Computer Modeling in
Engineering & Sciences

Tech Science Press

Doi:10.32604/cmes.2025.064283

REVIEW

Bridging 2D and 3D Object Detection: Advances in Occlusion Handling through Depth Estimation

Zainab Ouardirhi[1,2,*], **Mostapha Zbakh**[2] **and Sidi Ahmed Mahmoudi**[1]

[1]Computer and Management Engineering Department, UMONS Faculty of Engineering, Mons, 7000, Belgium
[2]Communication Networks Department, Ecole Nationale Supérieure d'Informatique and Systems Analysis, Mohammed V University in Rabat, Rabat, 10000, Morocco

*Corresponding Author: Zainab Ouardirhi. Email: zainab.ouardirhi@umons.ac.be

Received: 11 February 2025; Accepted: 16 May 2025; Published: 30 June 2025

ABSTRACT: Object detection in occluded environments remains a core challenge in computer vision (CV), especially in domains such as autonomous driving and robotics. While Convolutional Neural Network (CNN)-based two-dimensional (2D) and three-dimensional (3D) object detection methods have made significant progress, they often fall short under severe occlusion due to depth ambiguities in 2D imagery and the high cost and deployment limitations of 3D sensors such as Light Detection and Ranging (LiDAR). This paper presents a comparative review of recent 2D and 3D detection models, focusing on their occlusion-handling capabilities and the impact of sensor modalities such as stereo vision, Time-of-Flight (ToF) cameras, and LiDAR. In this context, we introduce FuDensityNet, our multimodal occlusion-aware detection framework that combines Red-Green-Blue (RGB) images and LiDAR data to enhance detection performance. As a forward-looking direction, we propose a monocular depth-estimation extension to FuDensityNet, aimed at replacing expensive 3D sensors with a more scalable CNN-based pipeline. Although this enhancement is not experimentally evaluated in this manuscript, we describe its conceptual design and potential for future implementation.

KEYWORDS: Object detection; occlusion handling; multimodal fusion; monocular; 3D sensors; depth estimation

1 Introduction

Object detection remains a fundamental problem in computer vision, playing a crucial role in applications such as autonomous driving, surveillance, and robotics. Although a lot of work has been put toward improving deep learning-based detection models, occlusion is still one of the key issues that affect detection performance in real-world environments eminently [1]. Occlusion occurs when objects partially or fully obstruct each other, leading to loss of visual information and creating ambiguities in object localization and classification. The ability to track and detect occluded objects is profoundly important in urban traffic monitoring, where pedestrians can be occluded by vehicles, or in warehouse automation, where objects overlap in cluttered surroundings, for which failing to do so puts productivity and safety at stake [2]. This challenge calls for the need for robust occlusion-aware detection techniques that can effectively infer missing object structures while maintaining high detection performance across various environmental conditions.

Conventional 2D object detection models, predominantly based on convolutional neural networks (CNNs), have demonstrated remarkable success in detecting objects within images [3]. Frameworks such as Region-based Convolutional Neural Network (R-CNN) and You Only Look Once (YOLO) leverage hierarchical feature extraction and attention mechanisms to enhance object localization and classification accuracy [2]. However, 2D models inherently suffer from depth ambiguity, as they rely solely on appearance cues without direct spatial information. When objects are partially occluded, models can utilize context reasoning and feature aggregation to make informed predictions, but under severe occlusions, the lack of depth perception leads to detection failures [2]. Other 2D detection models, such as transformer-based approaches like Swin Transformer [4] and Detection Transformer (DETR) [5], have introduced global self-attention mechanisms to enhance feature learning; however, they still rely heavily on learned priors rather than explicit depth cues [6]. Consequently, 2D object detection methods remain limited in their ability to fully address severe occlusions, as they lack the spatial depth necessary to disambiguate complex scenes.

To address these limitations, 3D object detection methods leverage additional depth information from sensors such as LiDAR, stereo cameras, and structured light systems [7]. One of the major advantages of including 3D spatial cues is the improved ability to differentiate between background and foreground objects, thus enhancing detection accuracy in occluded scenes [2]. Methods such as PointPillars and Point-Voxel Region-based Convolutional Neural Network (PV-RCNN) have demonstrated superior performance compared to 2D-based approaches, particularly in autonomous driving scenarios where occlusions are frequent and highly dynamic [8]. Despite their effectiveness, 3D object detection systems present critical drawbacks, including high hardware costs, increased computational complexity, and environmental sensitivity [2]. LiDAR sensors, while highly accurate, remain prohibitively expensive and struggle in adverse weather conditions such as rain or fog. Alternative depth-sensing technologies offer alternative depth estimation strategies but still require extensive calibration and can be prone to measurement inconsistencies [7]. This raises an important question regarding when 3D information becomes essential for object detection and whether a more cost-effective solution can bridge the gap between 2D and 3D detection.

An emerging approach in occlusion handling is monocular depth estimation, which seeks to generate 3D representations from 2D images using deep learning techniques [2]. By predicting depth maps directly from monocular images, recent methods such as Mixed Depth and Scale (MiDaS) [9] and Adaptive Binning Network (AdaBins) [10] have demonstrated promising results in reconstructing scene geometry without the need for explicit depth sensors [7]. The potential of learning-based depth estimation to replace LiDAR or stereo sensors offers a scalable and low-cost alternative for applications where high-resolution depth perception is necessary but unavailable due to hardware constraints [2]. However, challenges such as depth estimation accuracy, generalization across diverse scenes, and computational overhead remain significant limitations [11]. Evaluating whether synthetic depth cues can sufficiently replace sensor-based depth information remains a key research challenge in occlusion-aware object detection.

While numerous studies have reviewed advancements in 2D and 3D object detection, the unique focus of this paper lies in evaluating their effectiveness in occlusion handling across different levels of occlusion severity (Easy, Moderate, Hard). This study systematically examines: (i) the capabilities and limitations of 2D object detection models in occluded environments, (ii) the necessity and impact of integrating 3D data for improving occlusion robustness, and (iii) the feasibility of replacing traditional 3D sensors with depth estimation techniques. Furthermore, this review introduces the perspective of integrating learned depth estimation into FuDensityNet, our existing multimodal occlusion-handling framework. By bridging the gap between sensor-based and learning-based depth perception, this work aims to establish new directions for scalable and adaptable occlusion-aware object detection.

Compared to prior reviews, this paper offers several key contributions:

- A comparative benchmark evaluation of state-of-the-art 2D and 3D object detection models under varying occlusion levels, providing empirical insights into their robustness.
- A quantitative analysis of occlusion-induced performance degradation, highlighting the extent to which occlusion affects detection accuracy across different architectures.
- A comparative study assessing the performance trade-offs within 2D object detection models and within 3D object detection models, following a structured state-of-the-art evaluation.
- The introduction of an updated version of FuDensityNet [12], which integrates depth estimation to provide an alternative approach to occlusion handling while reducing reliance on traditional 3D sensors.

These contributions provide a new perspective on occlusion-aware detection, addressing gaps in existing literature by evaluating both detection robustness and the feasibility of scalable depth-based solutions.

The remainder of this paper is structured as follows. Section 2 reviews the various sensor technologies used in object detection, outlining their advantages and limitations in occluded scenarios. Section 3 presents an in-depth analysis of 2D and 3D object detection methods, detailing their architectural advancements and performance constraints. Section 4 discusses comparative evaluations conducted on both standard and occlusion-aware datasets, highlighting key performance metrics. In Section 5, we introduce the updated FuDensityNet framework, demonstrating how learned depth estimation can serve as a viable alternative to traditional 3D sensors for occlusion handling. Finally, Section 6 summarizes key findings and outlines future research directions for improving occlusion-aware detection.

2 Sensor Technologies for Object Detection

Object detection relies on a diverse set of sensor technologies, each offering distinct advantages and limitations in perceiving the environment. Visual sensors, such as RGB cameras, provide high-resolution *2D* imagery, making them the most widely adopted modality due to their affordability and ease of deployment. However, these sensors inherently suffer from depth ambiguity, limiting their effectiveness in occlusion-heavy environments where object separation and spatial reasoning are critical.

To address these challenges, depth-sensing technologies such as LiDAR, stereo vision, and Time-of-Flight (ToF) cameras enable explicit spatial measurements, improving object localization and robustness in occlusion-aware detection. These sensor modalities collectively define the data acquisition pipeline for modern vision systems, providing essential geometric and appearance-based cues necessary for object detection across various applications.

Fig. 1 illustrates the classification of sensor technologies based on their data acquisition methods and functional roles in object detection. The section first examines visual sensing technologies, focusing on RGB and monochrome cameras, which are widely used for extracting appearance-based features. It then explores depth-sensing modalities, which introduce spatial awareness to improve detection performance under occlusions. By analyzing their capabilities, limitations, and adaptability to environmental constraints, this section provides a structured comparison of these technologies in occlusion-aware object detection.

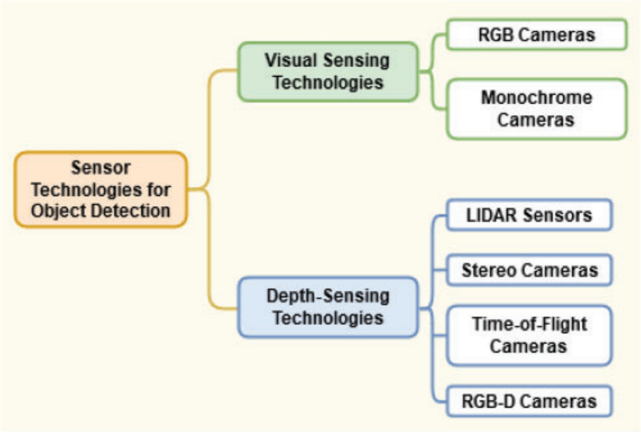

Sensor Technologies for Object Detection
Visual Sensing Technologies
RGB Cameras
Monochrome Cameras
Depth-Sensing Technologies
LIDAR Sensors
Stereo Cameras
Time-of-Flight Cameras
RGB-D Cameras

Figure 1: Taxonomy of sensor technologies for object detection, categorizing visual and depth-sensing modalities

2.1 Visual Sensing Technologies

Visual sensing technologies play a crucial role in object detection, surveillance, and pattern recognition tasks [13]. These sensors capture high-resolution 2D images that provide essential visual cues for deep learning-based detection models. Their affordability and ease of deployment make them widely used across industries, from autonomous driving to industrial automation [14]. However, their primary limitation lies in the lack of depth perception, making them susceptible to occlusion-related errors.

2.1.1 RGB Cameras

RGB cameras are widely used in computer vision, capturing scene information through three color channels: red, green, and blue. Equipped with CMOS or CCD sensors, they produce high-resolution images that serve as the primary input for deep learning-based object detection models [15]. Their affordability and ease of integration make them essential for applications such as autonomous driving, surveillance, and industrial inspection [13].

However, RGB cameras lack depth perception, making occlusion handling a significant challenge. Without explicit spatial information, distinguishing foreground objects from the background in cluttered scenes becomes difficult, leading to detection failures under severe occlusions [14]. Environmental factors such as low lighting, fog, and adverse weather further impact their reliability [15]. To mitigate these limitations, alternative sensing technologies such as stereo cameras or depth estimation techniques are explored to enhance detection in occluded environments.

2.1.2 Monochrome Cameras

Monochrome cameras [16], also known as grayscale cameras, capture images using intensity values alone, offering superior light sensitivity and contrast compared to RGB cameras. By eliminating the color filter array, they maximize photon efficiency, making them well-suited for industrial inspections, biomedical imaging, and low-light environments. Their enhanced structural detail improves edge detection and object recognition, particularly in challenging lighting conditions.

Despite these advantages, monochrome cameras lack depth perception, making them insufficient for occlusion handling in complex scenes. Without color information, distinguishing overlapping objects depends solely on texture and intensity variations, which are unreliable in cluttered environments. This limitation restricts their standalone effectiveness in occlusion-aware object detection [17]. However, when integrated with structured light or stereo vision, monochrome cameras contribute valuable contrast and detail, improving spatial feature extraction. Fig. 2 illustrates an example of grayscale data captured by a monochrome camera.

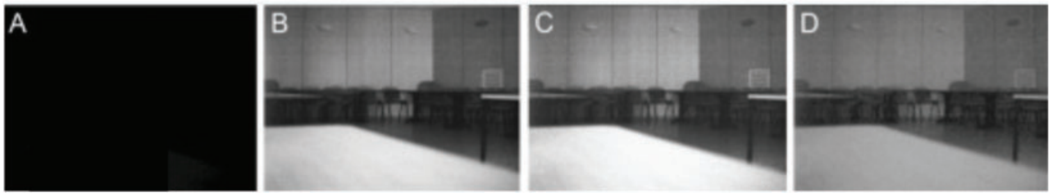


Figure 2: Example of grayscale image captured by a monochrome camera, demonstrating enhanced contrast and edge clarity [18]. **(A)** Raw infrared capture. **(B)** Enhanced grayscale image. **(C)** Denoised version. **(D)** Final preprocessed input

2.2 Depth-Sensing Technologies

Depth-sensing technologies provide explicit spatial information, enabling accurate distance measurement between objects in a scene. Unlike visual sensors that rely on color and texture cues, depth sensors generate spatial representations that significantly improve occlusion handling in complex environments [19]. These sensors play a crucial role in robotics, autonomous navigation, and augmented reality applications by providing robust 3D object localization.

This section examines the major depth-sensing modalities, including LiDAR and alternative technologies designed to estimate depth through different sensing principles. Each sensor type is evaluated based on its ability to enhance object detection under occlusion conditions, as well as its limitations in terms of cost, environmental sensitivity, and computational complexity.

2.2.1 LiDAR Sensors

LiDAR (Light Detection and Ranging) is a depth-sensing technology that provides precise 3D spatial information by emitting laser pulses and measuring their return time. This enables accurate object localization and segmentation, making LiDAR essential in applications such as autonomous driving, urban mapping, and surveillance [19]. Unlike 2D sensors, LiDAR captures depth discontinuities and geometric structures, improving occlusion handling by distinguishing objects in cluttered environments. Fig. 3 illustrates an example of a LiDAR-generated point cloud.

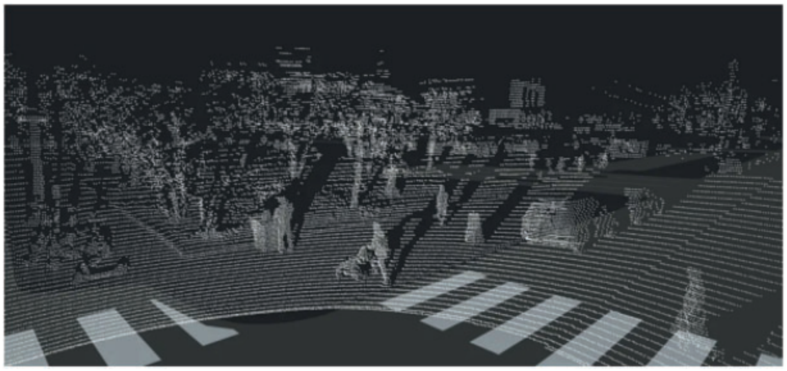


Figure 3: Example of LiDAR point cloud illustrating depth perception capabilities [20]

Despite its high accuracy and ability to operate in low-light conditions, LiDAR faces significant limitations. Its high cost restricts large-scale deployment, while its dependence on precise calibration complicates integration [21]. Furthermore, LiDAR performance degrades under adverse weather conditions such as rain, fog, and dust, affecting laser reflections and reducing depth reliability [22]. These constraints have driven interest in alternative depth-sensing approaches, such as monocular depth estimation, which aims to provide spatial perception without relying on expensive 3D sensors.

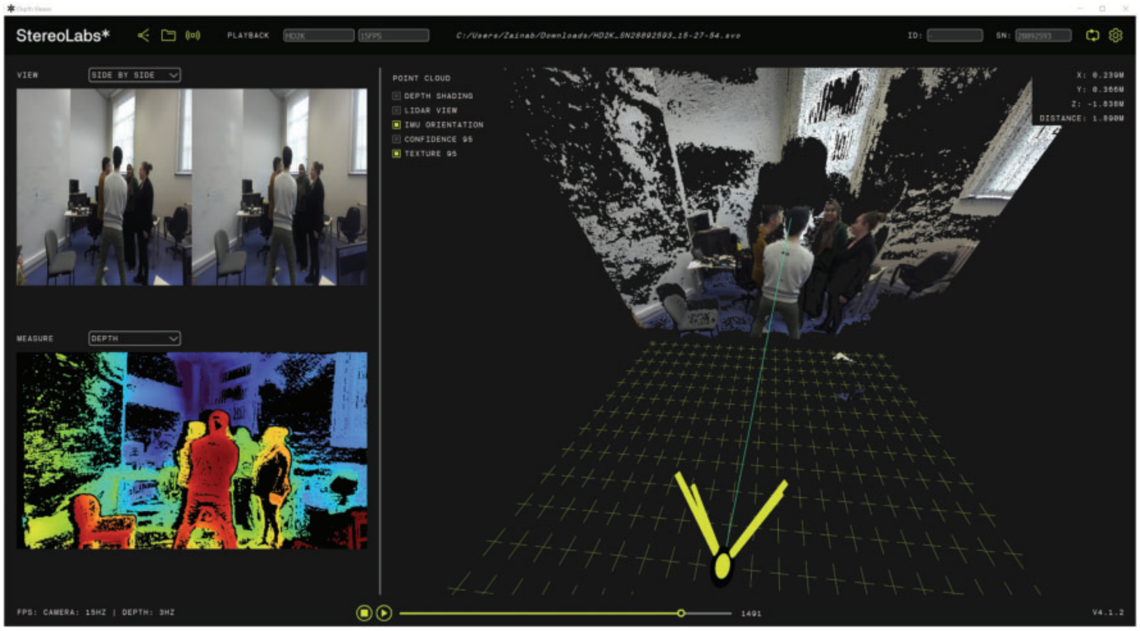
2.2.2 Stereo Cameras

Stereo cameras estimate depth by capturing two offset images from slightly different perspectives, mimicking human binocular vision. By computing disparities between corresponding pixels in the left and right images, they generate dense depth maps without requiring active depth sensing [23]. This makes them a cost-effective alternative to LiDAR, widely used in robotics, surveillance, and augmented reality applications. The depth Z of an object is determined through triangulation (Eq. (1)):

$$Z = \frac{f \cdot B}{d} \tag{1}$$

where f is the focal length, B is the baseline distance between the two lenses, and d is the disparity between matched pixels in the two images [24]. The ZED 2 stereo camera exemplifies a high-resolution depth estimation system, often integrated with IMUs for motion stabilization.

While stereo cameras provide real-time depth perception at a lower cost than LiDAR, they face limitations in environments with low-texture surfaces or extreme lighting, which can disrupt disparity calculations [25]. Additionally, occlusions between objects can create disparity mismatches, affecting depth accuracy. Fig. 4 illustrates a depth map captured by the ZED 2, highlighting its ability to generate structured depth data for object detection and navigation [26].



PLAYBACK HD2K 15FPS C:/Users/Zainab/Downloads/HD2K_SN28892593_15-27-54.svo

ID: - SN: 28892593

VIEW SIDE BY SIDE

POINT CLOUD

□ DEPTH SHADING

□ LIDAR VIEW

■ IMU ORIENTATION

□ CONFIDENCE 95

■ TEXTURE 95

X: 0.239M

Y: 0.366M

Z: -1.838M

DISTANCE: 1.890M

MEASURE DEPTH

FPS: CAMERA: 15HZ | DEPTH: 3HZ

1491

V4.1.2

Figure 4: Depth map visualization: RGB image (top-left), depth map (bottom-left), and point cloud (right) captured by the ZED 2

2.2.3 Time-of-Flight Cameras

Time-of-Flight (ToF) cameras provide real-time depth perception by emitting modulated infrared light and measuring the time taken for the reflected signal to return. This enables fast and accurate depth estimation, making ToF sensors valuable in robotics, augmented reality, and security systems [27]. Unlike passive stereo cameras, ToF sensors do not rely on scene texture, allowing them to function effectively in low-light environments. The depth$Z$ is computed using the Eq. (2):

$$Z = \frac{c \cdot \Delta t}{2} \tag{2}$$

where c is the speed of light, and Δt is the measured time delay [28]. Modern ToF cameras, such as the Microsoft Azure Kinect and PMD Technologies' sensors, leverage this principle to generate dense depth maps, enhancing object detection in occlusion-heavy scenes [29].

While ToF cameras offer fast and precise depth sensing, they are susceptible to errors caused by reflective surfaces, multi-path interference, and ambient light distortions [30]. Additionally, adverse weather conditions like fog and rain can degrade ToF depth accuracy, posing challenges for outdoor deployment. Fig. 5 illustrates a depth map captured by a ToF camera, highlighting its ability to provide structured depth information in various environments [29].

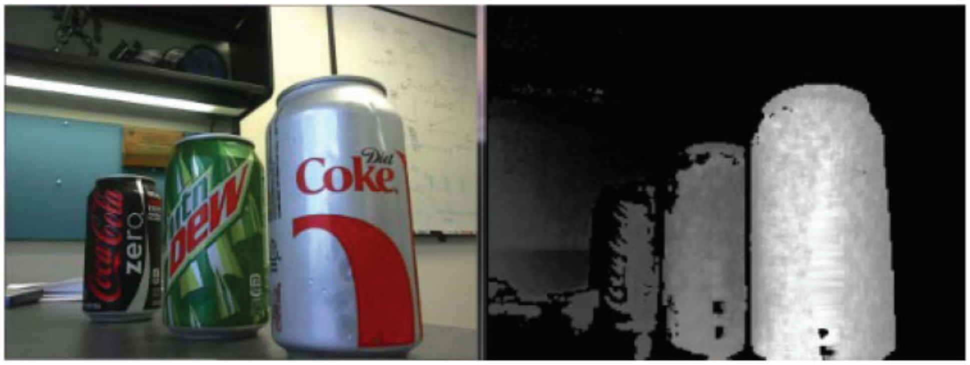


Figure 5: Depth map representation captured by a ToF camera [28]

2.2.4 RGB-D Cameras

RGB-D (Red-Green-Blue and Depth) cameras integrate traditional RGB imaging with depth sensing, providing both visual and spatial information for improved scene understanding. By capturing pixel-wise depth alongside color data, these sensors enhance object detection in occlusion-heavy environments and enable tasks such as 3D modeling, motion tracking, and real-time navigation [31]. Microsoft's Kinect first popularized RGB-D technology, which is now widely adopted in robotics, augmented reality, and autonomous systems.

These cameras combine an RGB sensor with depth estimation techniques such as structured light or ToF [31]. The Microsoft Kinect, for example, projects infrared (IR) patterns onto the scene and measures their deformation to infer depth. This fusion of RGB and depth data results in enriched datasets, allowing for accurate object localization (Fig. 6) [32]. While RGB-D cameras improve depth perception and remain cost-effective compared to LiDAR, they are constrained by shorter depth ranges and lower resolution, limiting their applicability in large-scale environments. Additionally, their depth accuracy is affected by lighting variations and adverse weather, requiring application-specific adaptation [32].

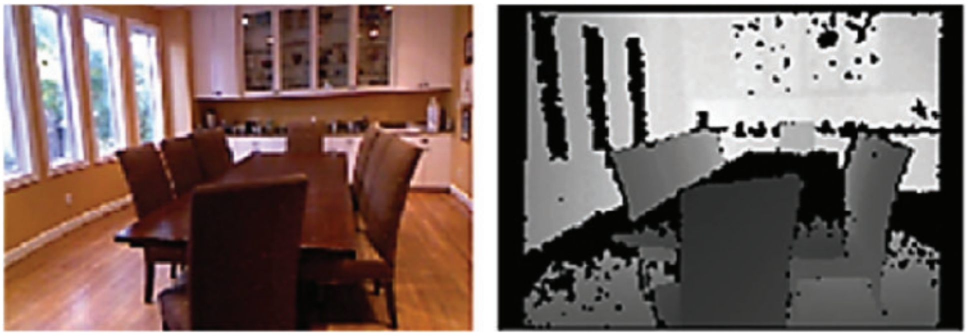


Figure 6: Example of data captured by an RGB-D camera: RGB image (left) and depth map (right) [32]

2.3 *Applications of Sensor Technologies*

Sensor technologies are integral to object detection, providing crucial visual and spatial data for various real-world applications. This section summarizes the key applications of 2D and 3D sensing technologies, emphasizing their relevance to occlusion handling.

2.3.1 Applications of 2D Sensors

RGB and monochrome cameras are widely used in applications requiring high-resolution imaging but lacking explicit depth perception. They support video surveillance by enabling real-time monitoring and anomaly detection, with monochrome cameras enhancing visibility in low-light conditions [13]. In industrial inspection, monochrome imaging improves defect detection, while RGB cameras play a key role in facial recognition and biometric authentication [18]. Despite their advantages, 2D sensors struggle with occlusions due to their inability to capture depth information.

2.3.2 Applications of 3D Sensors

Depth-sensing technologies such as LiDAR, stereo cameras, and ToF cameras provide essential spatial information, making them indispensable for occlusion-aware perception. In autonomous driving, they enhance obstacle detection and lane navigation by generating real-time depth maps [31]. Robotics and industrial automation leverage stereo and ToF cameras for precise object manipulation and collision avoidance [19]. Additionally, LiDAR enables large-scale 3D reconstructions for urban mapping and infrastructure modeling [22]. While these sensors significantly improve occlusion handling, their effectiveness depends on how object detection models process and integrate depth information.

The next section examines state-of-the-art 2D and 3D object detection models, evaluating their ability to leverage sensor data for robust detection in occluded environments.

