of CNNs and transformers. For example, Swin Transformer introduces a hierarchical vision transformer with shifted window attention, which balances global context modeling with local inductive biases. Hybrid architectures often achieve better trade-offs between accuracy and efficiency, making them competitive with CNN-based detectors in practice.

F. Vision-Language Detectors

Recent efforts have extended transformer-based detection into the multimodal domain by integrating vision and language. Models such as GLIP, MDETR, and OWL-ViT leverage large-scale vision-language pretraining to enable open-vocabulary detection. Instead of being limited to a fixed set of categories, these models can detect novel classes described by textual prompts. This capability significantly expands the applicability of detection systems, enabling zero-shot and few-shot generalization across domains.

G. Neighborhood Attention and Neural Architecture Search

Neighborhood Attention Transformers (NAT) were proposed as a more efficient alternative to global self-attention, restricting the receptive field to local neighborhoods while preserving transformer flexibility. This reduces quadratic complexity and makes transformer-based detectors more computationally feasible. In parallel, neural architecture search (NAS) has been applied to discover efficient hybrid CNN-transformer designs tailored for detection, enabling models that are both accurate and resource-aware.

H. Comparison: Strengths and Weaknesses

Transformer-based detectors offer several advantages:

- **Simplicity:** End-to-end design without anchors, proposals, or NMS.
- **Global Context:** Self-attention effectively captures long-range dependencies.
- **Flexibility:** Natural extension to multimodal settings (vision+language).
- **Strong Benchmarks:** Competitive or superior accuracy on COCO and other datasets.

However, they also exhibit limitations:

- **Data Requirements:** Transformers typically require large-scale datasets for effective training.
- **Computational Cost:** Global attention mechanisms are expensive in memory and time.
- **Convergence:** Early variants like DETR converge slowly and require long training schedules.
- **Small Object Detection:** Despite improvements, transformer-based models can struggle with fine-grained details.

I. Summary

Overall, transformer-based detectors have introduced a paradigm shift in object detection, replacing complex hand-engineered components with simple, end-to-end architectures. With ongoing innovations such as deformable attention, hybrid CNN-transformers, multimodal integration, and efficient attention mechanisms, transformers are poised to play a central role in the next generation of object detection models.

VI. Joint Detection and Segmentation Models

While object detection provides bounding boxes and class labels, many real-world applications demand more fine-grained understanding of object boundaries and pixel-level masks. This has motivated the development of models that unify detection and segmentation into a single framework, often referred to as *joint detection and segmentation models*. The goal is **unified perception**, where a single model performs both bounding-box detection and instance-level segmentation, providing richer outputs that are essential for applications such as autonomous driving, medical imaging, and video analytics.

A. Mask R-CNN

Mask R-CNN extended Faster R-CNN by adding a parallel branch for pixel-level segmentation. Specifically, it introduced a lightweight Fully Convolutional Network (FCN) head that predicts a binary mask for each Region of Interest (RoI). The use of RoI Align instead of RoI Pooling improved localization accuracy by avoiding quantization errors. Mask R-CNN became a cornerstone in instance segmentation research due to its modularity, accuracy, and ability to extend to related tasks such as keypoint detection. However, its reliance on a two-stage detection pipeline makes it computationally heavy.

B. PANet

Path Aggregation Network (PANet) built upon Mask R-CNN by enhancing feature fusion across multiple levels of the backbone. It introduced bottom-up path augmentation and adaptive feature pooling to better capture low-level details and improve segmentation accuracy, especially for small objects. PANet significantly boosted the performance of instance segmentation benchmarks but at the cost of additional architectural complexity.

C. YOLACT

YOLACT (You Only Look At CoefficienTs) proposed a real-time instance segmentation framework by decoupling mask generation from localization. It introduced a set of prototype masks predicted for the entire image and combined them with per-instance coefficients to generate final instance masks. YOLACT achieved near real-time speeds, making it suitable for practical applications, though with some loss in mask precision compared to Mask R-CNN.

D. SOLO and SOLOv2

Segmenting Objects by LOcations (SOLO) redefined instance segmentation as a per-pixel classification problem. Each pixel was assigned a category and an instance ID based on its spatial location, enabling a fully convolutional and anchor-free approach. SOLOv2 improved upon this by incorporating dynamic convolution kernels, enhancing efficiency and accuracy. These models removed the need for bounding box proposals entirely, simplifying the segmentation pipeline.

E. Transformer-based Models: MaskDETR

MaskDETR extended DETR by directly predicting segmentation masks in addition to bounding boxes and class labels. Leveraging the self-attention mechanism, MaskDETR modeled global context for both detection and segmentation. This unified transformer-based framework demonstrated competitive results on COCO benchmarks and highlighted the potential of end-to-end architectures for joint perception tasks.

F. Comparison: Multi-task vs. Single-task

Joint detection and segmentation models highlight a fundamental trade-off between multi-task and single-task approaches:

- **Multi-task (e.g., Mask R-CNN, PANet):** These models share feature extraction across tasks, enabling high accuracy but often at the expense of increased computational cost and slower inference.
- **Single-task (e.g., YOLO, RetinaNet):** Optimized solely for detection, these models achieve higher speeds but cannot provide fine-grained instance-level information.
- **Unified frameworks (e.g., YOLACT, SOLO, MaskDETR):** These approaches seek a balance by achieving competitive accuracy while supporting both detection and segmentation in real time or near real time.

G. Summary

The evolution of joint detection and segmentation models demonstrates a shift toward unified perception systems that deliver richer outputs for real-world tasks. While traditional two-stage models remain highly accurate, recent one-stage and transformer-based methods illustrate the potential for efficient, scalable, and multi-task perception systems. This direction bridges the gap between object detection and dense segmentation, moving closer to general-purpose vision frameworks.

VII. Generative Approaches

Recent years have witnessed the integration of generative modeling techniques into object detection. Generative Adversarial Networks (GANs) and diffusion-based models, which were originally developed for high-quality image synthesis, have been adapted to enhance detection performance. These methods introduce new perspectives on how generative priors and data augmentation can improve object detection, particularly in challenging scenarios such as small object detection, limited data regimes, and domain adaptation.

A. GAN-based Approaches

Generative Adversarial Networks (GANs) consist of a generator that synthesizes realistic samples and a discriminator that distinguishes between real and fake data. In the context of object detection, GANs have been employed in several ways:

- **Small Object Detection:** GANs can generate super-resolved or context-enhanced representations of small objects, enabling detectors to capture finer details that would otherwise be lost in downsampled feature maps.
- **Data Augmentation:** GAN-based methods generate synthetic training samples for rare or underrepresented object categories, improving class balance and detector robustness.
- **Domain Adaptation:** GANs are also applied to reduce domain gaps, for example, by translating synthetic images (e.g., from simulation environments) into realistic styles that better match real-world datasets.

Although GANs are powerful, training is often unstable, requiring delicate tuning of loss functions and architectures to avoid issues such as mode collapse.

B. Diffusion-based Approaches

Diffusion models have emerged as a new class of generative methods, surpassing GANs in generating diverse and high-quality samples. In object detection, diffusion models have been adapted in innovative ways:

- **DiffusionDet:** Reformulates object detection as a denoising diffusion process, where bounding boxes and categories are progressively refined from noisy inputs. DiffusionDet achieves competitive accuracy while offering a fundamentally new perspective on detection.
- **Pix2Seq-Diffusion:** Extends sequence-based detection models by incorporating diffusion processes, enabling sequential refinement of bounding boxes and class tokens in a probabilistic framework.

- **MaskDiffusion:** Integrates mask prediction with diffusion processes, unifying detection and segmentation in a generative setting. This approach highlights the versatility of diffusion models in handling multiple vision tasks simultaneously.

Diffusion-based methods are generally more stable to train than GANs and can leverage large-scale pretraining for improved performance. However, they often involve high computational costs due to iterative denoising steps.

C. GAN vs. Diffusion: Strengths and Future Directions

Generative approaches for detection highlight complementary strengths:

- **GANs:** Efficient, capable of generating high-resolution samples quickly, and useful for tasks like data augmentation and domain adaptation. However, they suffer from training instability and limited diversity in generated samples.
- **Diffusion Models:** Highly stable training, diverse sample generation, and natural integration into probabilistic frameworks for detection. Yet, they remain computationally intensive, making real-time deployment challenging.