3 State-of-the-Art Object Detection Methods

Object detection is a key area of computer vision, focused on identifying and localizing objects in both 2D and 3D domains. This section explores state-of-the-art advancements, emphasizing the evolution of architectures and techniques (Fig. 7). By addressing both 2D pixel-based data and 3D spatial representations, it highlights the unique challenges and innovations shaping object detection across diverse environments.

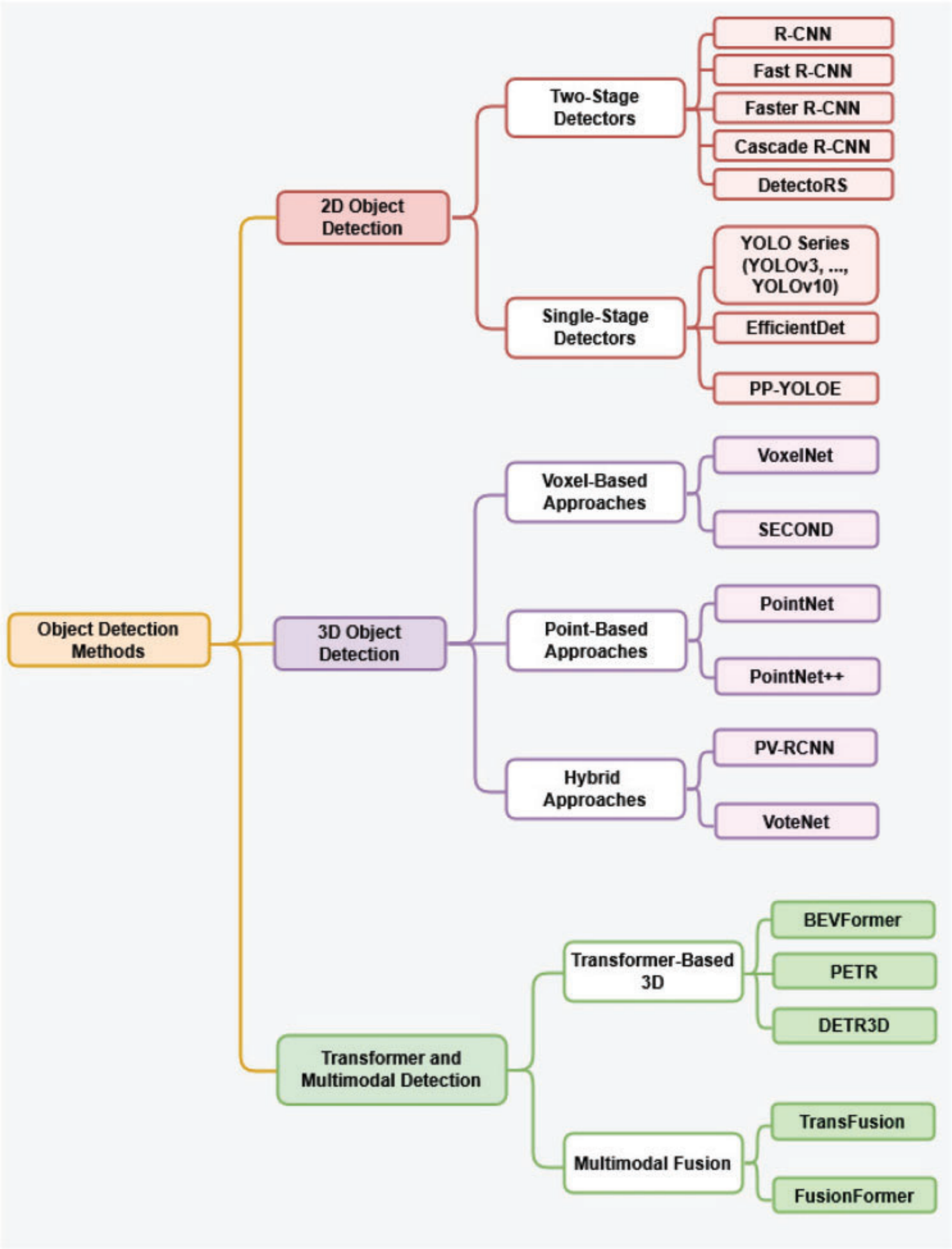


Figure 7: Taxonomy of object detection models, extended beyond 2D and 3D paradigms to include Transformer-based and Multimodal Fusion approaches

3.1 2D Object Detection

Deep learning has significantly advanced 2D object detection, improving both accuracy and efficiency. This section reviews state-of-the-art approaches, emphasizing CNNs as foundational architectures. Two-stage detection frameworks, known for high precision via region-based proposals, are contrasted with one-stage methods, which prioritize real-time performance. These advancements enable applications in video surveillance, urban monitoring, and autonomous systems, where fast and accurate object recognition is critical.

3.1.1 Two-Stage Detectors (Region-Based CNNs)

Two-stage detectors divide object detection into two sequential steps: region proposal generation and classification. The first stage identifies regions of interest (RoIs), while the second stage refines their bounding boxes and classifies objects (Fig. 8). This structure enables high precision, particularly in complex or crowded scenes [33].

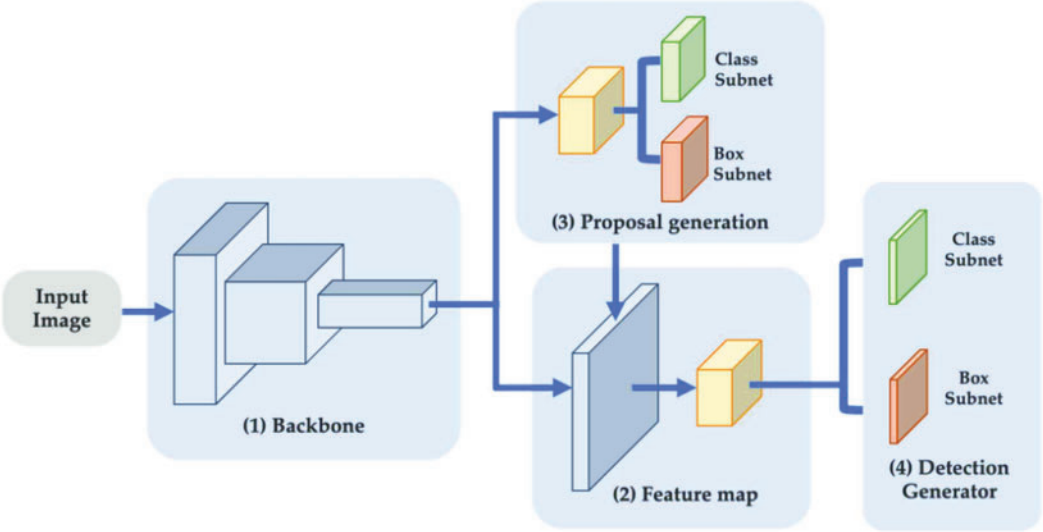


Figure 8: Basic architecture of two-stage detectors [34]

Compared to single-stage detectors, two-stage models prioritize accuracy over speed, making them suitable for applications requiring high precision, such as medical imaging and autonomous driving. This section reviews key frameworks, including R-CNN [35], Fast R-CNN [36], Faster R-CNN [37], Cascade R-CNN [38], and DetectoRS [39], outlining their evolution and functionality.

R-CNN (Region-Based Convolutional Neural Network)

R-CNN [35] introduced the concept of region proposal-based object detection by combining selective search [40] with CNN-based feature extraction. It generates region proposals by hierarchically grouping image segments based on color, texture, and shape, then processes each region independently through a CNN. The extracted features are classified using a support vector machine (SVM) [41], and bounding box regression refines object localization (Fig. 9).

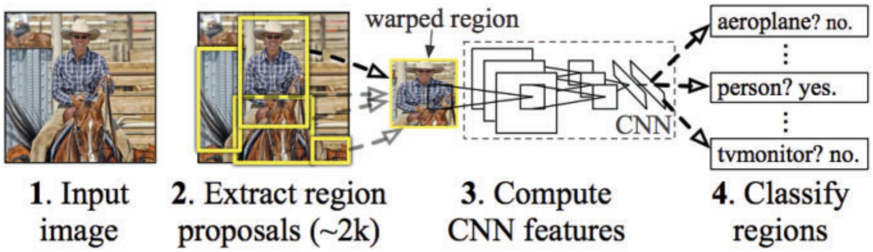


Figure 9: A flow diagram illustrating the R-CNN pipeline, showing the separate stages of region proposal, feature extraction, classification, and bounding box regression [35]

Despite its accuracy improvements over traditional methods, R-CNN suffered from inefficiencies due to its multi-stage pipeline, where each proposal was processed separately, leading to redundant computations and slow inference. Its lack of end-to-end training hindered real-time deployment, while handling occluded objects required multiple overlapping proposals, further increasing computational costs. These limitations motivated the development of Fast R-CNN, which introduced a unified feature extraction mechanism to enhance efficiency.

Fast R-CNN

Fast R-CNN [36] addressed the inefficiencies of R-CNN by introducing a shared CNN feature map for the entire image, significantly reducing computational redundancy. Instead of processing each region proposal independently, Fast R-CNN applies a single convolutional pass over the input image to generate a feature map. Region proposals, obtained using selective search [40], are projected onto this feature map through a Region of Interest (RoI) pooling layer, which extracts fixed-length feature vectors for each proposal (Fig. 10).

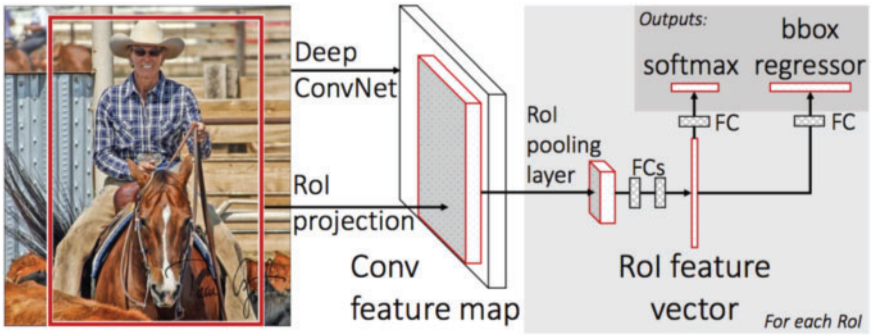


Figure 10: Architecture of Fast R-CNN, featuring a shared feature map and RoI pooling for region proposals [36]

By sharing computation across all region proposals, Fast R-CNN eliminated redundancy and allowed end-to-end training, significantly improving inference speed. However, its reliance on selective search still posed a bottleneck, limiting real-time performance. Additionally, the model remained sensitive to occlusions, as overlapping objects could lead to misaligned RoI features, reducing detection accuracy in cluttered environments.

The improvements in computational efficiency made Fast R-CNN a significant step forward, paving the way for Faster R-CNN, which further optimized the region proposal process.

Faster R-CNN

Faster R-CNN [37] improves upon Fast R-CNN by integrating the Region Proposal Network (RPN) directly into the detection pipeline, eliminating the need for selective search and enabling end-to-end training. The model extracts a shared feature map using a CNN backbone, such as VGG [42], which the RPN processes to generate object proposals. These proposals are refined through RoI pooling before classification and bounding box regression, significantly enhancing efficiency and detection accuracy (Fig. 11).

While Faster R-CNN achieves higher precision and reduced inference time, it remains sensitive to occlusions. Partial occlusions can generate incomplete region proposals, leading to false negatives, and its anchor-based mechanism may misalign detections in cluttered scenes. Despite these limitations, Faster

R-CNN set the stage for more advanced two-stage detectors, improving computational efficiency without sacrificing accuracy in complex visual environments.

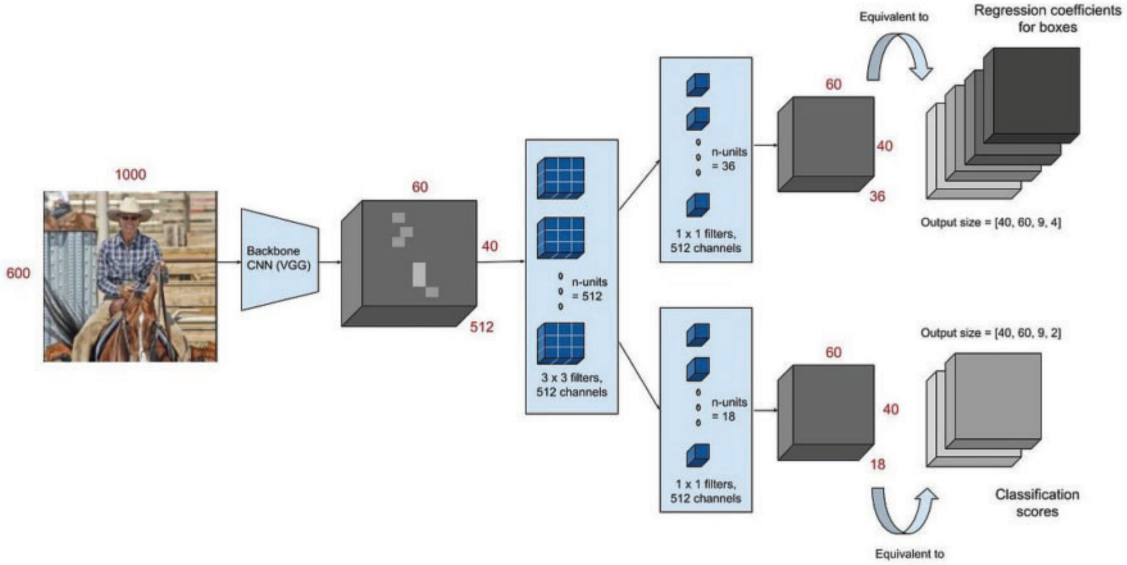

1000
600
Backbone CNN (VGG)
60
40
512
n-units = 512
3 x 3 filters, 512 channels
n-units = 36
1 x 1 filters, 512 channels
60
40
36
Equivalent to
Regression coefficients for boxes
Output size = [40, 60, 9, 4]
Output size = [40, 60, 9, 2]
60
40
18
n-units = 18
1 x 1 filters, 512 channels
Classification scores
Equivalent to

Figure 11: Architecture of Faster R-CNN, integrating an RPN for region proposal generation within the detection pipeline [37]

Cascade R-CNN

Cascade R-CNN [38] enhances Faster R-CNN by introducing a multi-stage refinement process that iteratively improves object localization and classification. Unlike traditional single-stage detectors, it employs sequential detection stages with progressively stricter Intersection over Union (IoU) thresholds, ensuring improved accuracy for objects of varying scales and occlusion levels (Fig. 12).

The model generates region proposals using an RPN, similar to Faster R-CNN, but instead of applying a fixed IoU threshold, it refines predictions through multiple cascaded stages. Early stages focus on coarse proposals, while later stages apply stricter localization criteria, reducing misaligned detections and improving object segmentation. The incorporation of a Feature Pyramid Network (FPN) further enhances multi-scale detection, making the model more effective for small and occluded objects.

The cascading structure allows for more precise detection, particularly in cluttered scenes where occlusions degrade feature quality. However, this refinement process increases computational overhead, making inference slower compared to single-stage approaches. Despite this, Cascade R-CNN remains a highly effective model for high-accuracy object detection in challenging real-world scenarios, including those with significant occlusions.

DetectoRS

DetectoRS (Detecting Objects with Recursive Feature Pyramid and Switchable Atrous Convolution) [39] enhances two-stage object detection through Recursive Feature Pyramid (RFP) and Switchable Atrous Convolution (SAC). RFP iteratively refines feature extraction through a top-down and bottom-up structure, strengthening multi-scale object detection (Fig. 13). SAC dynamically adjusts receptive fields by switching between dilation rates, improving adaptability to varying object sizes and aspect ratios (Fig. 13b).

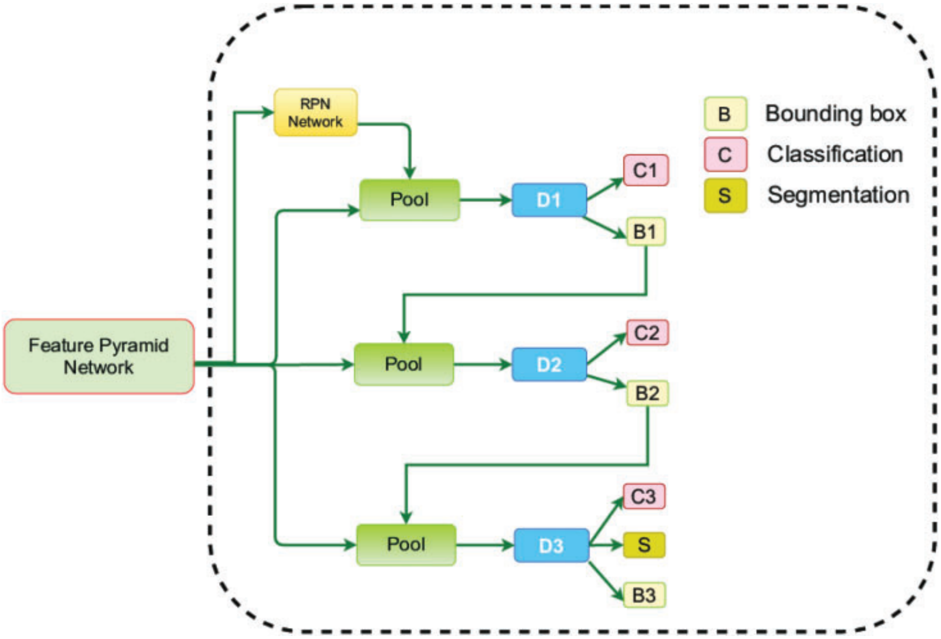
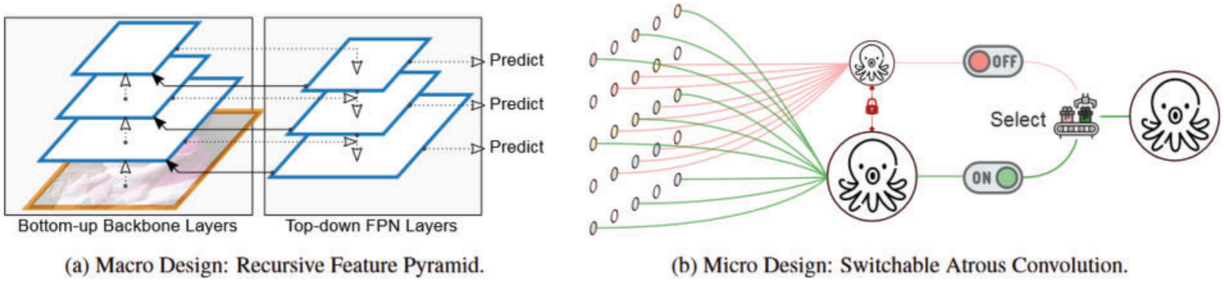


Figure 12: Cascade R-CNN architecture, illustrating iterative refinement across multiple detection stages [38]



(a) Macro Design: Recursive Feature Pyramid.

(b) Micro Design: Switchable Atrous Convolution.

Figure 13: The DetectoRS architecture, illustrating the Recursive Feature Pyramid **(a)** and Switchable Atrous Convolution **(b)** components [39]

Built upon Faster R-CNN, DetectoRS employs an RPN for proposal generation, with RFP introducing an iterative feedback loop that refines feature extraction. SAC further augments the model by adaptively modulating the receptive field, improving object boundary detection and enhancing robustness in cluttered scenes. These combined mechanisms improve object detection in complex environments, particularly when dealing with densely packed or irregularly shaped objects.

While DetectoRS enhances spatial reasoning and feature consistency, severe occlusions still disrupt its detection accuracy. Its reliance on learned priors, rather than explicit depth cues, limits robustness under extreme occlusions. Despite these limitations, DetectoRS represents a significant advancement in two-stage detection, demonstrating the benefits of recursive feature refinement and adaptive convolution.

This exploration of two-stage detectors traces their evolution from R-CNN to advanced frameworks such as Cascade R-CNN and DetectoRS. These models illustrate significant improvements in precision and efficiency for 2D object detection, providing a foundation for the upcoming discussion on one-stage detectors in the next section.

3.1.2 Single-Stage Detectors (Fully CNN-Based)

Single-stage detectors streamline object detection by integrating the entire detection pipeline into a single unified step, bypassing the need for a separate region proposal stage. This design prioritizes computational efficiency, making these models well-suited for real-time applications where speed is critical. While single-stage detectors trade some precision for faster inference compared to two-stage models, they remain effective in scenarios requiring rapid and reliable detection (Fig. 14).

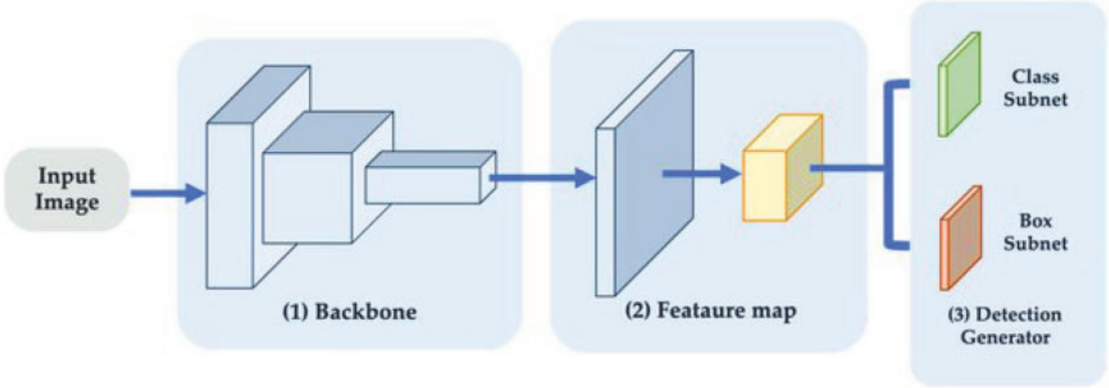


Figure 14: Basic architecture of single-stage detectors [34]

This section examines key single-stage detection frameworks, including the YOLO family [43–47], EfficientDet [48], Scaled-YOLOv4 [49], and PP-YOLOE [50], analyzing their architectural advancements and impact on real-time object detection.

YOLO Series: Evolution and Architectures

The YOLO (You Only Look Once) [43] series is a pioneering family of single-stage object detectors designed for real-time performance while maintaining accuracy. These models adopt a grid-based detection strategy, dividing the input image into cells, each responsible for predicting bounding boxes and class probabilities. Over the years, the YOLO series has undergone significant advancements, starting from YOLOv1 [43] and evolving through YOLOv2 [51], YOLOv3 [52], and more recent iterations such as YOLOv5 [44], YOLOv7 [45], YOLOv8 [46], and YOLOv10 [47]. This section presents the major innovations introduced in each version while analyzing their strengths and limitations, particularly in occlusion handling.

1. **YOLOv3**
   YOLOv3 [52] introduced anchor boxes and an FPN (Fig. 15) to enhance multi-scale detection. Unlike its predecessors, YOLOv3 employs three output layers corresponding to different feature map sizes (13 × 13, 26 × 26, and 52 × 52), enabling better detection of objects at varying scales. The model utilizes a Darknet-53 backbone, composed of residual blocks, to improve gradient flow and enhance feature extraction.
   The integration of FPN allows YOLOv3 to capture objects at different scales, improving detection in cluttered environments. However, despite its ability to process multi-scale features, YOLOv3 struggles with occlusions due to its reliance on grid-based predictions. When objects overlap significantly, the model may produce inaccurate bounding boxes as feature aggregation remains limited by the predefined anchor boxes. Additionally, the increased computational cost from multi-scale feature extraction and anchor-based predictions affects inference speed, making it less efficient for real-time applications.

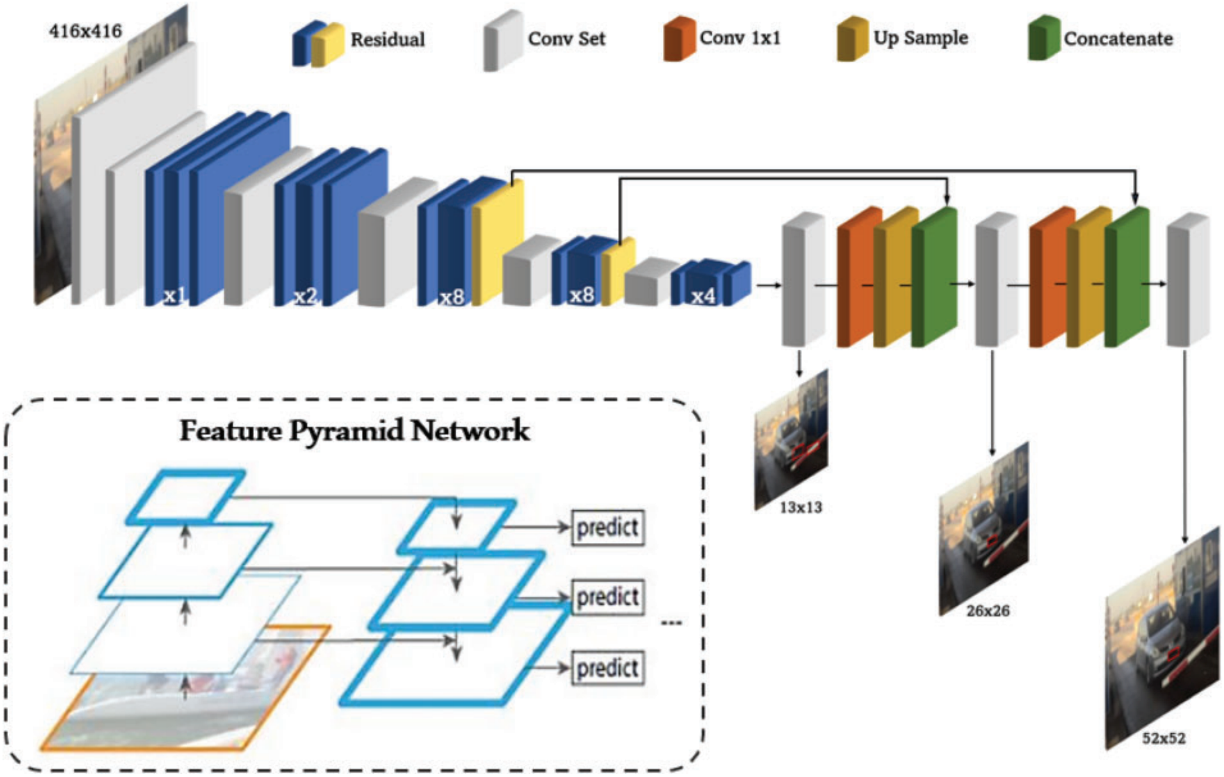


Figure 15: YOLOv3 architecture with multi-scale detection capabilities [8]

2. **YOLOv5**

   YOLOv5 [44] enhances computational efficiency with a Cross Stage Partial Network (CSPNet) backbone, reducing redundancy while improving gradient flow (Fig. 16). It integrates a Path Aggregation Network (PANet) for multi-scale feature fusion and employs Spatial Pyramid Pooling (SPP) to capture contextual information, optimizing both speed and accuracy.

   The incorporation of CSPNet and PANet improves feature propagation and enhances the model's ability to detect objects at different scales. However, YOLOv5 still faces challenges in heavily occluded environments. While its improved feature fusion mechanism provides better spatial reasoning, occlusions can still degrade detection performance, particularly when multiple objects overlap closely. Moreover, the reliance on appearance-based cues rather than explicit depth estimation contributes to reduced localization accuracy in highly cluttered scenes.

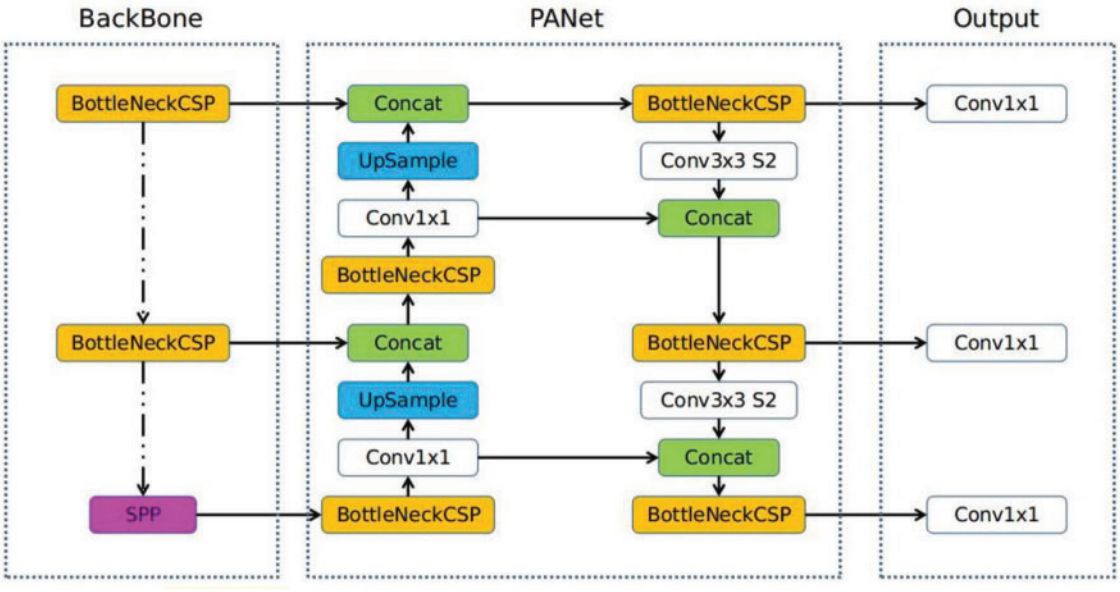


Figure 16: YOLOv5 architecture with CSPNet and PANet [44]

3. **YOLOv7**

YOLOv7 [45] introduces an Extended Efficient Layer Aggregation Network (ELAN), which improves feature extraction efficiency by splitting and merging features at multiple stages. This facilitates enhanced gradient propagation, allowing for better generalization across different object scales. The model retains FPN and PANet for multi-scale feature fusion, ensuring improved object localization (Fig. 17).

By incorporating ELAN, YOLOv7 achieves a superior balance between accuracy and inference speed. However, its performance in occlusion-heavy environments remains limited, as feature representations are still constrained to 2D spatial cues without explicit occlusion modeling. The model also requires careful post-processing to refine detections in cases of overlapping objects, where bounding box misalignment can occur.

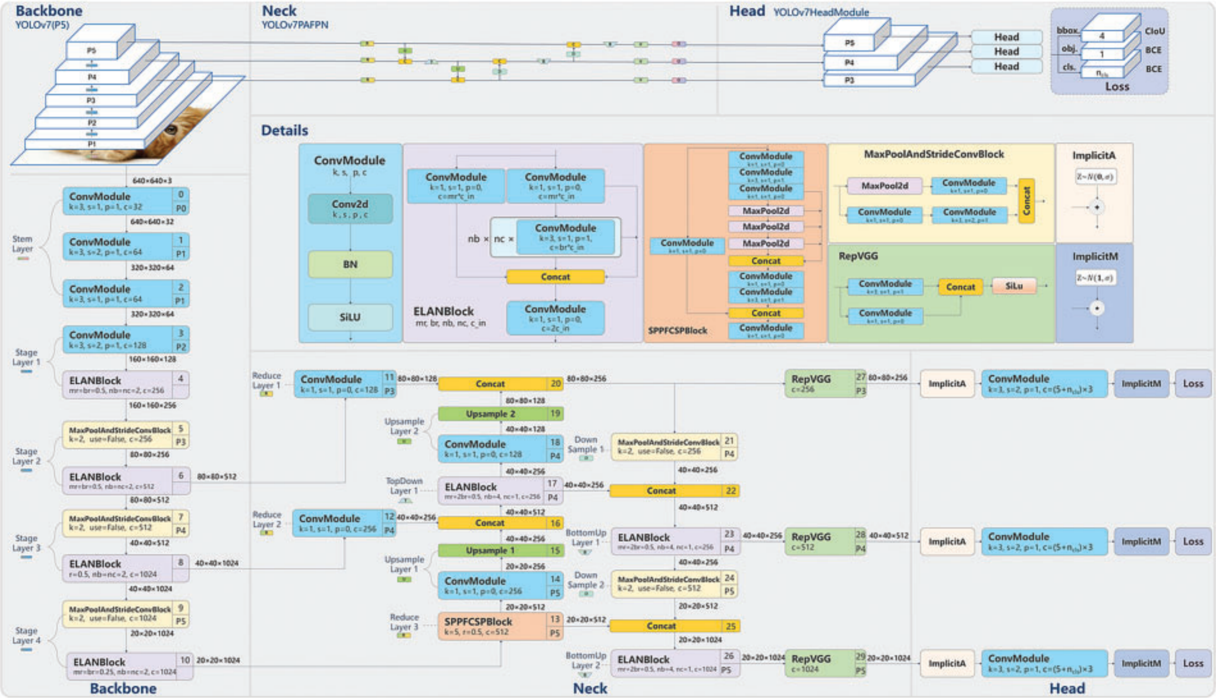
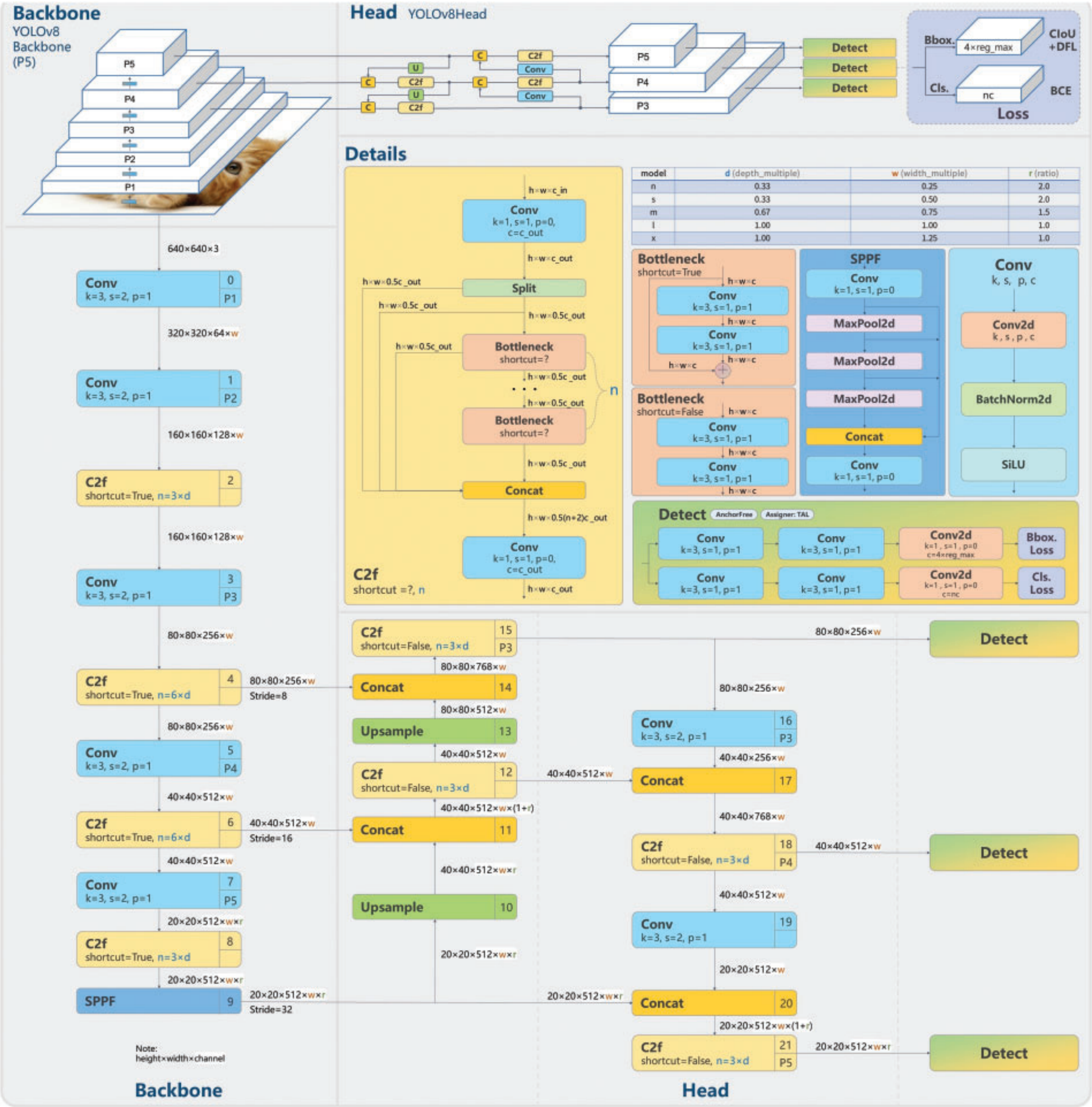


Figure 17: YOLOv7 architecture emphasizing ELAN [45]

4. **YOLOv8**

YOLOv8 [46] transitions to an anchor-free detection paradigm, removing predefined box sizes for a simpler training process. It integrates C2f modules for efficient feature reuse and employs a unified detection head for classification, localization, and segmentation (Fig. 18).

The anchor-free detection approach simplifies the model's architecture, reducing computational complexity and improving adaptability across datasets. However, while YOLOv8 enhances general detection performance, it struggles in highly cluttered environments where objects are occluded or closely packed. The absence of predefined anchors increases localization errors in such scenarios, necessitating additional refinements in post-processing to improve robustness against occlusions.



model	d (depth_multiple)	w (width_multiple)	r (ratio)
n	0.33	0.25	2.0
s	0.33	0.50	2.0
m	0.67	0.75	1.5
l	1.00	1.00	1.0
x	1.00	1.25	1.0

Figure 18: YOLOv8 architecture, showcasing its anchor-free detection [46]

5. **YOLOv10**

YOLOv10 [47] introduces a dual-label assignment strategy, integrating one-to-many and one-to-one matching mechanisms for classification and regression tasks. This refinement enhances detection in occluded scenes by improving object representation (Fig. 19).

The integration of dual-label assignment improves occlusion robustness by refining spatial alignment and detection confidence. The model effectively balances recall and precision, making it more resilient to object overlap. However, despite these advancements, YOLOv10 still lacks explicit depth reasoning, limiting its performance in extreme occlusion scenarios. The added complexity from the matching strategy also increases computational requirements, making it less suitable for real-time low-resource applications.

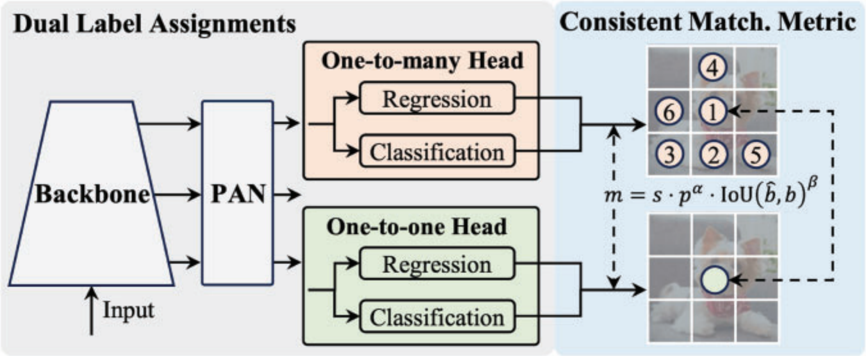


Figure 19: YOLOv10 with dual-label assignment and refined spatial alignment [47]

The YOLO series has evolved significantly, introducing improvements in feature extraction, multi-scale fusion, and detection efficiency. While these advancements enhance real-time detection capabilities, occlusion remains a key limitation due to the reliance on appearance-based feature extraction rather than explicit spatial depth modeling. Addressing these challenges requires further innovations in occlusion-aware architectures that integrate both 2D and 3D contextual information.

EfficientDet

EfficientDet [48] builds upon the EfficientNet backbone [53], employing a compound scaling method to optimize network depth, width, and resolution. A key innovation is the Bi-directional Feature Pyramid Network (BiFPN), which enhances multi-scale feature fusion by incorporating bidirectional connections and learnable feature importance weights (Fig. 20). Unlike traditional FPNs, BiFPN adaptively prioritizes the most relevant features, improving detection performance while maintaining computational efficiency.

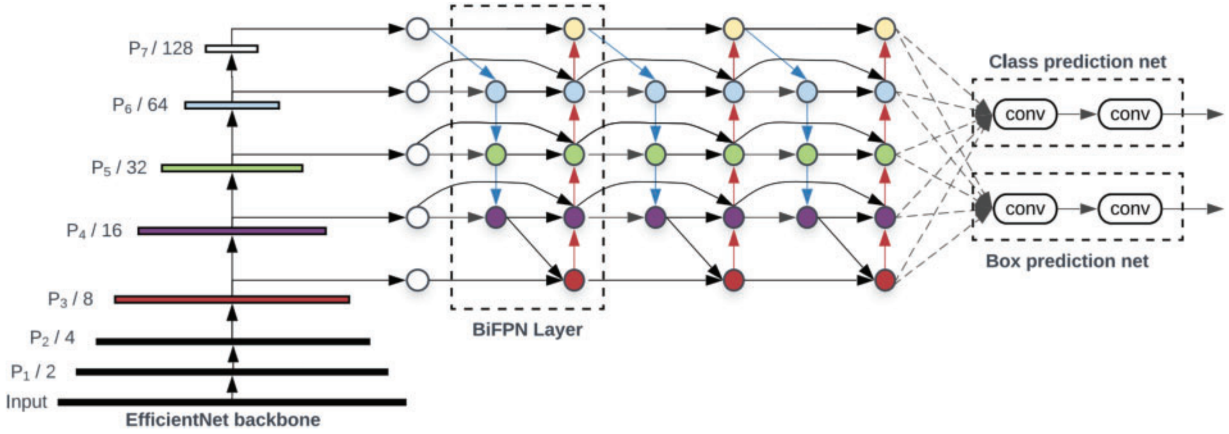


Figure 20: BiFPN structure in EfficientDet, illustrating its multi-scale feature fusion process [48]

While EfficientDet balances accuracy and efficiency, its reliance on feature fusion rather than explicit depth estimation makes it vulnerable to occlusions. In heavily cluttered scenes, missing object details may not be fully reconstructed, leading to detection failures. However, its lightweight architecture and high efficiency make it well-suited for real-time applications where computational constraints exist. By leveraging

EfficientNet's backbone and BiFPN's adaptive feature refinement, EfficientDet remains a competitive choice in single-stage detection models [48].

PP-YOLOE

PP-YOLOE [50] enhances the PP-YOLO series [54] by integrating a CSPRepResNet backbone, PAN neck, and Efficient Task-aligned Head (ET-head), optimizing both speed and accuracy (Fig. 21). The CSPRepResNet backbone employs RepResBlocks to improve feature extraction efficiency while minimizing inference overhead. Additionally, Effective Squeeze and Extraction (ESE) layers enhance channel-wise attention, refining object representations across different scales [50].

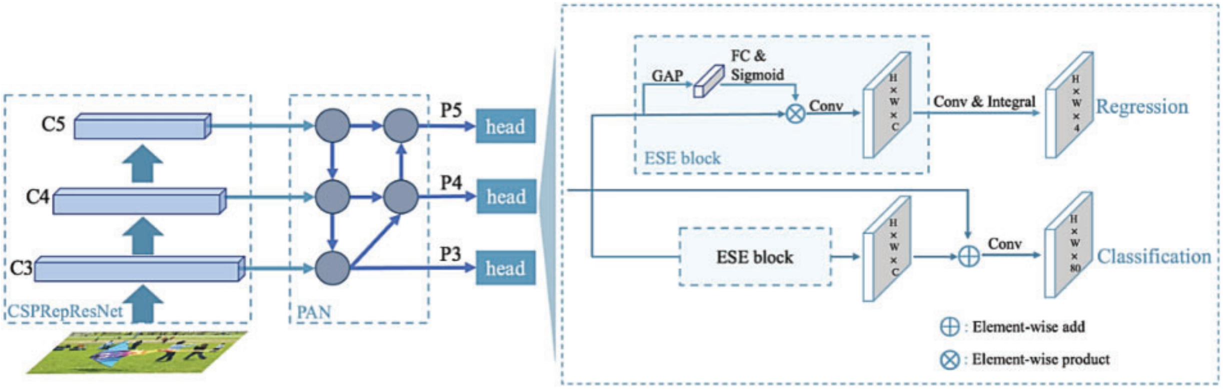


Figure 21: PP-YOLOE architecture, integrating CSPRepResNet, PAN, and ET-head [50]

The PAN neck improves multi-scale feature fusion by propagating fine-grained spatial details, enhancing detection across varying object sizes. The ET-head introduces lightweight IoU-aware layers, refining bounding box localization and confidence estimation [50]. These architectural optimizations enable PP-YOLOE to maintain high real-time performance while improving robustness against occlusions through better spatial reasoning and feature alignment.

The advancements in single-stage detectors highlight their pivotal role in achieving a balance between accuracy and real-time efficiency. These innovations have not only reinforced the foundations of 2D object detection but have also paved the way for more complex spatial representations, including 3D. As detection tasks increasingly demand enhanced depth perception and contextual awareness, the transition from 2D to 3D object detection becomes essential for applications such as autonomous systems and smart surveillance. This progression underscores the necessity of models that effectively integrate spatial depth information while maintaining computational efficiency, setting the stage for the exploration of 3D object detection approaches in the following section.

3.2 3D Object Detection

The advancement of 3D object detection has led to the development of diverse methodologies that process spatial information using different data representations. Unlike 2D models that rely solely on image-based features, 3D detection methods incorporate depth cues from point clouds, voxel grids, and multi-view representations, enabling improved spatial reasoning. These approaches are categorized into voxel-based, point-based, and hybrid models, each addressing challenges such as data sparsity, occlusion handling, and computational efficiency.

3.2.1 Voxel-Based Approaches

Voxel-based methods transform raw point cloud data into structured volumetric representations, allowing efficient feature extraction using 3D convolutional networks. By discretizing 3D space into a voxel grid, these methods facilitate robust spatial analysis, making them particularly effective in large-scale environments. However, voxelization introduces trade-offs between accuracy and computational complexity, requiring optimization strategies to balance performance and efficiency.

1. **VoxelNet**
   VoxelNet [55] is one of the pioneering voxel-based object detection models, integrating voxelization and feature learning into a unified framework. The pipeline begins by partitioning the input point cloud into a voxel grid, converting the unstructured data into a format compatible with 3D convolutions. A Voxel Feature Encoding (VFE) layer processes each voxel using shared Multi-Layer Perceptrons (MLPs) to extract local geometric and spatial features. These features are aggregated into a single descriptor per voxel, forming the foundation for higher-level feature extraction. The voxelization process and grouping strategy play a crucial role in structuring the raw point cloud for subsequent learning stages (Fig. 22).
   Despite its advantages in enhancing spatial reasoning and occlusion handling, VoxelNet suffers from high computational complexity due to dense voxelization, making real-time deployment challenging. Additionally, the quantization process in voxelization can introduce boundary inaccuracies, particularly in occluded or small-object scenarios. These limitations have led to the development of more efficient voxel-based models, focusing on improving sparsity handling and computational efficiency while maintaining robust 3D object detection performance.

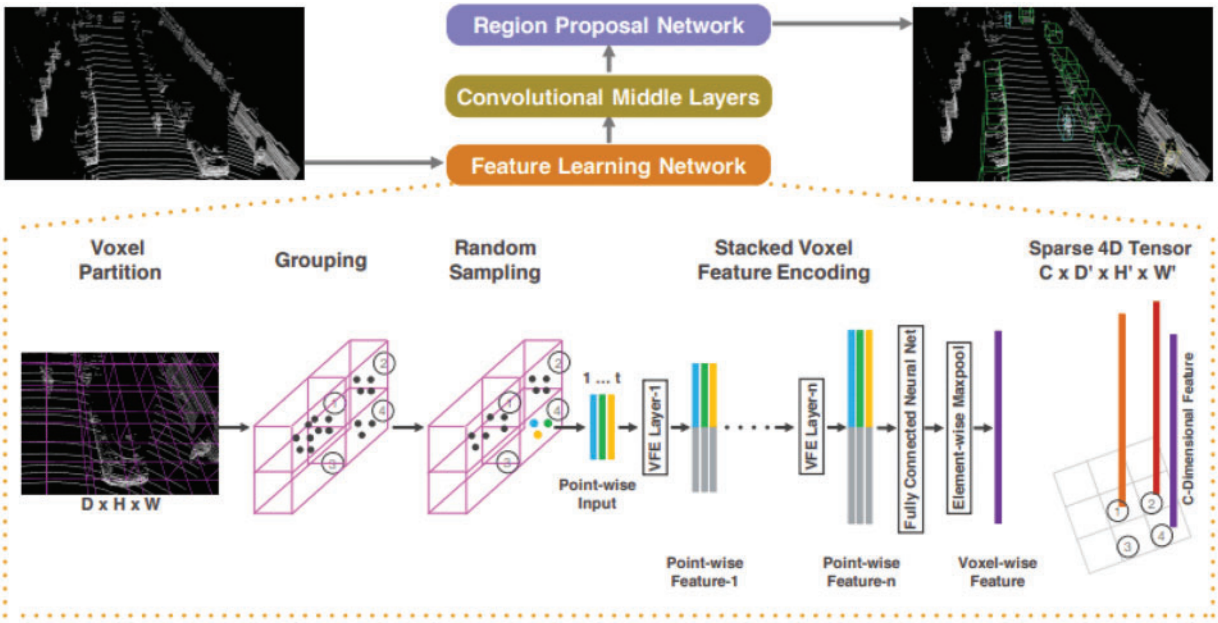


Figure 22: VoxelNet architecture illustrating the voxelization process, VFE layer for local feature aggregation, and 3D convolutional layers for hierarchical feature extraction [55]

2. **SECOND**
   SECOND (Sparsely Embedded Convolutional Detection) [56] enhances voxel-based 3D object detection by integrating sparse convolutional operations, significantly reducing computational costs while

maintaining strong detection accuracy. Unlike traditional 3D convolutional methods that process all voxels, SECOND selectively processes only non-empty voxels, improving efficiency without sacrificing spatial resolution. The voxelized input is first passed through a voxel feature extractor, which aggregates spatial information before feeding it into sparse convolution layers. This method preserves crucial geometric details while minimizing redundant computations (Fig. 23).

By leveraging sparse convolutions, SECOND improves inference speed, making it more viable for real-time applications compared to dense voxel-based models like VoxelNet. However, the voxelization process still introduces quantization errors, affecting object boundary precision. Additionally, while sparse convolutions optimize feature extraction, they become less effective in highly dense object clusters, where occlusions and overlapping structures can lead to feature fragmentation and reduced detection robustness.