Looking forward, diffusion models are expected to play a more central role in generative detection, especially as research explores accelerating inference through techniques such as distillation and fewer-step sampling. Meanwhile, GANs may continue to serve as lightweight solutions for augmentation and adaptation. Together, these paradigms illustrate a shift toward viewing object detection not only as a discriminative task but also as a generative modeling problem, bridging the gap between perception and synthesis.

VIII. Trends and Extensions

Beyond the mainstream paradigms of two-stage, one-stage, and transformer-based detectors, research in object detection has expanded toward diverse extensions and practical trends. These efforts aim to improve efficiency, adapt to new sensing modalities, and generalize to low-resource or unseen domains. This section reviews several key directions that reflect the evolving landscape of object detection.

A. Lightweight and Mobile Detection

With the proliferation of edge devices such as smartphones, drones, and IoT cameras, there is a growing demand for lightweight detectors capable of real-time inference under strict resource constraints. Approaches include:

- **YOLO-Nano:** A compact variant of YOLO designed for embedded devices, balancing speed and accuracy through neural architecture search.
- **MobileNet-SSD:** Combines the efficient MobileNet backbone with the SSD detection head, achieving real-time inference on mobile hardware.
- **Quantization and Pruning:** Techniques such as weight quantization, pruning, and knowledge distillation further compress models while retaining accuracy.

Lightweight detection has made computer vision accessible to low-power devices, but maintaining high accuracy under tight latency constraints remains a challenge.

B. 3D Object Detection

Traditional 2D detection is insufficient for applications like autonomous driving and robotics, where 3D spatial reasoning is required. 3D object detection incorporates additional modalities such as LiDAR point clouds, stereo vision, or RGB-D sensors:

- **LiDAR Fusion:** Methods like PointRCNN and PV-RCNN fuse LiDAR point clouds with image data for robust 3D detection in autonomous driving.
- **Voxelization and Point-based Networks:** Techniques such as VoxelNet and PointNet++ enable end-to-end 3D reasoning from raw point clouds.

While highly accurate, 3D detectors face scalability challenges due to the computational cost of processing point clouds.

C. Multi-modal Detection

Vision alone can be ambiguous in complex environments. Multi-modal approaches integrate complementary modalities such as language, depth, thermal imaging, or radar:

- **Vision-Language Models:** Frameworks like GLIP and OWL-ViT unify detection with natural language supervision, enabling open-vocabulary and zero-shot detection.
- **RGB-Thermal Detection:** Combining RGB with thermal imaging improves performance in low-light or adverse weather conditions.

These approaches highlight a shift toward general-purpose perception systems capable of reasoning across modalities.

D. Few-shot and Zero-shot Detection

Traditional detectors rely on large labeled datasets, which are expensive to collect. Few-shot and zero-shot detection address this limitation:

- **Few-shot:** Models such as Meta R-CNN leverage meta-learning to rapidly adapt to novel classes with only a few examples.
- **Zero-shot:** Vision-language pretraining enables detection of unseen categories by mapping textual descriptions to visual features, as in CLIP-based models.

These methods expand the applicability of detection to real-world domains with limited or evolving label sets.

E. Domain Adaptation

Detection models trained on one dataset often degrade when applied to different domains (e.g., synthetic to real, day to night). Domain adaptation techniques mitigate this issue by aligning feature distributions:

- **Adversarial Adaptation:** Methods introduce adversarial losses to reduce domain discrepancy.
- **Style Transfer:** GAN-based style translation adapts synthetic training images to realistic domains.

Domain adaptation is especially critical for autonomous driving, where deployment conditions may vary widely.

F. Semi-supervised and Self-supervised Detection

Annotating bounding boxes is expensive and time-consuming. Semi-supervised and self-supervised learning reduce reliance on labeled data:

- **Semi-supervised:** Frameworks such as STAC and Unbiased Teacher use pseudo-labeling to exploit large-scale unlabeled datasets.
- **Self-supervised:** Pretraining approaches such as MoCo, BYOL, and DINO enable models to learn general visual features without supervision, which can then be fine-tuned for detection.

These techniques have shown strong potential in reducing data dependence while maintaining competitive accuracy.

G. Summary

The trends and extensions in object detection reflect a clear trajectory toward efficiency, robustness, and generalization. Lightweight models enable deployment on edge devices, 3D detection expands applicability to autonomous systems, and multimodal integration moves toward general-purpose AI. At the same time, few-shot, zero-shot, and self-supervised approaches address the fundamental challenge of data scarcity. Together, these directions highlight the adaptability of object detection research to both technological advances and practical demands.