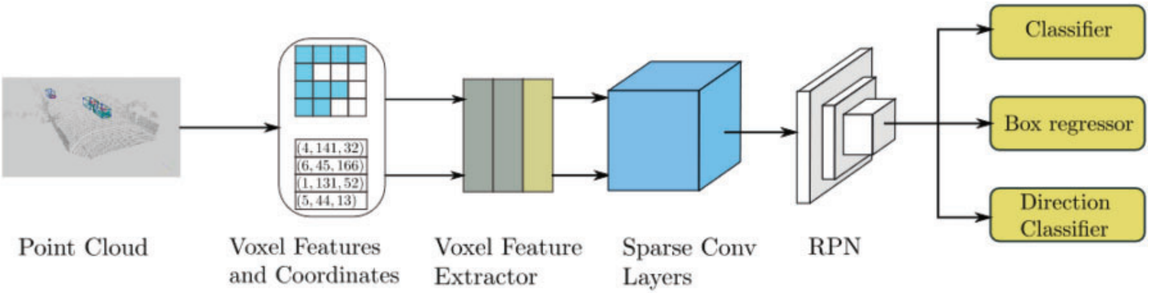


Figure 23: SECOND architecture: Sparse convolution operations applied to voxel grids, followed by their integration into 3D convolutional layers [56]

3.2.2 Point-Based Approaches

Point-based methods directly operate on raw point clouds, preserving the original 3D spatial structure without voxelization. These approaches maintain fine-grained geometric details, enabling precise shape and structure representation. By leveraging point-wise feature extraction, point-based models excel in scenarios where maintaining geometric fidelity is critical. However, their reliance on unordered point sets and high computational complexity presents challenges in large-scale 3D object detection.

1. **PointNet**

   PointNet [57] introduced a paradigm shift in 3D object detection by processing raw point clouds directly, bypassing voxelization. Each point is treated as an independent entity, with shared Multi-Layer Perceptrons (MLPs) encoding per-point features such as curvature and local density. To ensure geometric transformation invariance, PointNet incorporates a T-Net module, which learns a transformation matrix to align input points.

   Aggregation is performed through a symmetric max-pooling operation, which condenses the extracted features into a global feature vector representing the entire point cloud. This feature vector is used for classification, segmentation, and bounding box regression. For segmentation, additional layers refine point-wise predictions by integrating global and local features (Fig. 24).

   While PointNet effectively processes unordered point clouds, its reliance on global pooling limits its ability to capture fine-grained local structures. This constraint impacts object localization and segmentation in occlusion-heavy environments, where detailed spatial relationships are essential.

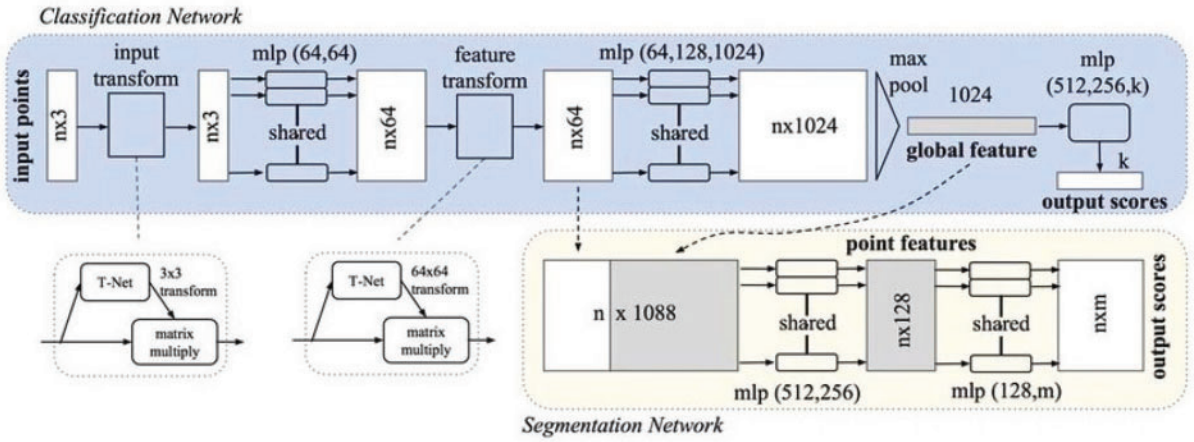


Figure 24: PointNet architecture, showing point-wise feature extraction with MLPs and global feature aggregation via max-pooling [57]

2. **PointNet++**
   PointNet++ [58] enhances PointNet by introducing a hierarchical structure that groups points into local neighborhoods based on spatial proximity. Shared MLPs extract features from each local region, which are aggregated hierarchically to capture geometric structures at multiple scales (Fig. 25).

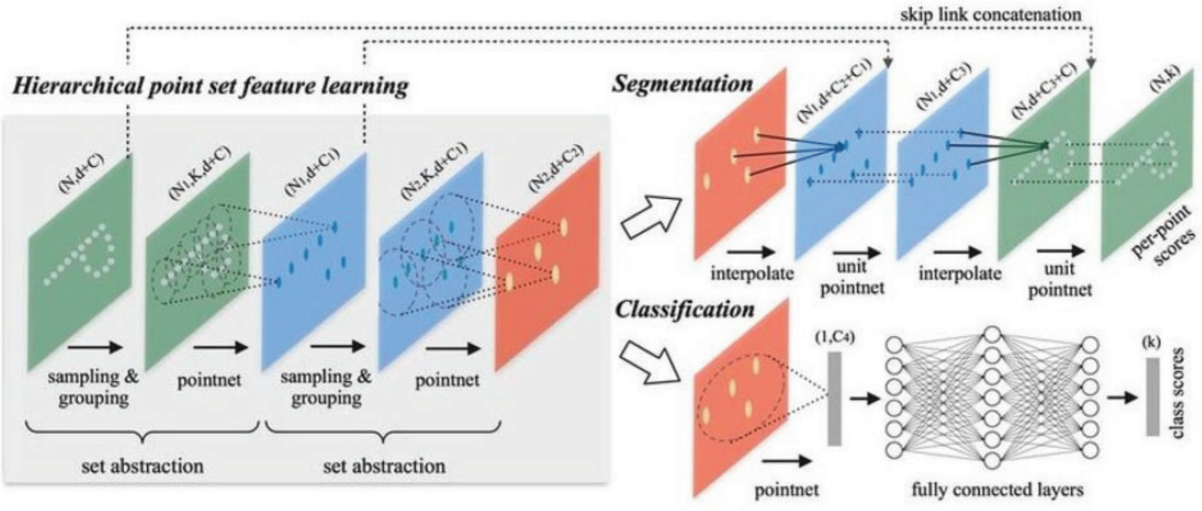


Figure 25: Hierarchical feature extraction in PointNet++, illustrating local grouping and multi-level pooling [58]

The hierarchical architecture consists of the following steps:

- **Set Abstraction:** Points are sampled and grouped into local regions using k-nearest neighbors (k-NN) or ball query methods. Shared MLPs extract local features from each group.
- **Multi-Level Pooling:** Local features from multiple regions are aggregated hierarchically, allowing the model to capture geometric structures at different scales.
- **Skip Link Concatenation:** Features from earlier layers are concatenated with higher-level representations, preserving fine-grained details for segmentation tasks.

The hierarchical approach enables PointNet++ to adapt to varying point densities, addressing challenges commonly encountered in outdoor LiDAR scans. Its ability to model local dependencies significantly improves occlusion handling, enhancing detection robustness in cluttered 3D scenes.

3.2.3 Hybrid Approaches

Hybrid approaches combine voxel-based and point-based techniques to leverage the structured representation of voxels and the fine-grained detail of point clouds. By integrating these complementary methods, hybrid models achieve a balance between computational efficiency and detailed geometric representation, enabling robust and accurate 3D object detection. These approaches are particularly effective in addressing challenges such as data sparsity and irregularity, commonly encountered in 3D data.

1. **PV-RCNN**
   PV-RCNN (Point-Voxel Region-Based CNN) [59] integrates voxel-based and point-based strategies to enhance 3D object detection. The architecture first voxelizes the raw point cloud to extract global structural features using sparse 3D CNNs. Simultaneously, key points are sampled from the raw point cloud to capture fine-grained local details. These global voxel features and local point features are then fused using a voxel set abstraction module to generate a comprehensive scene representation (Fig. 26).
   The detection pipeline consists of two stages: an RPN-based proposal generation and a refinement stage utilizing RoI-grid pooling. The predicted keypoint weighting module enhances object proposals by refining feature aggregation. This structure ensures precise classification and bounding box regression, improving the model's performance in challenging scenarios with sparse or occluded environments.
   By integrating the advantages of both voxel-based and point-based techniques, PV-RCNN reduces computational overhead while maintaining high detection accuracy. Its two-level fusion mechanism enhances robustness, making it effective for 3D object detection tasks in large-scale, unstructured environments.

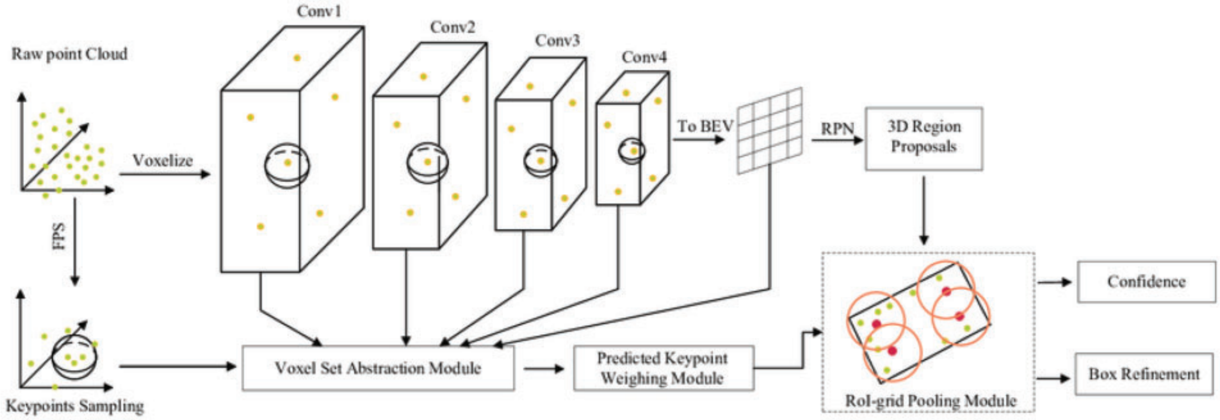


Figure 26: PV-RCNN architecture, illustrating voxel-based global features and point-based local features fusion [59]

2. **VoteNet**
   VoteNet [60] introduces a novel voting mechanism for 3D object detection in raw point clouds, diverging from conventional CNN- or voxel-based methodologies. Each point in the input cloud predicts offsets towards potential object centers, generating object proposals that are iteratively refined through feature propagation layers and attention mechanisms (Fig. 27).

The architecture incorporates attention modules, filtering out irrelevant spatial features while emphasizing informative regions within the point cloud. This attention-enhanced feature propagation enables VoteNet to differentiate objects from cluttered backgrounds, making it particularly effective in occlusion-heavy scenarios.

The model's hierarchical design balances efficiency and accuracy, employing set abstraction layers and adaptive feature refinement to detect objects of varying scales and positions. VoteNet's ability to model fine-grained geometric structures without voxelization makes it well-suited for applications such as indoor scene understanding and robotic perception.

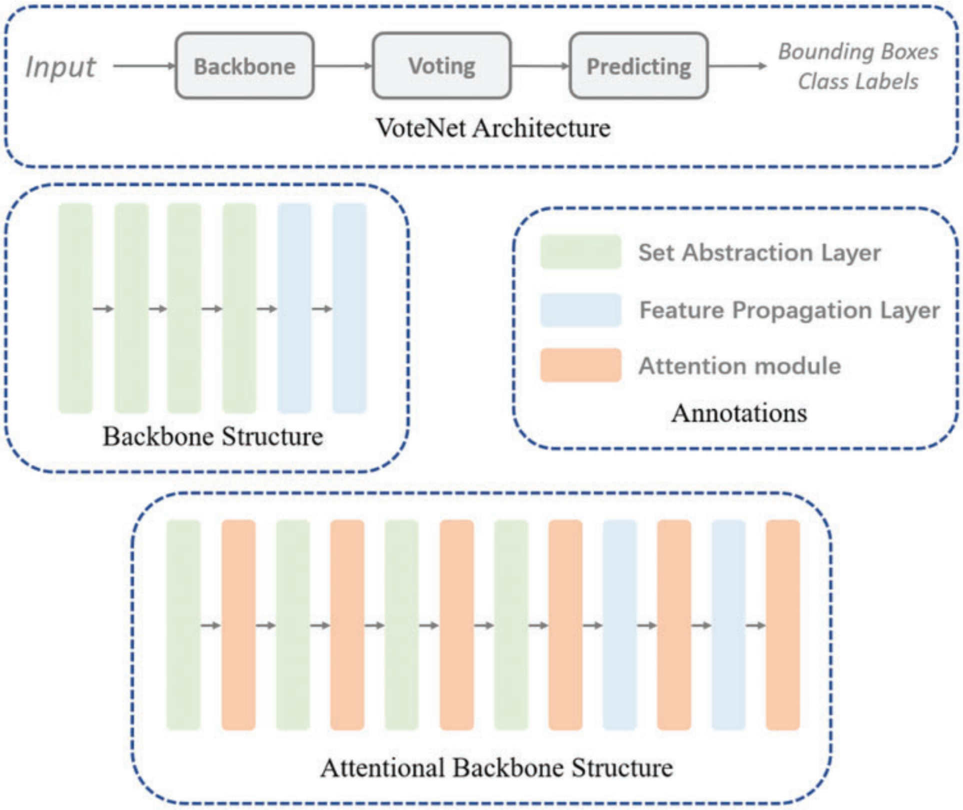


Figure 27: VoteNet's voting mechanism, where points generate and refine object proposals through offset predictions [60]

3.3 Transformer-Based and Multimodal Detection Models

Recent advances in object detection have introduced a new generation of architectures that leverage transformer-based mechanisms and multimodal data fusion. Unlike traditional CNN-based detectors, these models are explicitly designed to capture long-range spatial dependencies, align features across multiple views, and integrate complementary modalities such as LiDAR, RGB, and depth. These capabilities make them particularly well-suited for addressing occlusion-related challenges, where visual information from a single modality or limited receptive field often proves insufficient.

Although not included in our comparative evaluation due to their computational complexity and dataset-specific implementation requirements, these state-of-the-art models represent an important shift in the object detection landscape. In this section, we briefly present the design principles behind key transformer-based and multimodal fusion detectors and highlight their relevance as promising directions for future occlusion-aware detection frameworks.

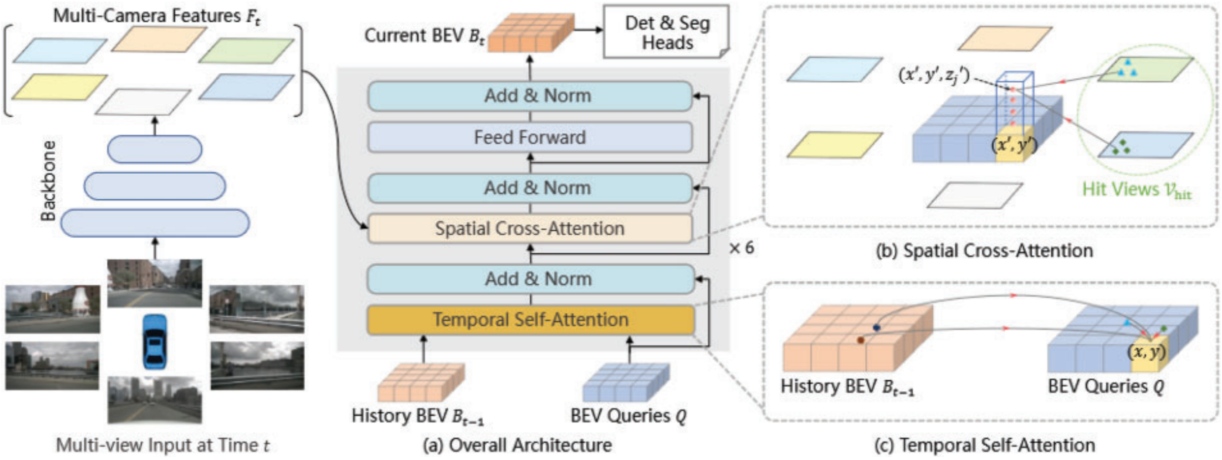
3.3.1 Transformer-Based Models

Transformer-based object detection models have recently emerged as powerful alternatives to traditional CNN-based architectures, particularly in complex scenarios involving occlusion and long-range spatial reasoning. These models leverage self-attention mechanisms to capture global contextual dependencies, making them more adept at handling partially visible objects, aligning features across multiple perspectives, and reasoning about object relationships in cluttered environments. In contrast to earlier approaches that rely heavily on localized receptive fields, transformers provide holistic scene understanding, which is critical for occlusion-aware detection tasks in both indoor and outdoor applications.

BEVFormer

BEVFormer [61] is a transformer-based model designed to generate bird's-eye-view (BEV) representations from multi-camera inputs, enabling robust perception in autonomous driving tasks. The model aggregates spatial and temporal features using attention mechanisms, facilitating accurate object detection and scene understanding across time.

As shown in Fig. 28, the BEVFormer encoder comprises six transformer layers tailored with three key components: *grid-shaped BEV queries*, *spatial cross-attention*, and *temporal self-attention*. BEV queries serve as learnable embeddings arranged over a 2D spatial grid, which attend to image features from multiple views.



(a) Overall Architecture

(b) Spatial Cross-Attention

(c) Temporal Self-Attention

Figure 28: Overview of BEVFormer architecture. The model processes multi-view inputs through a backbone and fuses spatiotemporal features using spatial cross-attention and temporal self-attention mechanisms to refine BEV representations [61]

During inference at time t, multi-view images are processed by a backbone network to extract camera-specific feature maps $F_t = \{F_t^i\}_{i=1}^{N_{\text{view}}}$. Simultaneously, the BEV features B_{t-1} from the previous timestep are stored. Each BEV queryQ interacts with B_{t-1} using temporal self-attention to encode motion and scene continuity. Then, it attends to the multi-camera features F_t using spatial cross-attention, focusing only on relevant regions of interest.

This two-step attention process allows BEVFormer to produce refined BEV embeddings B_t that integrate both temporal context and spatial observations. These are further processed by detection and segmentation heads to generate outputs such as 3D bounding boxes and semantic maps.

PETR

PETR (Position Embedding Transformation for 3D Object Detection) [62] introduces a pure transformer-based pipeline for 3D detection using multi-view 2D images, without relying on LiDAR or depth sensors. It reformulates 3D object detection as a set prediction task and integrates 3D positional encoding directly into the transformer framework to enhance geometric reasoning from 2D inputs.

As illustrated in Fig. 29, the pipeline begins by passing multi-view images through a shared 2D convolutional backbone (e.g., ResNet) to extract spatial image features. These features are then combined with 3D spatial priors generated by a dedicated *3D Coordinates Generator*, which discretizes the shared camera frustum space into a structured 3D meshgrid. Using known intrinsic and extrinsic camera parameters, the meshgrid is projected from the frustum to the 3D world space, enabling spatial grounding of the input features.

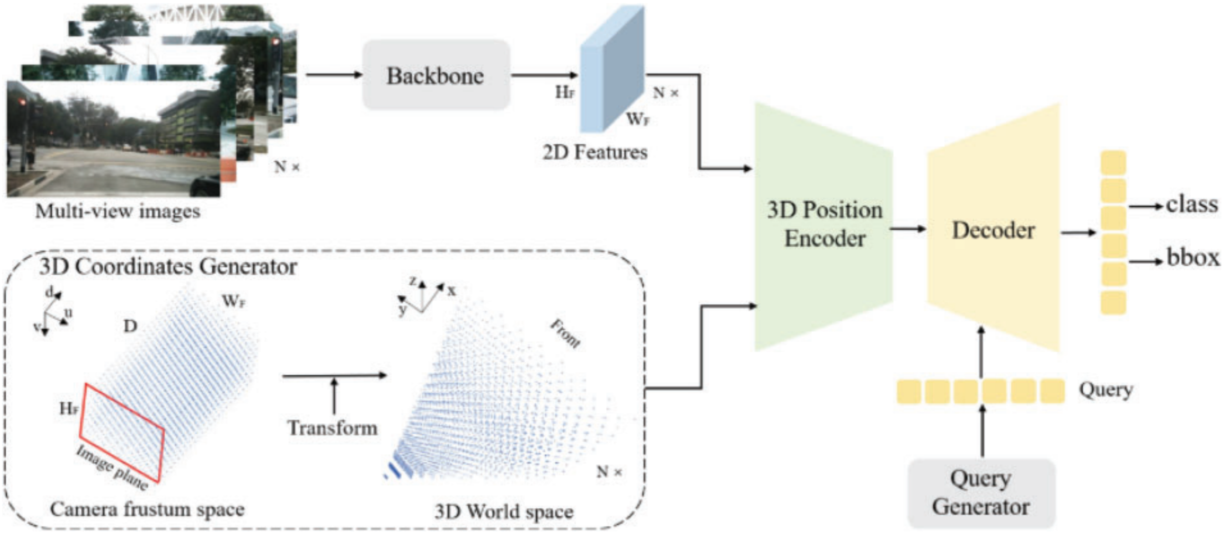


Figure 29: PETR architecture. Multi-view images are processed through a backbone and combined with 3D coordinates derived from the camera frustum. A 3D position encoder injects this spatial context into the transformer decoder, enabling direct prediction of 3D bounding boxes and classes from image-based queries [62]

The resulting 3D coordinates and 2D image features are injected into a *3D Position Encoder*, which creates position-aware embeddings that encode geometric context. These are passed to a standard transformer decoder, where learnable object queries, generated by a *Query Generator*, interact with the encoded 3D-aware features to update their semantic and localization information.

The updated queries are finally decoded into 3D bounding box coordinates and class predictions. Notably, PETR avoids intermediate BEV transformation, directly modeling 3D spatial relationships from 2D feature space, making it efficient for camera-only setups while retaining strong geometric understanding.

DETR3D

DETR3D (3D Detection Transformer) [63] is a transformer-based architecture that leverages multi-view images to directly predict 3D object bounding boxes without the need for explicit 3D reconstruction or post-processing steps like Non-Maximum Suppression (NMS). Its key innovation lies in alternating between 2D and 3D computations to maintain spatial awareness throughout the pipeline.

As illustrated in Fig. 30, the pipeline begins with a set of RGB images from multiple calibrated cameras, where intrinsic and extrinsic parameters are known. These images are passed through a shared ResNet+FPN backbone to extract dense 2D feature maps. Instead of generating dense voxelized 3D grids, DETR3D generates a sparse set of object queries, each of which corresponds to a 3D reference point in space.

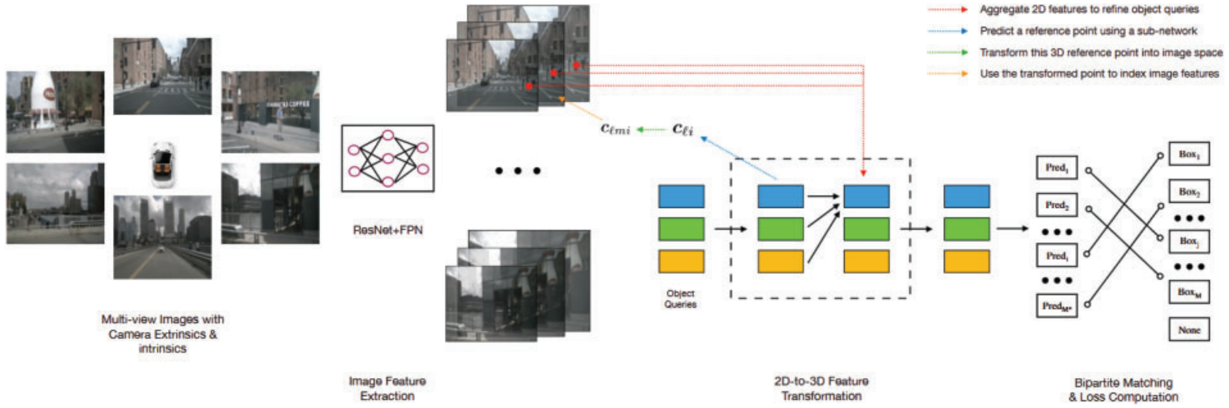

Aggregate 2D features to refine object queries
Predict a reference point using a sub-network
Transform this 3D reference point into image space
Use the transformed point to index image features
$c_{\ell mi}$
$c_{\ell i}$
ResNet+FPN
Object Queries
Pred_1
Pred_2
Pred_i
Pred_{M^*}
Box_1
Box_2
Box_j
Box_M
None
Multi-view Images with Camera Extrinsics & intrinsics
Image Feature Extraction
2D-to-3D Feature Transformation
Bipartite Matching & Loss Computation

Figure 30: DETR3D architecture: object queries interact with multi-view image features through 2D-3D projections and attention modules to predict 3D bounding boxes without dense reconstruction [63]

These 3D points are projected into the 2D image plane using camera parameters, allowing DETR3D to sample the corresponding image features via bilinear interpolation. A 2D-to-3D transformation module then fuses these sampled features with the original query embedding, refining the queries in a geometry-aware manner.

The queries are then passed through multiple transformer layers, where self-attention and cross-attention modules iteratively enhance the features by modeling inter-query dependencies and their relationships with the extracted 2D features. This process effectively links 2D and 3D representations, enabling robust spatial reasoning even under occlusion.

DETR3D is trained using a set-to-set matching loss inspired by DETR, ensuring one-to-one matching between predicted boxes and ground truth annotations. This architecture avoids NMS and dense 3D supervision, making it both computationally efficient and robust to visual clutter or overlapping objects.

3.3.2 Multimodal Fusion Networks

Multimodal fusion networks aim to combine complementary information from multiple sensor modalities, typically RGB images and LiDAR point clouds, to improve object detection performance, especially under challenging conditions such as occlusion or sparse observations. Unlike single-modal systems, these approaches benefit from the rich semantic content of images and the accurate depth perception of LiDAR. Recent transformer-based fusion models leverage attention mechanisms to align and integrate these modalities at various stages of the detection pipeline. In the following, we review representative methods that illustrate this paradigm, starting with TransFusion.