IX. Datasets and Benchmarks

The rapid progress in object detection has been driven in large part by the availability of standardized datasets and benchmarks. Large-scale annotated datasets provide training data for supervised learning, while benchmarks establish fair grounds for evaluating and comparing detection models. This section reviews major datasets across different domains and the commonly used evaluation metrics.

A. General-purpose Datasets

1) PASCAL VOC: The PASCAL Visual Object Classes (VOC) challenge, introduced in 2007, was one of the first widely used benchmarks for object detection. It includes 20 object categories with bounding box annotations. VOC established the mean Average Precision (mAP) metric and served as the primary benchmark until larger datasets emerged.

2) MS COCO: The Microsoft Common Objects in Context (MS COCO) dataset is the most widely adopted benchmark for modern detection research. It contains over 330,000 images with more than 80 object categories and dense annotations. COCO includes challenges such as small objects, crowded scenes, and contextual variation. Its evaluation metric, mAP averaged over multiple IoU thresholds (0.5:0.95), provides a more comprehensive measure of detection quality compared to VOC.

3) Open Images: The Open Images dataset is one of the largest publicly available datasets, with millions of annotated images across hundreds of categories. It includes bounding boxes, segmentation masks, and hierarchical label structures. Open Images is valuable for evaluating large-scale detection and long-tail distribution problems.

B. Autonomous Driving Datasets

1) KITTI: The KITTI benchmark provides stereo images, LiDAR point clouds, and annotations for cars, pedestrians, and cyclists. It is one of the earliest datasets for autonomous driving and has been widely used for both 2D and 3D detection.

2) Waymo Open Dataset: The Waymo dataset is a large-scale benchmark with multi-sensor data including LiDAR and camera images. It contains over 1,000 driving segments covering diverse conditions, making it one of the most comprehensive datasets for 3D detection and tracking.

3) nuScenes: The nuScenes dataset includes 1,000 driving scenes with camera, LiDAR, and radar data. It provides annotations for 23 object classes along with tracking information. nuScenes has become a standard benchmark for evaluating multi-modal and 3D object detection models.

C. Domain-specific Datasets

1) DOTA: The Dataset for Object Detection in Aerial Images (DOTA) focuses on detecting objects from aerial and satellite imagery. It includes objects with arbitrary orientations, such as airplanes, ships, and buildings, making it a challenging benchmark for rotated bounding box detection.

2) xView: The xView dataset is another large-scale dataset for remote sensing applications, containing millions of objects annotated across hundreds of categories. It is particularly challenging due to small object sizes and extreme scale variations.

3) Medical Datasets: Medical object detection datasets target specific applications such as tumor detection, polyp detection, or cell nucleus localization. Examples include DeepLesion (CT scans with lesion annotations) and polyp detection datasets in endoscopy. These datasets are smaller in scale but critical for domain-specific applications.

D. Evaluation Metrics

1) Intersection over Union (IoU): IoU measures the overlap between the predicted bounding box and the ground truth, defined as the ratio of intersection area to union area. A detection is considered correct if its IoU exceeds a threshold (e.g., 0.5).

2) Average Precision (AP): AP summarizes the precision-recall curve by integrating precision across all recall levels for a given IoU threshold. VOC traditionally reported AP at IoU=0.5.

3) Mean Average Precision (mAP): mAP is the mean of AP scores across all categories. The COCO benchmark extended this by averaging AP across multiple IoU thresholds (0.5 to 0.95 in steps of 0.05), providing a more rigorous evaluation.

4) Precision-Recall (PR) Curve: The PR curve plots precision against recall, illustrating the trade-off between false positives and false negatives. It provides deeper insights into the performance of a detector beyond a single numerical score.

E. Summary

Datasets and benchmarks have played a crucial role in shaping object detection research. From VOC and COCO to large-scale Open Images and specialized datasets like KITTI, nuScenes, and DOTA, each benchmark has introduced unique challenges. Evaluation metrics such as IoU, AP, and mAP provide standardized ways to measure progress. However, real-world deployment often requires robustness beyond benchmark conditions, motivating research into domain adaptation, long-tail recognition, and open-world detection.