TransFusion

TransFusion [64] is a transformer-based multimodal detection network that fuses information from LiDAR point clouds and RGB images for 3D object detection. Unlike traditional fusion strategies that operate at an early or late stage, TransFusion adopts a **two-stage query-based fusion mechanism** that enables rich, category-aware, and spatially aligned feature aggregation.

As shown in Fig. 31, TransFusion uses two separate backbones to extract features from both **LiDAR BEV maps** (top stream) and **camera images** (bottom stream). A sparse set of **object queries** is initialized from the LiDAR features, guided by an image-aware query initialization strategy. These queries are then processed sequentially through two transformer decoder layers:

- The **first decoder** focuses on LiDAR features and outputs an initial prediction using object queries learned from the BEV space.
- The **second decoder**, enhanced with **Spatially Modulated Cross Attention (SMCA)**, fuses these initial predictions with RGB image features. The SMCA module introduces locality bias to help queries focus on the most relevant image regions, improving texture and appearance modeling.

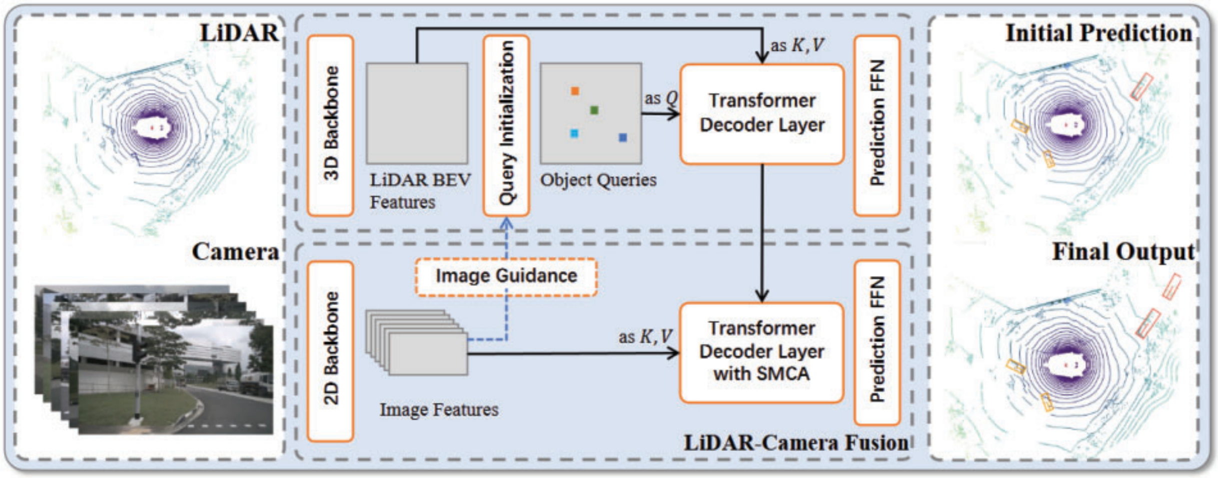


Figure 31: TransFusion architecture: LiDAR and image features are fused via two-stage transformer decoders. Initial queries from BEV features are refined through spatially modulated cross attention (SMCA) using image features, improving occlusion robustness and object localization [64]

This progressive refinement allows TransFusion to leverage both precise geometric structure from LiDAR and rich semantic cues from images. The fusion enhances the detection of small, occluded, or sparsely represented objects, particularly in cluttered scenes, by reinforcing weak LiDAR signals with complementary image guidance.

FusionFormer

FusionFormer [65] is a transformer-based multimodal detection model that integrates multi-view image features and LiDAR point cloud data to generate robust bird's-eye-view (BEV) representations. As shown in Fig. 32, the model begins by extracting 2D features from RGB images and voxel-based features from LiDAR points through separate backbones. These features are then processed by a *Multimodal Fusion Encoder (MMFE)*, which performs self-attention and cross-attention operations to jointly encode spatial cues from both modalities into BEV queries.

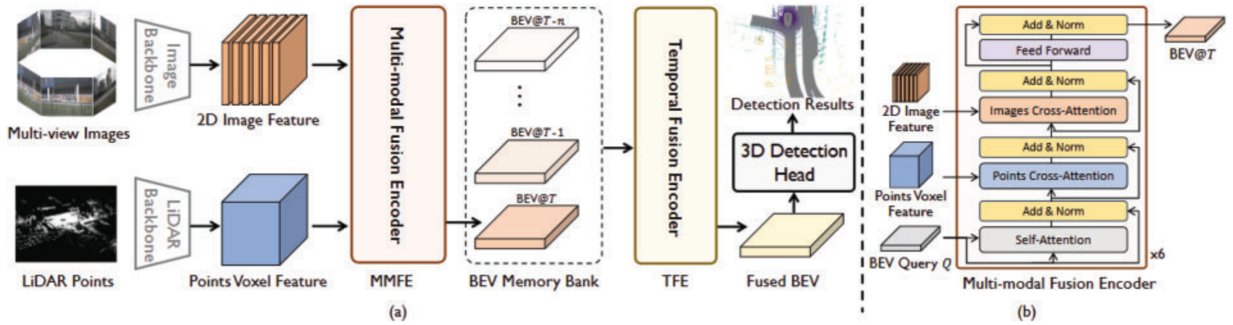


Figure 32: FusionFormer architecture: (a) Overall detection pipeline with multimodal fusion and temporal encoding; (b) Internal structure of the Multimodal Fusion Encoder (MMFE), showing sequential attention operations on BEV queries [65]

The generated BEV features are not only based on the current frame but are also temporally enriched using a *Temporal Fusion Encoder (TFE)*. The TFE aggregates sequential BEV representations stored in a memory bank to enhance temporal context awareness and continuity, which is especially valuable in occluded or motion-rich scenes. This temporal fusion produces the final fused BEV representation, which is then fed into a 3D detection head to output bounding boxes and class predictions.

As illustrated in Fig. 32, the MMFE module utilizes a stack of transformer layers, where BEV queries are iteratively updated through self-attention, image cross-attention, and voxel feature cross-attention mechanisms. This layered fusion allows FusionFormer to exploit complementary strengths of color-rich image features and geometry-aware LiDAR features, improving detection robustness in visually ambiguous scenarios.

3.4 Summary

Object detection has evolved with both 2D and 3D approaches, each addressing specific challenges while exhibiting inherent limitations. 2D object detection models, such as Faster R-CNN [37] and YOLOv5 [44], achieve high accuracy and efficiency through hierarchical feature extraction, multi-scale processing, and anchor-based predictions. However, their reliance on 2D appearance cues limits their robustness in occluded environments, where overlapping objects cause localization errors. Although later iterations, such as YOLOv8 [46] and YOLOv10 [47], introduced anchor-free detection and adaptive feature fusion, 2D models inherently lack depth perception, making them susceptible to severe occlusions.

3D object detection methods address these limitations by incorporating spatial depth information. Voxel-based methods, such as VoxelNet [55] and SECOND [56], enable structured 3D feature extraction using voxel grids, enhancing object localization under occlusion. However, dense voxelization increases computational cost, limiting real-time applicability. Point-based techniques, including PointNet [57] and PointNet++ [58], operate directly on raw point clouds, capturing fine-grained geometric details. Yet, they struggle with sparsity and uneven point distributions, affecting detection accuracy. Hybrid models, such as PV-RCNN [59] and VoteNet [60], integrate voxel- and point-based features to balance efficiency and precision, yet occlusion-induced feature loss remains a challenge.

To address the shortcomings of traditional architectures in occlusion-heavy settings, recent works have explored transformer-based and multimodal fusion paradigms. Transformer-based 3D detectors such as BEVFormer [61], PETR [62], and DETR3D [63] leverage spatiotemporal and multi-view attention mechanisms to better align features across frames and camera perspectives, enhancing object visibility even under partial occlusion. These models encode scene geometry into learnable queries, which interact with image features to infer precise 3D object representations. Additionally, multimodal fusion networks, including TransFuser [64] and FusionFormer [65], integrate LiDAR, RGB, and contextual data using advanced fusion modules such as cross-attention and temporal encoding. By combining the complementary strengths of 2D appearance and 3D geometry, these models improve detection robustness in complex urban environments.

Beyond occlusion, environmental factors such as fog, rain, and poor lighting further degrade detection performance. LiDAR-based models are sensitive to point cloud sparsity in adverse weather, while stereo and ToF-based depth estimation methods suffer from measurement inconsistencies due to reflectance variations. These challenges highlight the trade-offs between different architectures, necessitating a comparative analysis to evaluate their effectiveness in occlusion handling, computational efficiency, and real-world deployment.

To aid this comparison, we summarize the core characteristics of each model category–2D, 3D, transformer-based, and multimodal–in a synthetic overview Table 1, highlighting their respective contributions and limitations with respect to occlusion handling and sensor fusion. The following section builds on this comparison to guide future design choices in occlusion-aware object detection.

Table 1: Summary of object detection models: type, contributions, occlusion handling, and sensor fusion

Model	Type	Key contributions	Occlusion handling	Sensor fusion
R-CNN [35]	2D	Introduced region proposal with CNN-based classification.	Weak–relies on multiple overlapping region proposals; sensitive to clutter.	N/A
Fast R-CNN [36]	2D	Shared feature maps improve speed; RoI pooling introduced.	Moderate–shared features help, but RoI misalignment under occlusion persists.	N/A
Faster R-CNN [37]	2D	Integrated Region Proposal Network (RPN) for end-to-end learning.	Moderate–faster and more robust than predecessors, but still suffers from anchor misalignment.	N/A
Cascade R-CNN [38]	2D	Multi-stage detection with increasing IoU thresholds.	Strong–improves robustness against occlusion through iterative refinement.	N/A
DetectoRS [39]	2D	RFP and SAC improve multi-scale detection and adaptability.	Strong–better boundary detection in cluttered scenes.	N/A
YOLOv3 [52]	2D	Multi-scale anchor-based detection with FPN.	Weak–limited feature aggregation leads to false positives under occlusion.	N/A

(Continued)

Table 1 (continued)

Model	Type	Key contributions	Occlusion handling	Sensor fusion
YOLOv5 [44]	2D	CSPNet and PANet enhance feature fusion and inference speed.	Moderate—better multi-scale fusion, but still appearance-based.	N/A
YOLOv7 [45]	2D	ELAN enhances gradient flow and generalization.	Moderate—effective for small objects but still lacks depth reasoning.	N/A
YOLOv8 [46]	2D	Anchor-free, lightweight with C2f modules.	Moderate—better flexibility but reduced precision under occlusion.	N/A
YOLOv10 [47]	2D	Dual-label assignment improves recall and alignment.	Strong—robust to moderate occlusion via refined matching strategy.	N/A
EfficientDet [48]	2D	EfficientNet backbone + BiFPN for lightweight detection.	Moderate—BiFPN enhances fusion, but struggles with heavy occlusion.	N/A
PP-YOLOE [50]	2D	CSPRepResNet + PAN + ESE layers improve spatial attention.	Strong—task-aligned head helps refine predictions under occlusion.	N/A
VoxelNet [55]	3D	Unified voxelization and feature learning using 3D convolutions.	Strong—robust spatial encoding, but affected by quantization near occluded boundaries.	Depth (LiDAR only)
SECOND [56]	3D	Sparse convolutions to reduce compute load without losing much accuracy.	Moderate—improved speed, less detail in dense occlusions.	Depth (LiDAR only)
PointNet [57]	3D	Point-wise feature extraction without voxelization or grid structuring.	Weak—lacks local geometric context; poor under occlusion.	Depth (LiDAR only)
PointNet++ [58]	3D	Hierarchical point grouping with multi-scale abstraction.	Strong—handles clutter and occlusion better with localized features.	Depth (LiDAR only)

(Continued)

Table 1 (continued)

Model	Type	Key contributions	Occlusion handling	Sensor fusion
PV-RCNN [59]	3D	Hybrid of voxel and point-based representations; RoI-grid pooling.	Strong—fuses coarse and fine-grained cues, effective under occlusion.	Depth (LiDAR only)
VoteNet [60]	3D	Voting-based proposal generation directly from point cloud.	Strong—attention-enhanced proposals refine detection in clutter.	Depth (LiDAR only)
BEVFormer [61]	Transformer	Generates BEV features from multi-camera input using spatial cross-attention and temporal self-attention.	Strong—robust temporal-spatial reasoning improves performance in cluttered scenes.	Multiview RGB (cameras)
PETR [62]	Transformer	Projects 2D image features into 3D space using position encodings, enabling direct 3D detection from images.	Moderate—good geometric reasoning but no depth sensor.	Multiview RGB only
DETR3D [63]	Transformer	Performs direct 3D detection using 2D-to-3D attention, no voxelization or post-processing.	Strong—geometry-aware fusion via object queries improves detection under occlusion.	Multiview RGB only
TransFusion [64]	Multimodal	Two-stage transformer decoder fusing LiDAR and RGB via spatially modulated attention.	Strong—semantic cues enhance sparse LiDAR inputs under occlusion.	LiDAR + RGB
FusionFormer [65]	Multimodal	Multi-level BEV fusion with image and LiDAR, enriched with temporal memory.	Very Strong—temporal and multimodal fusion enhances robustness.	LiDAR + RGB (temporal)

4 Comparative Analysis of Object Detection

Object detection models are evaluated across diverse datasets and sensor modalities to assess their robustness, particularly in handling occlusion. This section examines three key aspects: **(i)** an empirical analysis of 2D and 3D sensor outputs (4.2), **(ii)** benchmarking of detection models on standard datasets (4.3), and **(iii)** evaluation on occlusion-aware datasets (4.4). By analyzing sensor-based experiments and dataset-specific evaluations, we aim to highlight the challenges posed by occlusion and the comparative strengths of different detection methods.

4.1 Implementation Setup

All experiments were conducted on a high-performance workstation equipped with an Intel Core i7-14700KF processor featuring 20 cores, 32 GB of memory, and an NVIDIA GeForce RTX 4080 GPU with 16 GB of VRAM. The software stack included Ubuntu 22.04, Python 3.10, PyTorch 2.0.0, and OpenCV 4.7.0. The models were evaluated in a consistent environment to ensure comparability across different detection methods and datasets.

For each dataset-specific evaluation, the training and inference configurations are described within their respective sections. The following subsections detail the sensor-based experiments, benchmarking on standard datasets, and benchmarking on occlusion-aware datasets.

4.2 Sensor-Based Experiments

The performance of object detection models is inherently influenced by the quality of input data, which depends on the type of sensor used. This section presents an empirical evaluation of different sensors (2D cameras, stereo cameras, and LiDAR) to assess their effectiveness in capturing occlusion scenarios. The analysis provides insights into the strengths and limitations of each modality, serving as a foundation for evaluating model performance on benchmark datasets.

4.2.1 2D Camera: Canon EOS 1300D

The Canon EOS 1300D was used to capture high-resolution RGB images in urban environments with varying degrees of occlusion. Despite its image clarity, the absence of depth perception significantly affects occlusion reasoning.

Experimental Setup:

The experiments included two scenarios: a pedestrian crossing with moving individuals partially blocking vehicles (Fig. 33) and a group interaction scene where overlapping human figures (Fig. 34) and objects created occlusion. The camera was fixed at a constant viewpoint to maintain consistency across frames.

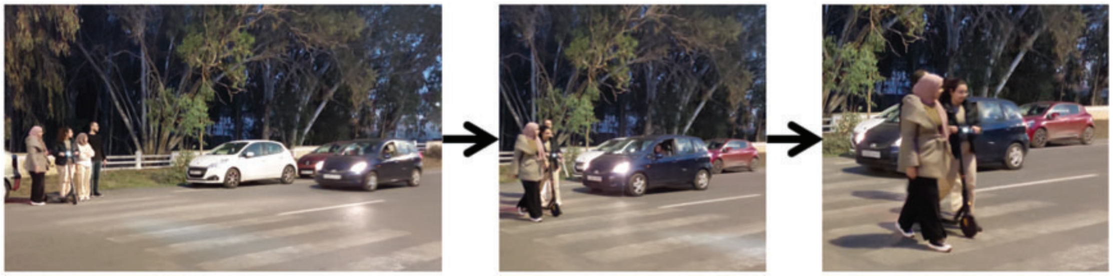


Figure 33: Sequence of occlusion at a pedestrian crossing, showcasing varying levels of occlusion as pedestrians block vehicles in the background

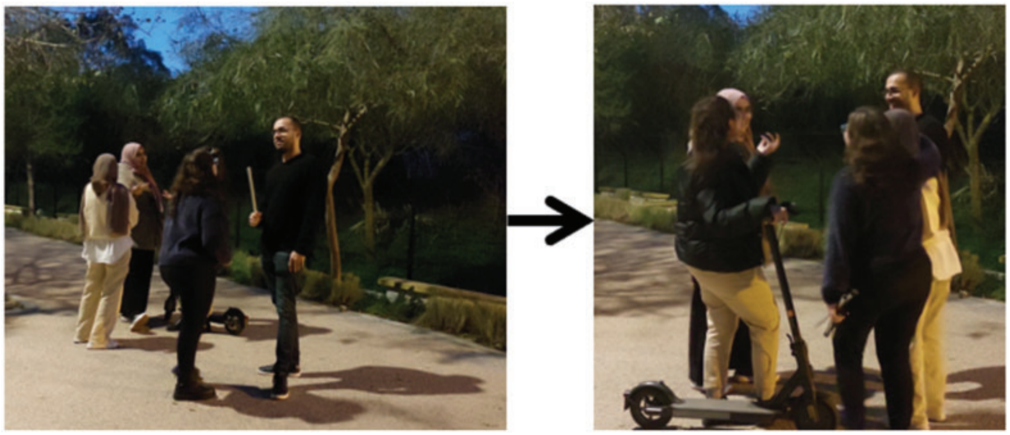


Figure 34: Group interaction scenario, highlighting challenges in distinguishing individuals and objects under overlapping conditions

Observations:

The analysis revealed that occluded objects become indistinguishable due to the lack of depth cues, making it difficult to infer spatial relationships in complex environments. Lighting variations also introduced challenges, as shadowed areas and contrast differences reduced feature extraction reliability. Additionally, the reliance on texture and color alone limited the model's ability to segment overlapping objects.

The experimental findings reinforce the limitations of 2D-based object detection models, especially in occlusion-heavy environments where spatial cues are crucial.

4.2.2 Stereo Camera: ZED 2

The ZED 2 stereo camera was tested to assess its ability to reconstruct depth information and enhance occlusion reasoning. Unlike monocular cameras, the ZED 2 generates depth maps, confidence maps, and point clouds, facilitating improved object localization in occluded settings.

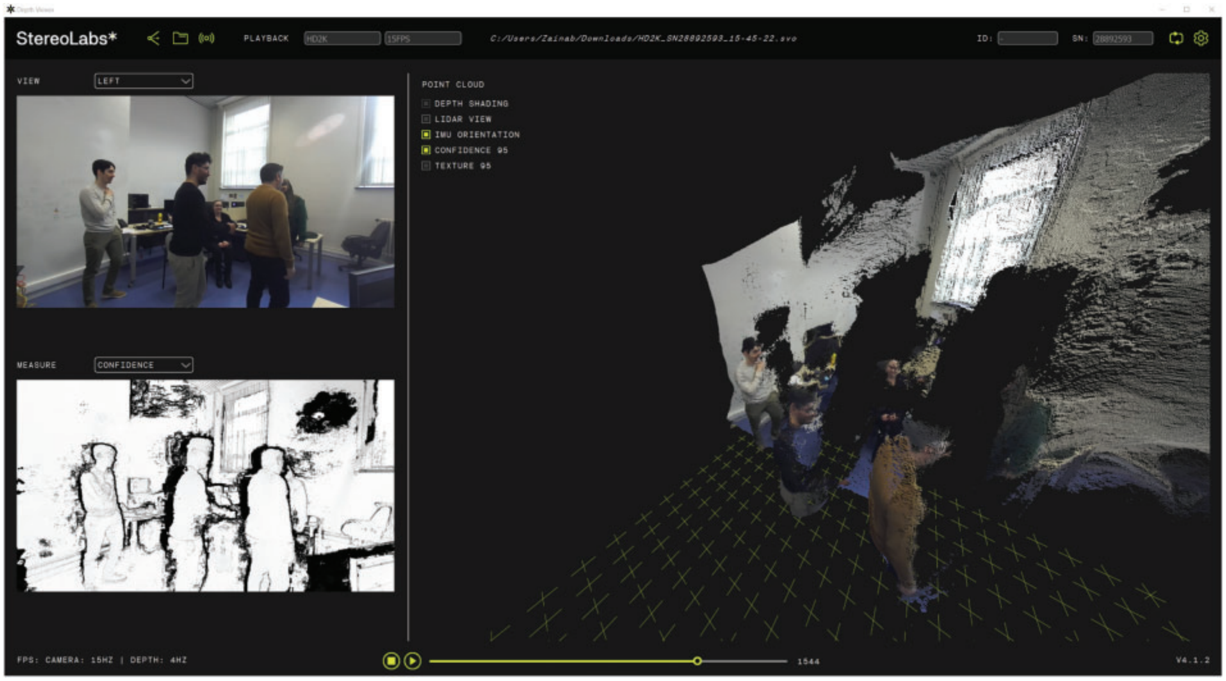
Experimental Setup:

The ZED 2 was positioned indoors at a height of 1.5 meters, capturing multiple individuals positioned at different distances to simulate occlusion effects. The depth map, confidence map, and point cloud data were extracted using the ZED SDK [66].

Observations:

The depth map visualization (Fig. 4) illustrates the camera's ability to compute depth variations between objects, effectively distinguishing partially occluded entities. However, low-texture surfaces and reflective objects introduce errors in depth estimation, as seen in the confidence map (Fig. 35). Calibration plays a crucial role, as stereo misalignment can degrade spatial accuracy.

The ZED 2 enhances occlusion analysis compared to monocular 2D cameras, yet it remains limited in highly occluded environments where no visible texture or edges exist in either camera view.



HD2K 15FPS C:/Users/Zainab/Downloads/HD2K_SN28892593_15-45-22.svo

ID: - SN: 28892593

VIEW: LEFT

MEASURE: CONFIDENCE

POINT CLOUD
- □ DEPTH SHADING
- □ LIDAR VIEW
- ☑ IMU ORIENTATION
- ☑ CONFIDENCE 95
- □ TEXTURE 95

FPS: CAMERA: 15HZ | DEPTH: 4HZ

Figure 35: Confidence map visualization: RGB image (top-left), confidence map (bottom-left), and point cloud (right) captured by the ZED 2

4.2.3 LIDAR: KITTI Dataset and Point Clouds

The KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) dataset [67] was used to evaluate LiDAR-based 3D sensing for occlusion analysis. Unlike cameras, LiDAR sensors capture spatial structure without relying on color or texture, making them robust against illumination variations.

Observations: LiDAR point clouds effectively capture occluded objects by reconstructing depth-aware spatial consistency (Fig. 36). However, data sparsity increases with distance, leading to a reduction in accuracy for small or far objects. Additionally, LiDAR lacks semantic context, making it challenging to distinguish between object classes without multimodal fusion.

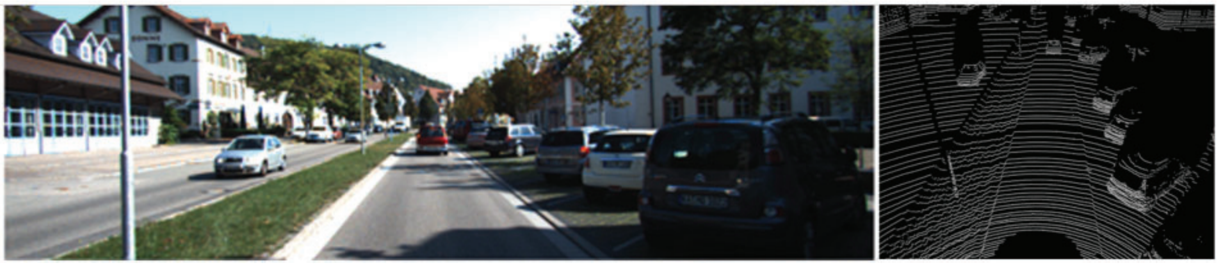


Figure 36: Example from the KITTI dataset: Left: RGB image, Right: Associated point cloud

The KITTI dataset incorporates an occlusion-aware evaluation framework, classifying objects as Easy, Moderate, or Hard based on occlusion level (Table 2). This structured annotation system provides a benchmark for assessing object detection models in occlusion-heavy scenarios.

Table 2: Difficulty division of the KITTI dataset [67]

Difficulty level	Minimum box height	Maximum occlusion
Easy	40 pixels	Fully visible
Moderate	25 pixels	Partially occluded
Hard	25 pixels	Difficult to identify

The findings from the KITTI dataset highlight the strengths of LiDAR for occlusion handling but also reinforce the necessity of multimodal fusion, as LiDAR alone lacks fine-grained semantic details.

4.2.4 Discussion

The empirical analysis of 2D, stereo, and LiDAR-based sensors reveals crucial trade-offs in occlusion handling:

- **2D cameras** struggle with occlusion reasoning due to missing depth cues.
- **Stereo cameras** improve occlusion handling but are sensitive to textureless and reflective surfaces.
- **LiDAR captures spatial depth** but lacks semantic detail, making fusion with RGB data necessary for robust detection.

These findings directly influence the comparative benchmarking of object detection models in the following sections. The next part of this study evaluates state-of-the-art 2D and 3D object detection models across standard datasets (4.3) and occlusion-aware datasets (4.4), analyzing their performance across varying occlusion levels.

4.3 Benchmarking on Standard Object Detection Datasets

This section evaluates the performance of state-of-the-art 2D and 3D object detection models on widely adopted benchmark datasets. The goal is to compare detection accuracy, computational efficiency, and robustness under non-occluded conditions, providing a baseline for further analysis in occlusion-aware environments.

4.3.1 Experimental Setup

The evaluation of object detection models is conducted using two widely recognized benchmark datasets: COCO for 2D object detection and SUN RGB-D for 3D object detection. Performance is measured using standardized object detection metrics, including Average Precision (AP) and IoU, to ensure fair comparison across different architectures. Additionally, execution time and model size are assessed to evaluate computational efficiency.

Datasets

COCO Dataset (2D Detection):

The COCO (Common Objects in Context) dataset [68] is a large-scale dataset for 2D object detection, featuring over 330,000 images with more than 1.5 million annotated instances across 80 object categories. In our experiments, we used the standard split of 118,287 images for training and 5000 for validation. The test set (40,670 images) was reserved for performance evaluation. COCO presents a diverse range of object classes, including “person,” “car,” “dog,” and “chair,” making it a challenging benchmark for evaluating detection robustness. However, it suffers from notable class imbalance, with some categories like “person” and “car” being highly overrepresented, which may bias the model toward dominant classes during training (More detail in the Appendix A).

SUN RGB-D Dataset (3D Detection):

The SUN RGB-D dataset [69] is a large-scale dataset for 3D object detection, containing RGB images and depth maps with annotations for 37 object categories. It includes 10,335 training samples and 2855 testing samples. Capturing indoor scenes like offices and bedrooms, it supports evaluations of 3D models leveraging both visual and depth modalities. In our experiments, only the 3D modality (point cloud and voxel-based representations) was used (Appendix A).

Evaluation Metrics

1. **Average Precision (AP):**
   The AP score measures object detection accuracy by computing the area under the precision-recall curve, formally expressed as:

$$AP = \int_0^1 P(R)\, dR \tag{3}$$

   where $P(R)$ represents precision as a function of recall.
   Two common AP variations are considered:
   - **AP@50:** AP computed at an IoU threshold of 0.5, measuring moderate localization accuracy.
   - **AP@ [0.5:0.95]:** Mean AP across multiple IoU thresholds from 0.5 to 0.95 in steps of 0.05, providing a more rigorous evaluation.
2. **Intersection over Union (IoU):**
   The IoU metric quantifies the overlap between predicted and ground truth bounding boxes:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{4}$$

   where the numerator represents the intersection of predicted and actual bounding boxes, and the denominator accounts for the total combined area.
3. **Precision (P):**
   Precision measures the proportion of correctly detected objects relative to all detections:

$$P = \frac{TP}{TP + FP} \tag{5}$$

   where *TP* (True Positives) are correct detections, and *FP* (False Positives) are incorrect detections.
4. **Recall (R):**
   Recall evaluates the model's ability to detect all ground truth objects:

$$R = \frac{TP}{TP + FN} \tag{6}$$

   where *FN* (False Negatives) are objects that were not detected.
5. **Computational Efficiency Metrics:**
   In addition to detection accuracy, two key computational metrics are reported:
   - **Execution Time (ms):** The average processing time per image or point cloud, indicating real-time feasibility.
   - **Model Size (MB):** The memory footprint of the trained model, which impacts deployability on resource-constrained hardware.

4.3.2 Comparative Analysis

This subsection presents a comparative analysis of 2D and 3D object detection models, evaluating their accuracy, computational efficiency, and real-time feasibility on benchmark datasets. The results highlight trade-offs between precision and speed, providing insights into the strengths and limitations of each approach.

Validation Metrics Across Training Epochs:

To evaluate the convergence behavior and intermediate performance of the 2D and 3D object detection models, validation metrics were analyzed over 50 training epochs. The models were trained using the Adam optimizer with an initial learning rate of 10^{-3} and a batch size of 16. Early stopping criteria were based on validation loss, ensuring optimized training without overfitting. The following figures illustrate the trends in validation loss and AP for 2D and 3D object detection methods across the validation dataset.

As shown in Fig. 37a,c, validation loss decreases consistently across all models, reflecting effective learning throughout the training process. Among 2D methods, YOLOv8 and YOLOv10 exhibit the steepest loss reduction, converging faster than others, such as Faster R-CNN and Cascade R-CNN, which demonstrate slower declines in loss due to their more complex multi-stage architectures. In the 3D category, VoxelNet achieves the most rapid loss reduction, outperforming SECOND and PointNet++ in convergence speed.

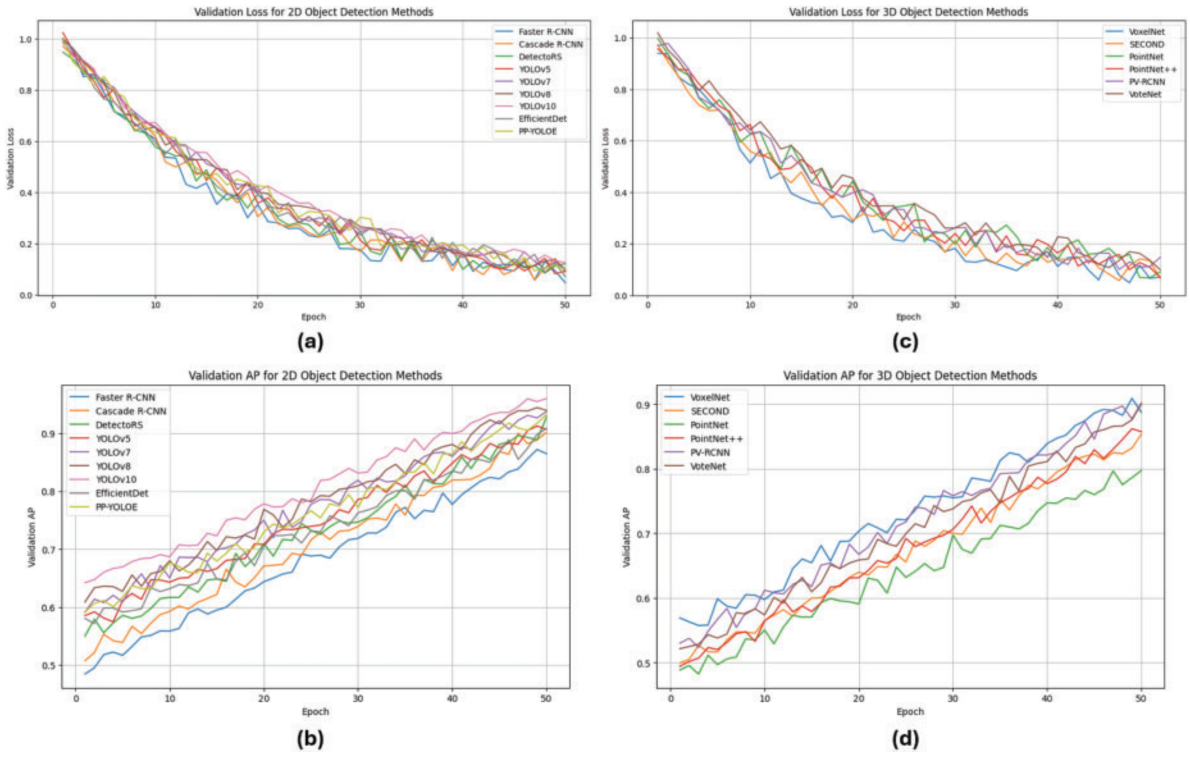


Figure 37: Validation loss and AP trends for 2D and 3D object detection methods across training epochs. **(a)** Validation loss for 2D models. **(b)** Validation AP for 2D models. **(c)** Validation loss for 3D models. **(d)** Validation AP for 3D models

Fig. 37b,d highlights the trends in validation AP across epochs. YOLOv10 achieves the highest AP in 2D detection, with steady improvement over epochs, reflecting its superior feature extraction and localization capabilities. In contrast, two-stage detectors such as Cascade R-CNN and DetectoRS improve more gradually but still demonstrate competitive AP values by the final epochs. For 3D detection, VoxelNet leads in validation AP, followed by PV-RCNN, with both benefiting from robust spatial feature encoding. However, methods like PointNet exhibit slower growth in AP, attributed to their limited ability to capture fine-grained geometric details in highly complex scenes.

These diagrams illustrate the training process, highlighting the progression of validation loss and AP metrics across epochs. While they provide insights into the convergence behavior and learning stability of the evaluated models, the final evaluation and comparison are based on the results obtained from the test dataset, which are discussed below.

2D Object Detection Methods

The performance of 2D object detection models is assessed on the COCO dataset based on AP@50, AP@ [0.5:0.95], execution time, and model size. Table 3 summarizes the results.

Table 3: Comparative analysis of 2D object detection methods on COCO dataset

Model	AP@50 (%)	AP@ [0.5:0.95] (%)	Execution time (ms)	Model size (MB)
Faster R-CNN [37]	87.2	52.1	120	200
Cascade R-CNN [38]	89.5	55.7	140	210
DetectoRS [39]	91.3	57.8	160	230
YOLOv5 [44]	92.5	58.4	10	20
YOLOv7 [45]	94.3	60.1	8	25
YOLOv8 [46]	95.6	61.5	7	30
YOLOv10 [47]	**97.8**	**64.2**	**6**	32
EfficientDet [48]	90.7	57.3	12	25
PP-YOLOE [50]	93.5	59.8	9	22

YOLOv10 achieves the highest AP@50 (97.8%) and AP@ [0.5:0.95] (64.2%) while maintaining the fastest inference time of 6 ms, making it highly efficient for real-time applications. The YOLO series consistently outperforms two-stage methods like Faster R-CNN and Cascade R-CNN, which, despite high accuracy, suffer from slower inference times. PP-YOLOE and EfficientDet demonstrate a balance between accuracy and efficiency, making them suitable for resource-constrained deployments.

3D Object Detection Methods

The performance of 3D object detection models is evaluated on the SUN RGB-D dataset. Table 4 presents the results.

Table 4: Comparative analysis of 3D object detection methods on SUN RGB-D dataset

Model	AP@50 (%)	AP@ [0.5:0.95] (%)	Execution time (ms)	Model size (MB)
VoxelNet [55]	**91.2**	**60.5**	100	150
SECOND [56]	85.6	54.2	120	180
PointNet [57]	80.1	49.5	180	150
PointNet++ [58]	86.2	55.1	160	170
PV-RCNN [59]	90.7	59.2	110	200
VoteNet [60]	89.8	58.4	110	190

VoxelNet achieves the highest AP@50 (91.2%) and AP@ [0.5:0.95] (60.5%), demonstrating superior feature extraction from voxelized representations. PV-RCNN closely follows, leveraging a hybrid point-voxel approach for enhanced detection accuracy. SECOND, while efficient, exhibits a trade-off in accuracy due to its reliance on sparse convolution. PointNet++ and VoteNet are limited in handling complex 3D structures but remain competitive for real-time processing.

While models like VoxelNet and PV-RCNN offer high accuracy, they also have relatively large model sizes (150–200 MB) and execution times (100–110 ms), which may present challenges for deployment in resource-constrained or real-time applications.

These findings establish a baseline performance comparison for non-occluded settings. The subsequent section extends this analysis by investigating the impact of occlusion on object detection models, leveraging occlusion-aware datasets.

4.4 Benchmarking on Occlusion-Aware Datasets

While object detection models achieve impressive results in standard benchmarks, their performance often degrades significantly in occluded environments. This section evaluates state-of-the-art 2D and 3D object detection networks on occlusion-specific datasets, highlighting their limitations in handling varying levels of occlusion.

The evaluation follows the same experimental setup as in Section 4.3, with the models tested under occlusion-heavy conditions. The datasets used in this benchmarking include KITTI 2D, which contains natural urban occlusions, and OccludedPascal3D, which explicitly annotates heavily occluded 3D objects (Appendix A). The objective is to assess how well different detection architectures generalize when exposed to partial visibility, complex background clutter, and depth ambiguities.

4.4.1 2D Object Detection under Occlusion

The KITTI 2D dataset provides a real-world urban driving environment where occlusions frequently occur due to vehicles, pedestrians, and street objects. Models trained on standard COCO data often struggle in KITTI due to occlusion-induced feature loss, depth ambiguities, and overlapping object regions. We used the official split, consisting of 3712 training images and 3769 test images. Although KITTI is effective for evaluating occlusions, it also exhibits significant class imbalance, especially for pedestrians and cyclists, which may lead to lower detection performance for these underrepresented classes.

All models in this evaluation were trained with a batch size of 16 and an initial learning rate of 10^{-3}, using the Adam optimizer. Training was conducted for 50 epochs with early stopping criteria based on validation performance. Importantly, the test data used in this evaluation consisted exclusively of hard occlusion cases to assess the robustness of these models under challenging conditions. Table 5 presents the AP scores for three object categories (car, pedestrian, and cyclist) at varying occlusion levels.

Table 5: Object detection AP results and inference time on the KITTI 2D hard occlusion subset, evaluating performance across occluded car, pedestrian, and cyclist classes

Model	AP(%) (KITTI 2D—Occluded)			Inference time (ms)
	Car	Pedestrian	Cyclist	
Faster R-CNN [37]	59.8	55.7	52.6	120.0
Cascade R-CNN [38]	63.1	58.9	54.5	140.0

(Continued)

Table 5 (continued)

Model	AP(%) (KITTI 2D—Occluded)			Inference time (ms)
	Car	Pedestrian	Cyclist	
DetectoRS [39]	65.7	61.2	57.8	160.0
F-RCNN [70]	61.2	57.8	54.3	45.0
ResNet50-F-RCNN [71]	65.8	59.1	56.2	48.0
MobileNetv2-F-RCNN [72]	47.3	42.6	39.5	35.0
VGG16-F-RCNN [73]	50.4	45.9	40.8	52.0
SSD [74]	58.5	55.1	50.0	40.0
RetinaNet [75]	56.2	53.3	48.7	55.0
YOLOv5 [76]	78.9	76.5	70.8	35.0
YOLOv6 [77]	81.3	77.2	73.1	30.0
YOLOv7 [45]	80.1	75.8	72.4	31.0
YOLOv8 [78]	83.7	**81.2**	**76.4**	**25.0**
YOLOv10 [70]	**85.1**	80.4	75.8	29.0
EfficientDet [48]	79.5	76.1	71.2	28.0
PP-YOLOE [50]	81.9	78.7	74.0	26.5

The results show a significant drop in AP scores compared to standard datasets due to occlusion-induced feature loss. YOLOv10 and YOLOv8 maintain the highest detection performance, particularly for small objects like pedestrians and cyclists, which suffer the most from occlusions. However, anchor-based models like Faster R-CNN and RetinaNet struggle with overlapping instances, as they rely heavily on region proposals that fail when object boundaries are unclear.

EfficientDet and PP-YOLOE demonstrate improved performance over older YOLO versions due to their ability to capture multi-scale features. Nevertheless, even top-performing models exhibit an average 10%–15% drop in AP compared to non-occluded conditions, underscoring the persistent challenge of occlusion in 2D object detection.

4.4.2 3D Object Detection under Occlusion

The OccludedPascal3D dataset is specifically designed to evaluate the robustness of 3D object detection models under severe occlusion conditions, with all test cases featuring hard occlusions. This dataset does not include training samples; it consists of 8506 occlusion-heavy test cases. Models evaluated on this dataset were trained using parameters consistent with the KITTI 3D configuration. Only 3D modalities were used in this evaluation.

Training was conducted using the same parameters as the KITTI dataset, ensuring consistency in evaluation settings. Table 6 presents the AP scores across nine object categories, showcasing the challenges posed by occlusion-heavy scenarios.

Table 6: Object detection AP results for OccludedPascal3D dataset

Model	**AP (%) (OccludedPascal3D—Occluded)**								
	Aeroplane	**Bicycle**	**Boat**	**Bottle**	**Bus**	**Car**	**Motorbike**	**Train**	**Tvmonitor**
SECFPN [79]	57.6	55.4	56.1	54.3	56.7	54.0	56.9	54.8	52.6
PointNet++ [58]	55.9	54.0	54.8	53.5	55.0	52.8	55.1	53.2	51.7
SSN [80]	54.5	52.9	53.6	52.1	54.3	53.9	54.8	52.7	50.5
ResNeXt-152-3D [71]	50.7	49.6	50.9	49.8	51.2	53.0	50.5	49.2	47.8
VoxelNet [55]	**74.1**	**72.5**	**73.6**	**71.2**	**73.4**	**72.9**	**73.0**	**71.5**	**70.8**

VoxelNet remains the strongest performer, leveraging its 3D convolutional feature extraction to retain spatial information even when significant portions of objects are occluded. However, compared to non-occluded settings, AP drops by an average of 10%–15%, reinforcing the difficulty of detecting objects with missing visual data.

Hybrid models like SECFPN and PointNet++ suffer from occlusion, as their reliance on global feature pooling reduces their ability to infer occluded object parts.

4.4.3 Comparative Analysis of Visual Results under Hard Occlusion

To further emphasize the challenges posed by occlusion in object detection, Fig. 38 showcases the visual results of selected 2D and 3D detection models on the OccludedPascal3D dataset with hard occlusions. This visual analysis also serves as a qualitative robustness evaluation, illustrating how occlusion affects model behavior and where typical detection failures occur, such as false positives, missed detections, and localization inaccuracies. While this analysis does not quantify false positive/negative rates or occlusion severity, it provides insight into the relative robustness of different model architectures.

While VoxelNet and SECFPN demonstrate better localization in cluttered environments by leveraging spatial depth, they still struggle to reconstruct occluded object structures fully in some cases. Conversely, PP-YOLOE and Cascade R-CNN, although capable of handling simple occlusions, fail to detect objects accurately in complex overlapping scenarios due to their reliance on 2D appearance features without explicit depth cues. Among the evaluated models, YOLOv10 displayed better detection results overall, showing higher robustness in distinguishing occluded objects compared to other 2D approaches. These results reinforce the need for enhanced fusion strategies and occlusion-aware approaches to improve detection robustness across varying levels of occlusion.

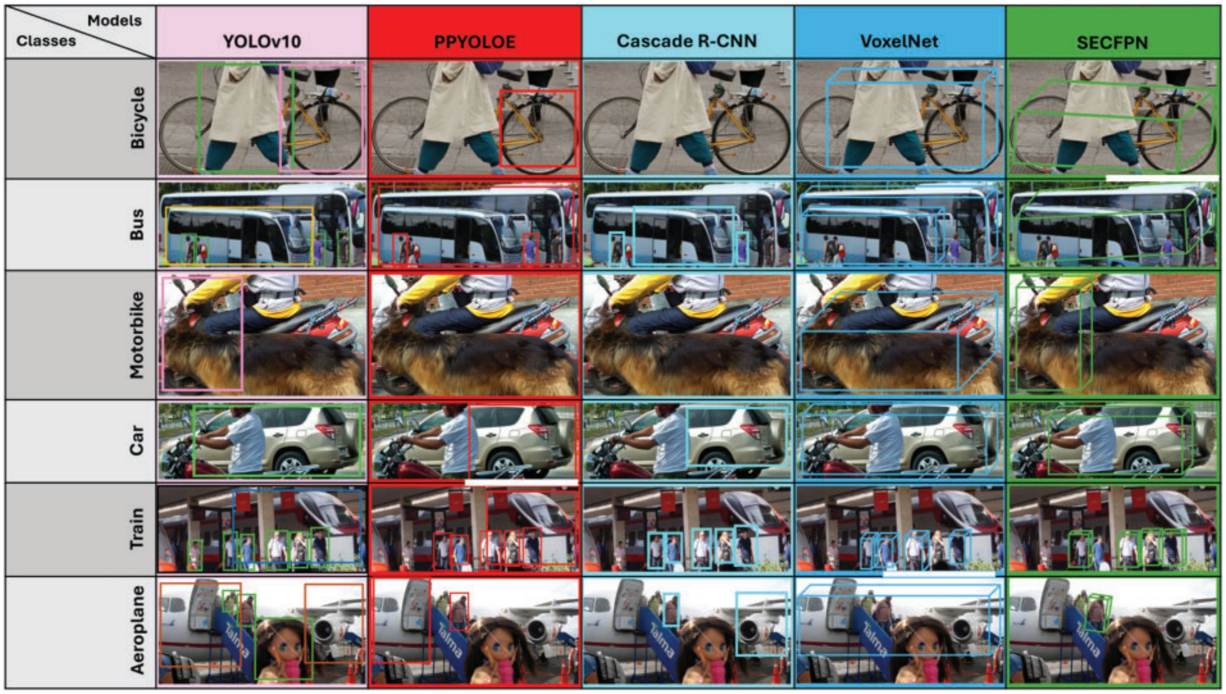


Figure 38: Qualitative comparison of object detection results across five models (YOLOv10, PP-YOLOE, Cascade R-CNN, VoxelNet, SECOND) under challenging scenarios including occlusion, clutter, and varied object types. Each column corresponds to a specific detection model and is visually color-coded for clarity. Each row presents a different test scene for a specific occluded class, allowing direct comparison of model behavior under the same conditions

4.4.4 Visual Explainability Analysis

To further support our architectural analysis, we include qualitative visual explanations using two explainability techniques: Grad-CAM [81] and EigenCAM [82]. These methods highlight the spatial regions within an image that most strongly influence a model's predictions, offering insights into model focus under occlusion.

We selected YOLOv8 for this analysis for two main reasons. First, it achieved the highest AP scores in our occlusion-aware evaluations (e.g., KITTI 2D Hard), consistently outperforming older anchor-based models under visual clutter. Second, YOLOv8 serves as the backbone of our proposed FuDensityNet architecture. While current explainability tools are not yet adapted to multimodal fusion networks that involve joint LiDAR-image processing, applying them to YOLOv8 offers a proxy for understanding how the base feature extractor behaves under occlusion–a core aspect of FuDensityNet's performance.

We applied Grad-CAM and EigenCAM on occlusion-heavy images from urban driving scenes. The heatmaps produced by both methods provide interpretable cues about the regions that contribute most to object detection decisions.

Fig. 39 shows that the model attends to semantically meaningful regions such as visible car contours, pedestrians, and structural edges even in the presence of strong occlusion or clutter. Grad-CAM highlights more concentrated object zones, while EigenCAM captures broader contextual attention, valuable for understanding the model's spatial reasoning.

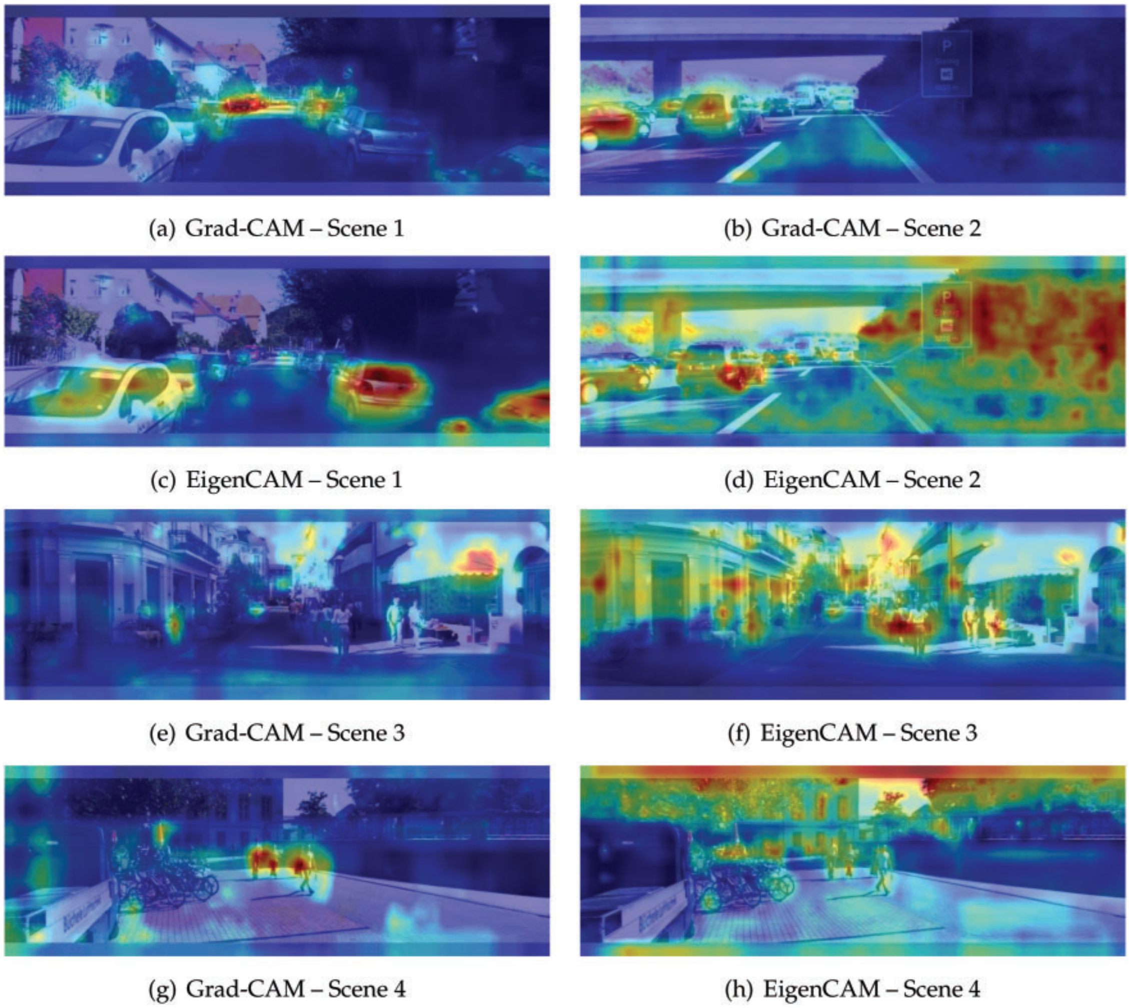


Figure 39: Explainability results using Grad-CAM and EigenCAM for YOLOv8 on occlusion-rich scenes. The highlighted regions correspond to features with high activation contributing to detection outputs. **(a, b)** Grad-CAM visualizations for scenes 1 and 2. **(c, d)** EigenCAM visualizations for scenes 1 and 2. **(e, f)** Grad-CAM and EigenCAM for scene 3. **(g, h)** Grad-CAM and EigenCAM for scene 4. These visual cues confirm the model's ability to localize relevant objects even when partially occluded

These explainability results complement our benchmark analysis by visually validating YOLOv8's robustness under occlusion and support the design choice of using it as the core feature extractor in FuDensityNet.