X. Challenges and Open Problems

Despite remarkable progress in object detection, several challenges remain unresolved, preventing current methods from achieving robust and universal performance. These challenges span technical limitations, data constraints, and broader societal considerations.

A. Small Object Detection

Detecting small objects remains one of the most persistent challenges. Small objects often occupy only a few pixels, making it difficult for convolutional and transformer-based models to capture sufficient semantic and spatial information. Downsampling operations in CNN backbones further exacerbate this issue by discarding fine-grained details. While feature pyramids and super-resolution techniques have been introduced, small object detection continues to lag significantly behind performance on medium and large objects.

B. Occlusion and Clutter

Objects in real-world scenes are often partially occluded by other objects or appear in cluttered environments. Traditional detectors may confuse overlapping instances or miss heavily occluded targets altogether. Techniques such as part-based modeling, attention mechanisms, and relational reasoning have improved robustness, but reliable detection under heavy occlusion remains unsolved. This is especially critical for safety-sensitive domains such as autonomous driving.

C. Edge Deployment and Efficiency

Although real-time object detection has become feasible, deploying models on edge devices with limited computational and memory resources remains challenging. Drones, mobile phones, and IoT sensors require efficient detectors that operate under strict latency and energy constraints. Lightweight backbones, model compression, quantization, and pruning have made progress, yet achieving the balance between high accuracy and low-power deployment continues to be an open research problem.

D. Domain Shift and Generalization

Object detectors trained on benchmark datasets often struggle when deployed in environments with different lighting, weather, or geographic conditions. For example, a detector trained on daytime urban scenes may fail under nighttime or adverse weather conditions. Domain adaptation and domain generalization methods aim to mitigate this problem, but achieving consistent performance across highly diverse real-world settings remains difficult. The broader challenge is to build detectors that generalize beyond closed-set benchmarks into open-world detection.

E. Data Annotation Cost

High-quality annotated datasets are crucial for training object detectors, but manual annotation of bounding boxes and segmentation masks is time-consuming and expensive. For certain domains such as medical imaging or remote sensing, annotation requires domain expertise, further increasing cost. Semi-supervised, self-supervised, and weakly supervised detection frameworks attempt to reduce reliance on labeled data, but achieving performance comparable to fully supervised approaches is still a major challenge.

F. Ethical Considerations: Bias and Privacy

As object detection systems are increasingly deployed in sensitive domains such as surveillance, healthcare, and autonomous driving, ethical concerns have gained importance. Training datasets often exhibit biases related to geography, demographics, or context, leading to unfair or inaccurate predictions for underrepresented groups. Additionally, large-scale deployment in surveillance raises concerns about privacy and civil liberties. Addressing these ethical issues requires not only technical solutions, such as fairness-aware training and privacy-preserving learning, but also careful regulatory and societal considerations.

G. Summary

In summary, current object detection methods face unresolved challenges related to accuracy, efficiency, robustness, and ethics. Progress on small object detection, occlusion handling, efficient edge deployment, domain generalization, and low-cost annotation remains essential for widespread adoption. At the same time, ethical considerations must be addressed to ensure that object detection technologies are deployed responsibly and fairly. These open problems present significant opportunities for future research in the field.

XI. Future Directions

Object detection research has advanced significantly, yet numerous opportunities exist for further exploration. Several emerging paradigms and technologies are likely to shape the next generation of detection models.

A. Foundation Models and Multimodal Learning

Recent breakthroughs in vision-language foundation models such as CLIP, ALIGN, and Florence, as well as segmentation-oriented models like SAM (Segment Anything Model), demonstrate the power of pretraining on massive multimodal datasets. Extending these models to detection offers the potential for open-vocabulary and zero-shot capabilities, enabling detectors to recognize categories unseen during training. Future work will likely focus on integrating detection into multimodal AI systems that combine vision, language, and possibly other modalities such as audio and 3D sensing.

B. Joint Detection and Segmentation

The line between detection, instance segmentation, and panoptic segmentation is increasingly blurred. Unified frameworks such as Mask R-CNN, SOLOv2, and Mask-DETR suggest that future research may move toward holistic perception models capable of object localization, recognition, and segmentation in a single pipeline. Such joint models improve efficiency by sharing representations and may provide richer scene understanding for applications such as autonomous driving and medical imaging.