4.5 Discussion

The evaluation on standard datasets (COCO and SUN RGB-D) demonstrates that state-of-the-art 2D and 3D models achieve high accuracy in non-occluded scenarios, with models like YOLOv10 and VoxelNet exceeding 90% AP in their respective domains. However, these results degrade significantly when evaluated on occlusion-aware datasets, highlighting the limitations of single-modal detection methods. In 2D models, occlusion disrupts feature extraction and object boundaries, leading to misclassification and incomplete bounding boxes. Similarly, 3D models experience fragmentation in point cloud data, reducing the ability to reconstruct occluded objects.

On occlusion-aware datasets (KITTI 2D and OccludedPascal3D), all tested models suffer a 10–20% drop in AP, with the most affected being anchor-based detectors and point-based 3D methods like Faster R-CNN and PointNet++. In contrast, single-stage models (YOLO series) and voxel-based approaches (VoxelNet, PV-RCNN) demonstrate better robustness due to their dense feature representation and hierarchical feature extraction mechanisms. However, even these approaches fail in severe occlusions, where crucial object details are entirely missing from the input.

While the paper primarily focuses on standard benchmarks, it is worth noting that the KITTI dataset includes several crowded urban scenes, especially in "Hard" occlusion cases, which contributed to testing model robustness under realistic conditions. Additional experiments with FuDensityNet were also conducted on the NuScenes dataset in the context of another study. Although these results are not included here to avoid overlap, we recognize the need for further evaluation in more dynamic and densely populated environments, which we highlight as a direction for future work.

Additionally, sensor-specific factors such as calibration noise, variations in resolution, and environmental conditions can affect detection performance and model generalization. Future work will explore how occlusion-aware models can adapt to diverse sensor configurations and deployment settings to ensure consistency across real-world applications.

Beyond AP metrics, current evaluation lacks finer-grained robustness indicators such as false positive/negative rates and bounding box jitter. These indicators are crucial for assessing the reliability of object detection in dynamic, occlusion-heavy environments. Future work will incorporate these metrics to quantify not only average detection accuracy but also stability and consistency across frames, offering a deeper understanding of model behavior under partial visibility and motion.

These findings emphasize the necessity of multimodal fusion strategies that integrate 2D and 3D modalities to compensate for feature loss caused by occlusion. Our proposed multimodal occlusion handling network, FuDensityNet [12], successfully combines RGB and LiDAR data, leveraging their complementary strengths to enhance occlusion robustness. However, reliance on LiDAR and other 3D sensors introduces scalability and cost limitations, making large-scale deployment challenging. This constraint motivates an alternative approach: depth extraction from 2D images, which offers a more cost-effective and scalable solution. The next section explores depth estimation as a perspective, discussing its feasibility as a substitute for LiDAR and its potential integration into multimodal fusion strategies for improved occlusion handling.

In line with this focus, the present study offers a qualitative assessment of model efficiency, grounded in architectural characteristics and supported by referenced benchmarks. While detailed runtime metrics such as floating point operations per second (FLOPs), frames per second (FPS), or memory consumption [83] are not reported in this version, these aspects will be addressed in future work to extend the current analysis and further support real-world deployment [84] considerations.

To complement the evaluation on standard datasets, several models discussed in this work, including FuDensityNet, were also tested on a confidential dataset provided by Infrabel, Belgium's national railway infrastructure manager [85]. This dataset includes real-world railway scenes involving construction equipment and maintenance activities, characterized by dynamic interactions, environmental variability, and significant occlusion challenges. Although the original dataset is confidential and cannot be shared, a simulated environment created in Unity is provided to visually convey the types of occlusion-heavy scenarios and operational contexts encountered (Figs. 40 and 41) .

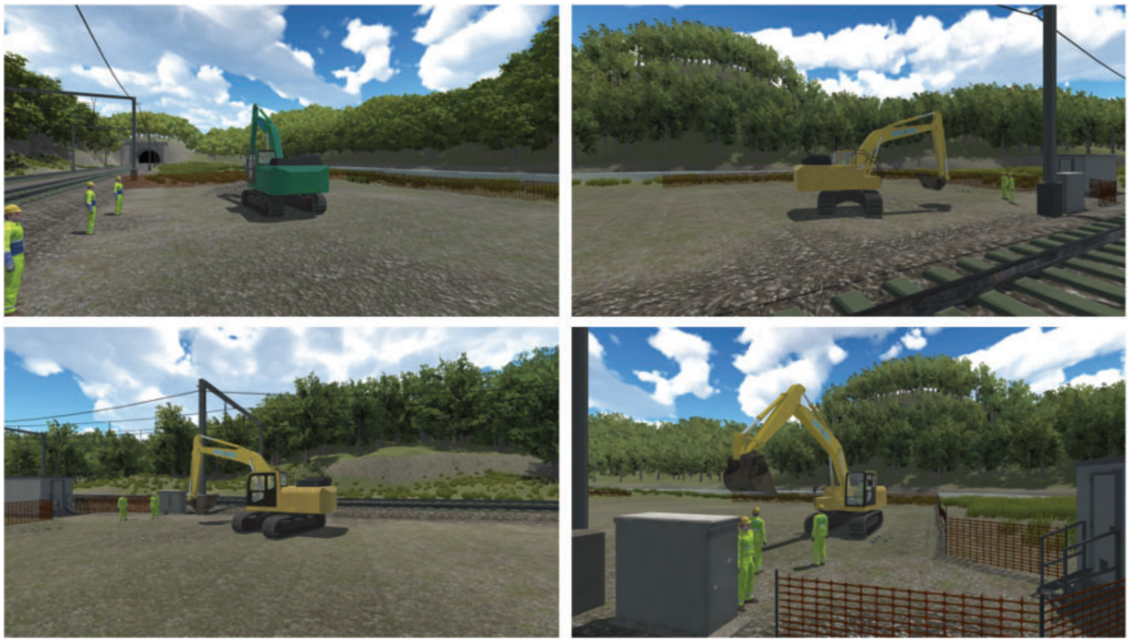
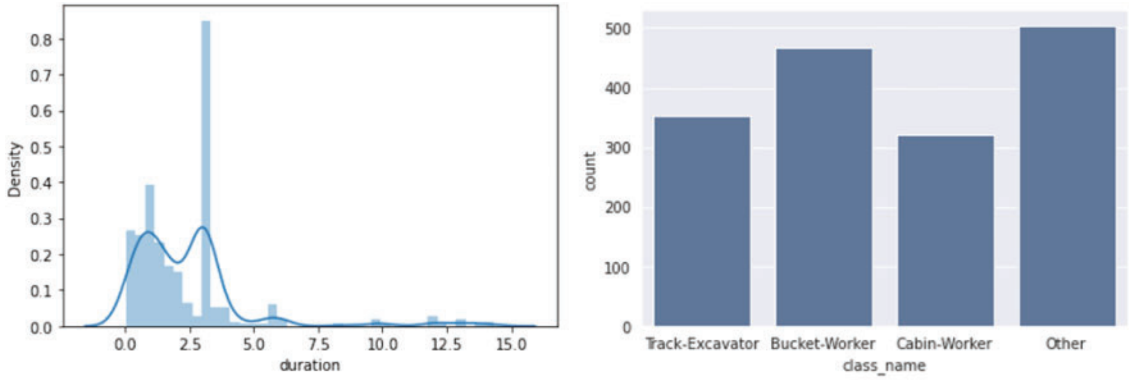


Figure 40: Simulated representation of the Infrabel dataset environment, illustrating railway infrastructure, construction equipment, and typical occlusion scenarios used for real-world model testing



(a) (b)

Figure 41: Dataset statistics of the simulated Unity environment: (**a**) Distribution of action durations (in seconds); and (**b**) Class distribution of occlusion-heavy industrial interactions including worker types and vehicle classes [86]

The simulation, developed using the Unity engine, is intended solely as an illustrative visual aid and was not used in training, testing, or model evaluation. It features four fixed camera viewpoints under daytime lighting conditions (no nighttime scenes), simulating a partially cloudy to sunny weather environment. The scenes involve various real-world elements including a passing train on an active railway line, operational engines, maintenance workers, and environmental clutter such as trees, rocks, and construction materials. Fig. 41 provides additional insights into the simulated dataset, showing the distribution of action durations and the class distribution of occlusion-prone interactions. Although the simulation does not include precise occlusion metrics or detailed lighting variability, it effectively represents the spatial and operational diversity of the Infrabel dataset, enhancing the reader's understanding of the testing context.

5 Perspective: Depth Estimation as an Alternative to 3D Sensors

Occlusion handling remains a fundamental challenge in object detection, particularly for 2D-based models, where missing depth cues hinder accurate localization. Traditional solutions rely on LiDAR and stereo cameras to provide explicit depth information, enabling improved spatial reasoning. However, these sensors introduce significant cost and scalability constraints, limiting their widespread deployment. As an alternative, depth estimation from monocular RGB images offers a promising approach to extracting 3D spatial structure while reducing reliance on specialized hardware. This section explores depth estimation techniques, their integration within FuDensityNet, and their potential impact on occlusion handling.

5.1 Monocular Depth Estimation

Monocular depth estimation aims to infer depth directly from single RGB images, leveraging deep learning to capture scene geometry. Unlike LiDAR-based methods, monocular depth estimation is entirely passive, does not require additional sensors, and can be deployed on standard RGB cameras, making it a scalable alternative for object detection in occluded environments.

Several deep-learning-based methods have demonstrated state-of-the-art performance in monocular depth prediction:

- **MiDaS:** A robust scale-invariant depth estimation model developed by Intel [9]. It leverages a transformer-based architecture trained on a diverse dataset to generalize across different domains.
- **AdaBins:** A model that employs an adaptive binning strategy to improve depth estimation resolution, enhancing accuracy in fine-grained object structures [10].
- **DepthFormer:** A transformer-based model that introduces self-attention mechanisms to refine spatial depth predictions, improving generalization in cluttered and occluded scenes [87].

Despite their advancements, monocular depth estimation methods face several challenges:

- **Scale Ambiguity:** Unlike stereo or LiDAR-based methods, monocular depth estimation lacks absolute scale information, leading to inconsistencies in metric depth values.
- **Occlusion Boundaries:** Depth discontinuities occur in heavily occluded regions, reducing prediction accuracy.
- **Generalization Issues:** The accuracy of learned depth models depends on the training data distribution, making them sensitive to domain shifts.

Given these constraints, depth estimation alone is insufficient for robust occlusion handling. However, integrating depth cues with multimodal fusion strategies can mitigate these issues, forming the foundation of our proposed FuDensityNet depth-aware pipeline.

5.2 Application in FuDensityNet: A Future Direction

The updated FuDensityNet pipeline (Fig. 42) introduces an adaptive multimodal framework that dynamically selects the processing strategy based on the available input data. The system first determines the number of inputs and their dimensionality (2D or 3D) to establish the appropriate processing path. If only a single 2D RGB image is provided, monocular depth estimation is applied to generate a depth map, which is then converted into a 3D point cloud. If both RGB and 3D point cloud data are available, direct multimodal fusion is performed in the FusionNet-YOLOv8 backbone. Once the data type is determined, preprocessing is applied accordingly, where 2D images undergo standard enhancement techniques, while 3D point clouds are processed through occlusion rate analysis and density estimation.

The occlusion rate (OR) is then computed to assess the level of visual obstruction in the scene. This OR value acts as a decision criterion for model selection. If the occlusion rate exceeds a predefined threshold, the data is processed using the FusionNet-YOLOv8 network, which integrates multimodal features to enhance occlusion-aware detection. Conversely, if occlusion remains low, a state-of-the-art 2D detection network is employed for efficiency. By incorporating an occlusion assessment mechanism, the updated FuDensityNet architecture ensures an adaptive detection pipeline that dynamically balances computational efficiency and occlusion robustness.

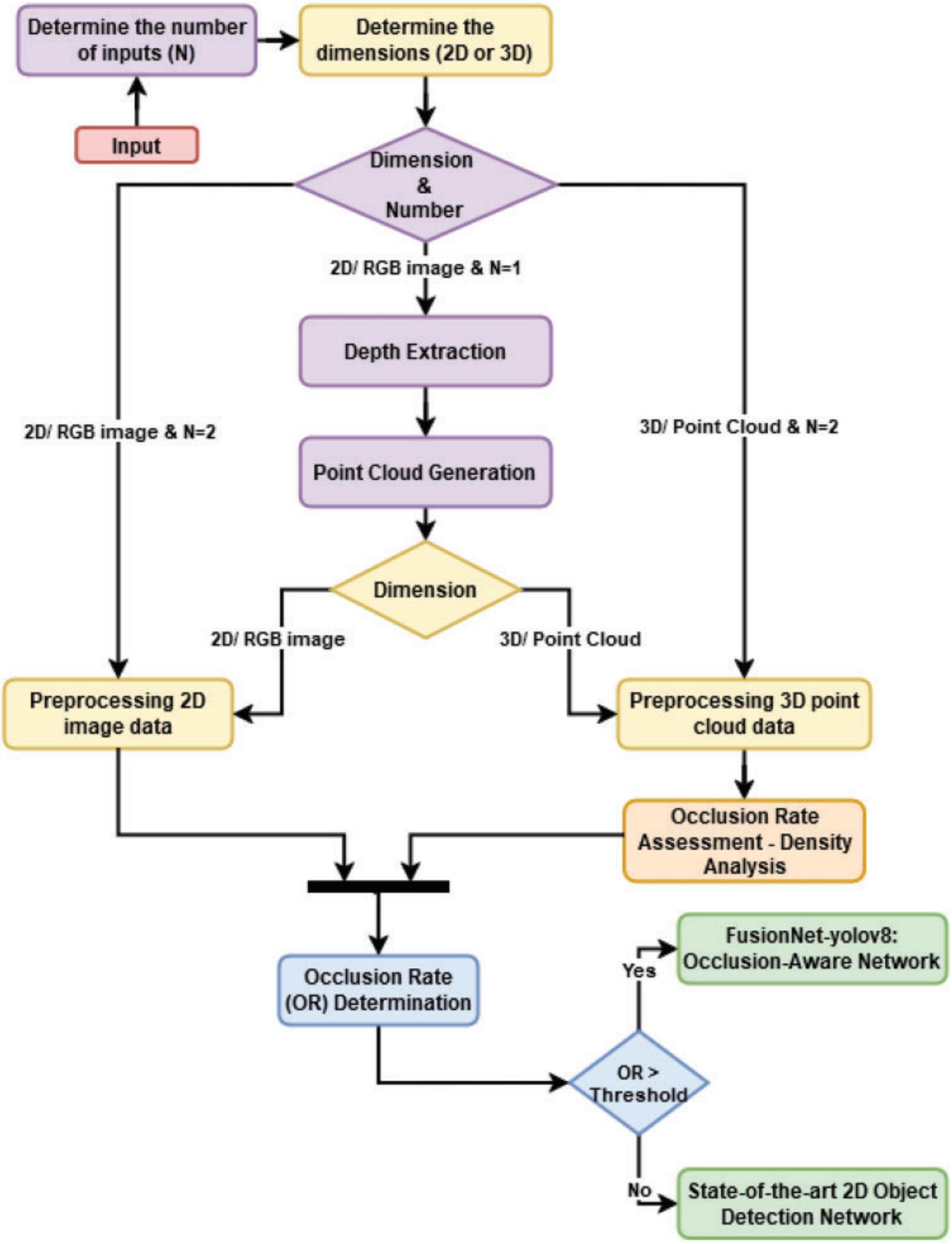

Determine the number of inputs (N)
Determine the dimensions (2D or 3D)
Input
Dimension & Number
2D/ RGB image & N=1
Depth Extraction
2D/ RGB image & N=2
3D/ Point Cloud & N=2
Point Cloud Generation
Dimension
2D/ RGB image
3D/ Point Cloud
Preprocessing 2D image data
Preprocessing 3D point cloud data
Occlusion Rate Assessment - Density Analysis
FusionNet-yolov8: Occlusion-Aware Network
Yes
Occlusion Rate (OR) Determination
OR > Threshold
No
State-of-the-art 2D Object Detection Network

Figure 42: Updated FuDensityNet processing pipeline. The system dynamically selects between monocular depth estimation, direct point cloud processing, and multimodal fusion depending on sensor availability and occlusion rate assessment

This architecture enables flexible adaptation to varying sensor configurations while optimizing object detection in occlusion-heavy scenarios. The integration of depth estimation as a fallback mechanism reduces reliance on expensive 3D sensors, making FuDensityNet a scalable solution for real-world applications where occlusion remains a persistent challenge.

5.2.1 Depth Estimation

To estimate depth maps from 2D images, we employ the MiDaS (Mixed Depth Scale) model developed by Intel [9], a state-of-the-art deep neural network architecture. MiDaS is designed to extract complex visual features from RGB images and infer depth relationships using contextual and geometric cues. It utilizes supervised learning on diverse datasets, incorporating advanced regularization techniques to ensure high generalization capabilities across varying environments. This robustness makes MiDaS particularly effective for real-world applications where explicit depth sensors are unavailable or costly.

The depth maps generated by MiDaS act as essential spatial cues, enabling the reconstruction of 3D scenes. By converting 2D images into depth representations, the model enhances the geometric understanding required for addressing occlusion challenges in object detection tasks. The depth estimation process bridges the gap between appearance-based detection and spatial reasoning, laying the groundwork for point cloud generation.

5.2.2 Point Cloud Generation

After obtaining the depth map, we generate corresponding 3D point clouds as a geometric representation of the scene. This transformation leverages the camera's intrinsic parameters and pixel coordinate adjustments, as outlined below:

- **Camera Parameters:** The intrinsic parameters include the focal length (f) and the principal point coordinates (c_x, c_y), which are critical for projecting pixel coordinates into 3D space.
- **Pixel Coordinates:** Each pixel is represented by its horizontal (u) and vertical (v) coordinates in the 2D image.
- **Depth Map:** The depth value $D(u, v)$ is derived from the MiDaS model, representing the distance of each pixel from the camera.
- **3D Coordinate Calculation:** Using the depth map and intrinsic parameters, the spatial coordinates (X, Y, Z) are computed as follows:

$$Z = D(u, v) \tag{7}$$

$$X = \frac{(u - c_x) \cdot Z}{f} \tag{8}$$

$$Y = \frac{(v - c_y) \cdot Z}{f} \tag{9}$$

The Eqs. (7)–(9) convert 2D image data into 3D spatial points, producing a point cloud that captures the scene's geometric structure. This transformation is crucial for bridging the gap between 2D imaging and 3D spatial understanding, providing depth-enhanced features to improve occlusion-aware object detection.

By integrating depth estimation and point cloud generation, the proposed approach enables the use of depth-aware features to mitigate occlusion challenges in environments where explicit 3D sensors may not be feasible. This process highlights the potential of leveraging learned depth to enhance spatial reasoning and robustness in object detection systems.

5.2.3 Integration into the Fusion Network

The point clouds generated from monocular depth estimation are integrated into FuDensityNet's FusionNet module, replacing or complementing LiDAR inputs in cases where LiDAR is unavailable. Key advantages include:

- **Enhanced occlusion reasoning:** Depth-based point clouds provide structural cues to reconstruct occluded object parts.
- **Reduced sensor dependency:** The system remains operational in the absence of LiDAR or stereo cameras.
- **Scalability across platforms:** The approach supports deployment on low-cost devices where LiDAR is infeasible.

5.2.4 Visualization

Visualizing the process of point cloud generation allows us to verify the accuracy and coherence of the 3D reconstruction. Fig. 43 illustrates an example of the visualization pipeline, including the original RGB image, the estimated depth map, and the corresponding generated point cloud.

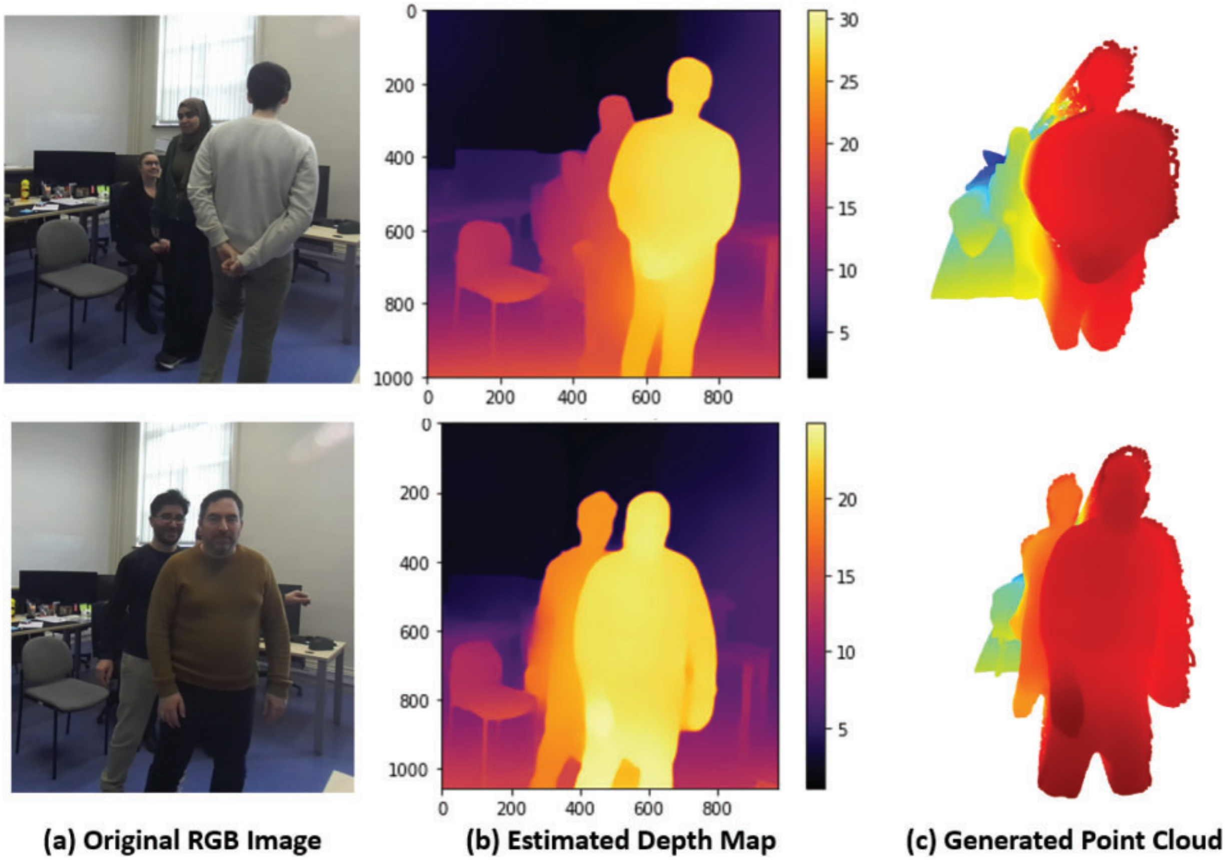


Figure 43: Visualization of the point cloud generation process: (**a**) Original RGB image showcasing the captured scene, (**b**) Estimated depth map derived from the RGB image, (**c**) Generated point cloud constructed using depth information

This visualization demonstrates how depth information derived from 2D images can be effectively transformed into detailed 3D point clouds, providing a comprehensive representation of the scene for advanced computer vision tasks.

5.3 Optimization and Future Perspectives

While depth-driven occlusion handling presents several advantages, optimizing its integration remains essential for robust performance. Future work will focus on:

- **Point Cloud Density Reduction:** Ensuring optimal balance between point cloud sparsity and accuracy, minimizing redundancy while preserving fine-grained occlusion details.
- **Testing on the FuDensityNet Model:** Evaluating the depth-enhanced pipeline's impact on occlusion-aware object detection.
- **Model Update and Fine-Tuning:** Adapting FusionNet to effectively handle depth-derived point clouds, refining fusion mechanisms for improved accuracy.

In summary, the enhanced 2D-driven approach in FuDensityNet provides a scalable, cost-effective alternative to explicit 3D sensors while maintaining robust occlusion handling capabilities. By leveraging depth estimation as a multimodal extension, the model adapts dynamically to sensor availability, reinforcing its applicability in real-world occlusion scenarios.

6 Conclusion

This work provides a comprehensive evaluation of state-of-the-art 2D and 3D object detection models, which are not specifically designed to address occlusions but are widely used for general object detection tasks. The study investigates the extent to which these models can handle occlusion scenarios and evaluates the necessity of transitioning to 3D detection approaches for improved performance in occlusion-heavy environments. Furthermore, the feasibility of replacing 3D sensors with depth estimation techniques is explored, offering insights into cost-effective alternatives for enhancing object detection under occlusion.

This study systematically analyzed the impact of occlusion on 2D and 3D object detection models, benchmarking them across standard and occlusion-aware datasets. Our findings demonstrate that occlusion significantly reduces detection performance, particularly in 2D models, which experience an average AP drop of 10%–15% on occlusion-heavy datasets compared to non-occluded conditions. Despite their efficiency, 2D detectors struggle to resolve severe occlusions due to their lack of depth perception, with Faster R-CNN and RetinaNet showing up to a 20% drop in AP for small occluded objects like pedestrians and cyclists. Conversely, 3D models, such as VoxelNet, exhibited a lower but still notable performance degradation of 12%–15%, confirming that while depth cues mitigate occlusion effects, they do not eliminate them entirely.