C. Continual and Lifelong Learning

Traditional detectors assume access to static datasets, yet real-world applications require adaptation to evolving environments. Continual learning aims to enable detectors to acquire new object categories or adapt to new domains without catastrophic forgetting. Achieving robust lifelong object detection will require advances in memory-efficient adaptation, replay mechanisms, and architectures that balance plasticity with stability.

D. Self-Supervised and Weakly Supervised Detection

Self-supervised learning has revolutionized representation learning, and its integration into detection is a promising direction. Methods that leverage large amounts of unlabeled images to pretrain backbones (e.g., MoCo, SimCLR, MAE) can reduce the dependency on costly bounding-box annotations. Similarly, weakly supervised and semi-supervised detection approaches will play a central role in democratizing object detection for domains where annotations are scarce or expensive.

E. Diffusion-Based Paradigms

Diffusion models have shown remarkable success in generative modeling, and recent works such as DiffusionDet and Pix2Seq-Diffusion suggest that they can also serve as detection architectures. Unlike traditional discriminative frameworks, diffusion-based detectors formulate detection as a generative process, offering new opportunities for flexible modeling and improved robustness. Future exploration may include hybrid diffusion-transformer architectures and applications in open-world or few-shot detection.

F. Hardware-Aware Neural Architecture Search

Deployment of object detection systems in resource-constrained environments will increasingly require co-design of algorithms and hardware. Hardware-aware neural architecture search (NAS) seeks to automatically discover detection architectures that optimize accuracy, latency, and energy consumption for specific platforms (e.g., edge GPUs, FPGAs, mobile SoCs). This approach could enable broader adoption of detection models in embedded and real-time applications.

G. Summary

Future research directions in object detection are guided by the convergence of multimodal foundation models, unified perception tasks, continual adaptation, and self-supervised learning. The rise of generative paradigms such as diffusion and the integration of hardware-aware optimization promise to further enhance both accuracy and efficiency. Together, these advancements will shape the development of more general, robust, and scalable object detection systems capable of addressing complex real-world challenges.

XII. Conclusion

Object detection has undergone a remarkable transformation over the past two decades. From the early days of sliding-window classifiers and hand-crafted features such as Haar cascades, HOG, and DPM, the field shifted toward deep learning approaches that leveraged CNNs to learn features directly from data. Two-stage detectors such as Faster R-CNN pushed accuracy to new levels, while one-stage models like YOLO and SSD brought real-time detection into practical applications. More recently, transformer-based models such as DETR have introduced a paradigm shift, relying on global self-attention and end-to-end training without anchors or non-maximum suppression.

This survey has highlighted several key milestones across these generations of object detection methods. We emphasized the importance of the YOLO family for real-time detection, the growing impact of transformer-based architectures, and the emerging role of generative paradigms such as GANs and diffusion models. We also discussed extensions to multi-modal and multi-task frameworks, as well as the influence of large-scale datasets and evaluation benchmarks in driving progress.

Despite these advances, significant gaps remain. Small object detection, robustness under occlusion and clutter, domain generalization, and efficient deployment on edge devices are still open problems. Furthermore, the reliance on large-scale annotated datasets continues to pose challenges in terms of cost, scalability, and fairness. Ethical issues such as bias and privacy further highlight the need for responsible deployment of detection systems.

Looking ahead, the future of object detection is likely to be shaped by foundation models, multimodal integration, self-supervised learning, and generative approaches such as diffusion-based detection. Continual learning and hardware-aware optimization will also be critical for real-world adaptability. By bridging accuracy, efficiency, and fairness, the next generation of object detectors will move closer to becoming general-purpose perception systems, playing a foundational role in AI applications across robotics, healthcare, autonomous driving, and beyond.

In conclusion, object detection remains one of the most dynamic areas of computer vision. Its historical progress demonstrates how innovations in algorithms, architectures, and datasets can drive rapid advancements. At the same time, the open challenges point to exciting opportunities for future research. This survey has aimed to provide a comprehensive

overview of the field, guiding both researchers and practitioners toward the next frontiers of object detection.

REFERENCES

Please number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use "Ref. [3]" or "reference [3]" except at the beginning of a sentence: "Reference [3] was the first ..."

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors' names; do not use "et al.". Papers that have not been published, even if they have been submitted for publication, should be cited as "unpublished" [4]. Papers that have been accepted for publication should be cited as "in press" [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.