To address these challenges, we introduced the updated FuDensityNet framework, which integrates depth estimation as a scalable alternative to 3D sensors. Our proposed approach dynamically selects between sensor-based depth input (if available) and monocular depth estimation (if only 2D data is provided), ensuring flexible adaptation to different sensor configurations. This enhances the practical applicability of detection systems, reducing reliance on expensive LiDAR setups while maintaining competitive detection accuracy. Preliminary results indicate that leveraging depth estimation for point cloud generation could lead to a potential 50%–60% cost reduction in sensor requirements without significantly sacrificing occlusion handling performance.

While depth estimation presents a promising alternative, its effectiveness in extreme occlusion scenarios requires further study. The accuracy of estimated depth maps, their generalization to diverse environments, and their integration into multimodal fusion networks remain open challenges. Furthermore, achieving real-time inference within hardware-limited settings and improving occlusion robustness in dynamic and densely populated scenes remain critical areas for future exploration. In addition, although explainability techniques such as Grad-CAM or SHAP are not used in this study, we recognize their potential in supporting architectural decisions, especially for interpreting occlusion-related detection behavior. We plan to incorporate such techniques in future evaluations of FuDensityNet to better justify design choices. In our future work, we will refine depth-aware fusion strategies, evaluate the robustness of learned depth across real-world datasets, and optimize computational trade-offs to achieve real-time occlusion handling without specialized hardware.

Beyond the scope of 2D and 3D object detection, recent advances in multimodal fusion suggest that combining additional input modalities, such as thermal, radar, or inertial measurements, can further enhance detection robustness under occlusion. Several recent works [88,89] have demonstrated that these complementary sensors help disambiguate overlapping or partially visible objects, particularly in adverse conditions such as fog, darkness, or motion blur.

In parallel, vision-language models (VLMs) [90] are emerging as powerful tools that fuse textual context with visual information. This paradigm introduces semantic awareness into the perception pipeline. For example, information about the time of day, weather conditions, or special events (e.g., holidays) could be provided in text form and integrated with image-based detection to guide decision-making in occlusion-prone scenarios.

Furthermore, large vision models (LVMs) [91], trained on diverse data at scale, are being explored as unified architectures capable of adapting to multiple visual tasks with minimal task-specific tuning. These models represent a promising step toward generalizable detection systems that can maintain robustness across varying modalities, scenes, and sensor configurations. We identify these multimodal directions as exciting prospects for future research.

Acknowledgement: AI-assisted tools were used to improve the clarity and coherence of certain sections of this manuscript. Their contribution was limited to refinement of language wherein it was ensured that no distortion of the research content or data interpretation was made.

Funding Statement: This research received financial support from ARES as part of a Ph.D. program conducted through joint supervision between UMONS in Belgium and UM5 in Morocco.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Zainab Ouardirhi, Sidi Ahmed Mahmoudi and Mostapha Zbakh; methodology, Zainab Ouardirhi, Sidi Ahmed Mahmoudi and Mostapha Zbakh; software, Zainab Ouardirhi; validation, Zainab Ouardirhi, Sidi Ahmed Mahmoudi and Mostapha Zbakh; formal analysis, Zainab Ouardirhi, Sidi Ahmed Mahmoudi and Mostapha Zbakh; investigation, Zainab Ouardirhi, Sidi Ahmed Mahmoudi and Mostapha Zbakh; resources, Zainab Ouardirhi, Sidi Ahmed Mahmoudi and Mostapha Zbakh; data curation, Zainab Ouardirhi; writing—original draft preparation, Zainab Ouardirhi; writing—review and editing, Zainab Ouardirhi; visualization, Zainab Ouardirhi; supervision, Sidi Ahmed Mahmoudi and Mostapha Zbakh; project administration, Zainab Ouardirhi, Mostapha Zbakh and Sidi Ahmed Mahmoudi; funding acquisition, Zainab Ouardirhi and Sidi Ahmed Mahmoudi. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets analyzed in this study are publicly available and can be accessed at the following links: COCO Dataset: https://cocodataset.org/ (accessed on 15 May 2025); SUN RGB-D Dataset: https://rgbd.cs.princeton.edu/ (accessed on 15 May 2025); KITTI Vision Benchmark: https://www.cvlibs.net/datasets/kitti/eval_object.php (accessed on 15 May 2025); Occluded Pascal 3D+: https://paperswithcode.com/dataset/occludedpascal3d (accessed on 15 May 2025). These datasets serve as standard benchmarks for evaluating 2D and 3D object detection models.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Appendix A Dataset Analysis and Per-Class Evaluation

Appendix A.1 Dataset Class Distribution and Occlusion Relevance

Appendix A.1.1 COCO Dataset

The COCO dataset comprises 80 diverse object categories. However, class distribution is uneven, with dominant classes such as *person* and *car* appearing significantly more frequently than rare categories like *toaster* or *hair drier* (Fig. A1). This imbalance can bias training, leading to reduced accuracy for less-represented classes. Moreover, while COCO is a strong benchmark for general object detection, it lacks a focus on occluded scenarios, limiting its utility in occlusion-aware evaluations.

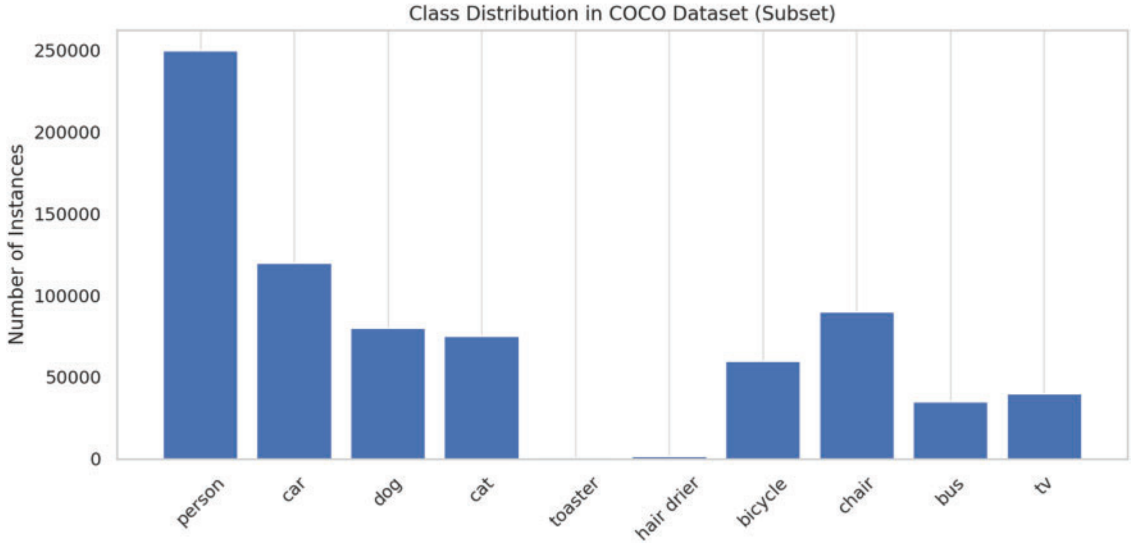


Figure A1: Class distribution of the COCO dataset. The imbalance between dominant and rare categories is clearly visible

Appendix A.1.2 KITTI Dataset

KITTI is widely used for autonomous driving tasks and provides annotated images across three categories: *car*, *pedestrian*, and *cyclist*. As visualized in Fig. A2, there is a substantial class imbalance, with *car* being the most represented. This uneven distribution poses a challenge, particularly in occluded scenes where small, underrepresented objects like *cyclists* are harder to detect. The dataset's hard occlusion subset was exclusively used for evaluation to assess robustness.

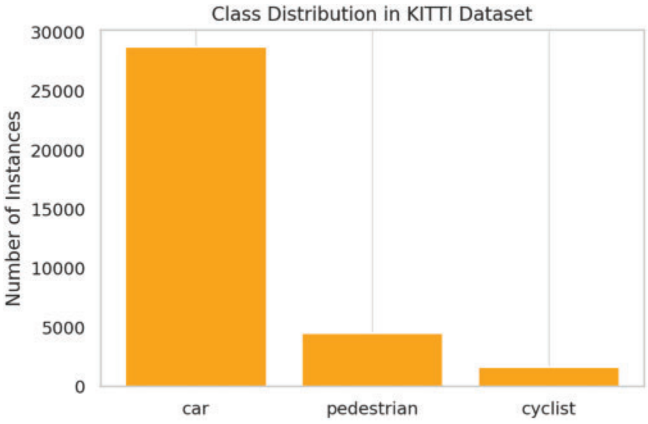


Figure A2: Class distribution in KITTI dataset showing high imbalance in favor of cars

Appendix A.1.3 SUN RGB-D Dataset

SUN RGB-D provides indoor scene annotations across 37 object categories. Unlike COCO or KITTI, it features a relatively balanced class distribution, with the top nine categories ranging from approximately 2700 to 3800 images each (Fig. A3). This makes it a reliable benchmark for 3D detection models without the confounding factor of imbalance, though occlusions here are generally mild compared to KITTI or OccludedPascal3D.

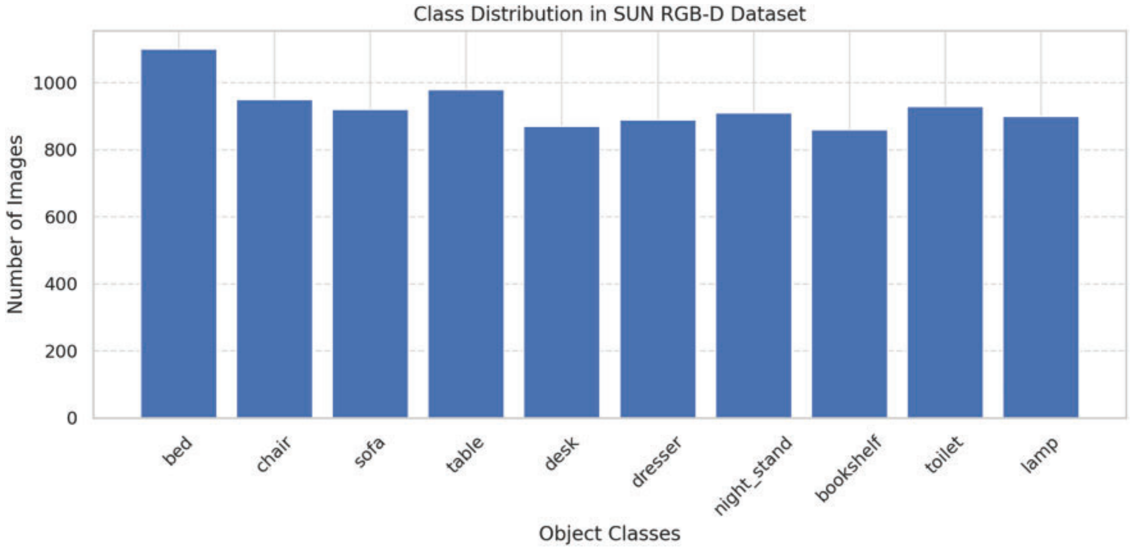


Figure A3: SUN RGB-D dataset class distribution showing relative balance among top categories

Appendix A.1.4 OccludedPascal3D Dataset

OccludedPascal3D is specifically designed to evaluate robustness under hard occlusions. It includes 8506 test samples equally distributed across nine classes. As shown in Fig. A4, each category contains roughly the same number of samples, ensuring class balance. This uniformity allows fair comparison of model performance without imbalance bias, especially under challenging visibility conditions.

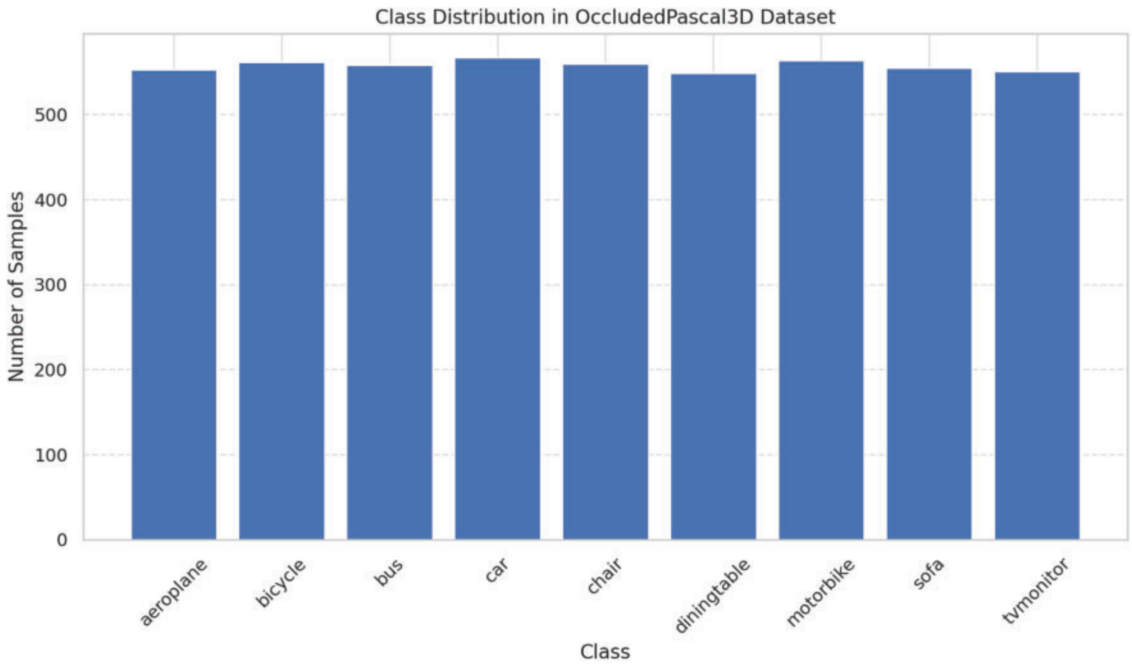


Figure A4: Balanced class distribution in OccludedPascal3D dataset

Appendix A.2 Per-Class AP Evaluation for COCO and KITTI Datasets

This section provides a detailed breakdown of per-class AP scores for selected object detection models evaluated on the COCO and KITTI datasets. The goal is to highlight the effect of class imbalance on model performance and underscore the need for strategies to mitigate it.

To account for variability across repeated runs, standard deviation values were computed for each class and used to generate error bars in the visualizations. These error bars reflect the consistency of detection performance across runs and help identify classes where predictions are unstable due to imbalance or dataset complexity.

Appendix A.2.1 COCO Dataset—Per-Class AP Results

The COCO dataset, while extensive and widely used, exhibits significant class imbalance. For example, frequently occurring classes such as *person* and *car* tend to dominate training, whereas rarer classes like *chair* and *dog* are underrepresented. Fig. A5 presents the per-class AP results for four prominent models: YOLOv5, YOLOv7, YOLOv8, and YOLOv10, on a subset of COCO categories.

To incorporate statistical insight, we computed the standard deviation σ (Eq. (A1)) of AP scores over three repeated inference runs per model, for each object class. This provides a measure of variability and confidence in the reported performance:

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(AP_i - \bar{AP})^2} \tag{A1}$$

where $\bar{AP}$ represents the average precision across runs. These error bars are visualized in the diagrams to reflect performance consistency across categories.

To help interpret these values, Table A1 summarizes standard deviation thresholds and their implications for object detection performance:

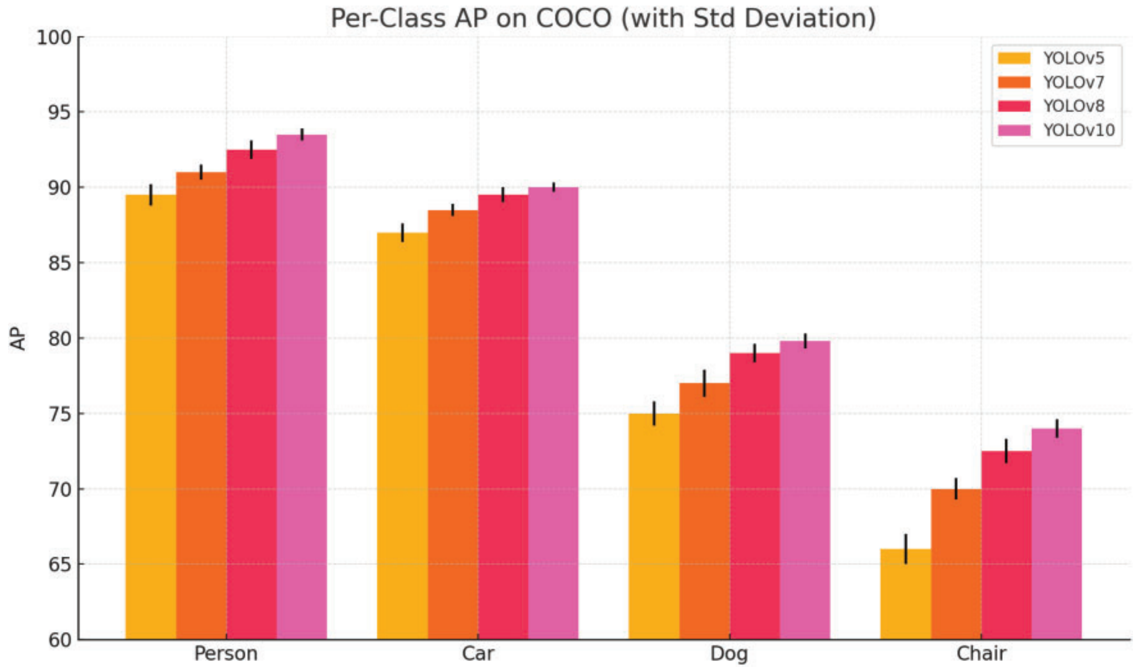


Figure A5: Per-class AP scores (with standard deviation) for selected models on COCO dataset

Table A1: Interpretation of standard deviation (σ) in detection performance

Standard deviation (σ)	Interpretation	Detection implication
$\sigma < 2\%$	Very low variability	High stability and model confidence
$2\% \leq \sigma < 4\%$	Moderate variability	Acceptable consistency, minor fluctuations
$\sigma \geq 4\%$	High variability	Unstable predictions, may indicate occlusion or class imbalance effects

Summary: Across the evaluated models, YOLOv10 achieved the highest average AP of **78.4%** and the lowest standard deviation of **±2.7%**, indicating both strong detection capability and high consistency across object categories. YOLOv8 followed with an average AP of **75.1%** and a slightly higher standard deviation of **±3.1%**, reflecting solid generalization but slightly more fluctuation across classes. Frequent classes such as *person* and *car* exhibited high AP values (above 85%) with very low variability (e.g., **±1.2%** for *person*), aligning with the "very low variability" category as defined in Table A1, and thus reflecting high model confidence.

In contrast, underrepresented categories like *dog* and *chair* not only showed lower AP scores (below 60%) but also higher standard deviations reaching **±5.4%**, which corresponds to the "high variability" range. This implies less stable predictions and potential model uncertainty. Such inconsistency becomes especially critical in occlusion scenarios, where rare classes, already underrepresented in training, face compounding detection challenges. Occlusion amplifies visual ambiguity, leading to increased false negatives and localization drift for these vulnerable classes. These findings underscore the need for occlusion-aware and class-balanced training strategies to improve detection fairness and robustness.

Appendix A.2.2 KITTI Dataset—Per-Class AP Results

The KITTI 2D dataset is known for its urban driving scenes but suffers from class imbalance, most notably, the *car* category is vastly overrepresented compared to *pedestrian* and *cyclist*. This affects the detection reliability of smaller or less frequent classes under occlusion-heavy conditions. The AP results for each class across various models are presented in Fig. A6.

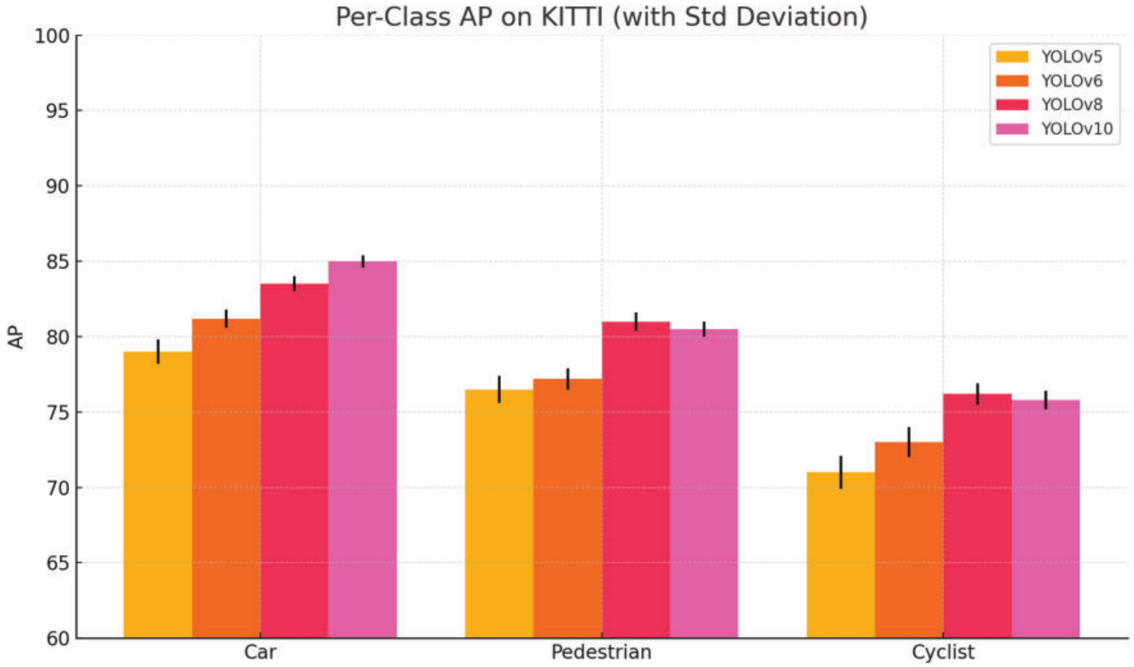


Figure A6: Per-class AP scores (with standard deviation) for selected models on KITTI dataset

Summary: In the KITTI dataset evaluation, YOLOv10 again demonstrated the strongest performance, achieving an average AP of **80.4%** with a low standard deviation of **±2.3%**, indicating robust and stable detection across the *car*, *pedestrian*, and *cyclist* classes. YOLOv8 followed closely with an average AP of **78.6%** and a slightly higher deviation of **±3.0%**. The *car* class, being highly represented, consistently produced the highest AP values (over 85%) and the lowest variability (**±1.1%**), while the *pedestrian* and *cyclist* categories exhibited weaker detection (with APs often below 75%) and more variability (up to **±4.8%**). These discrepancies in standard deviation signal the instability of model predictions for underrepresented and visually ambiguous classes. Under occlusion-heavy conditions, these weaker classes become even more prone to detection failure. The compounded effect of class imbalance and occlusion reduces detection reliability, particularly for smaller or less frequent classes, reaffirming the need for architectures that integrate robustness mechanisms, including data augmentation, weighted loss functions, and fusion with geometric depth cues.

Appendix A.2.3 Analysis and Perspective on Mitigation Strategies

The statistical analysis, including error bars computed from standard deviation across multiple inference runs, confirms that class imbalance not only reduces average precision for rare categories but also increases prediction variability. As shown in Table A1, higher standard deviations indicate less stable detections, particularly in underrepresented classes such as *dog*, *chair*, *cyclist*, and *pedestrian*. These fluctuations, when combined with occlusion, further degrade reliability, leading to missed detections and poor localization.

Although the current study does not apply dedicated mitigation techniques, several well-established strategies remain relevant and promising:

- **Weighted Loss Functions:** Assign higher loss weights to rare classes during training to counteract data imbalance.
- **Data Augmentation:** Use oversampling, synthetic data, or adversarial occlusions to diversify underrepresented classes.
- **Focal Loss:** Focus training on hard-to-classify or highly variable examples, particularly where standard deviation is elevated.

In future work, these approaches will be explored to enhance both average detection accuracy and prediction consistency across all object categories, especially under occlusion-heavy and class-imbalanced conditions.

References

1. Ye H, Zhao J, Pan Y, Cherr W, He L, Zhang H. Robot person following under partial occlusion. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). London, UK; 2023. p. 7591–7.
2. Ouardirhi Z, Mahmoudi SA, Zbakh M. Enhancing object detection in smart video surveillance: a survey of occlusion-handling approaches. Electron Personal Commun. 2024;13(3):541. doi:10.3390/electronics13030541.
3. Zhiqiang W, Jun L. A review of object detection based on convolutional neural network. In: 2017 36th Chinese Control Conference (CCC); 2017 Jul 26–28; Dalian, China. 2017. p. 11104–9.
4. Liu Z, Ning J, Cao Y, Wei Y, Zhang Z, Lin S, et al. Video swin transformer. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun 18–24; New Orleans, LA, USA. p. 3202–11.
5. Liu S, Li F, Zhang H, Yang X, Qi X, Su H, et al. DAB-DETR: dynamic anchor boxes are better queries for DETR. arXiv:2201.12329. 2022.
6. Chen X, Ma H, Wan J, Li B, Xia T. Multi-view 3D object detection network for autonomous driving. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition; 2017 Jul 21–26; Honolulu, HI, USA. p. 1907–15.
7. Yang T, Gu F. Overview of modulation techniques for spatially structured-light 3D imaging. Optics Laser Technol. 2024;169:110037.
8. Ouardirhi Z, Mahmoudi SA, Zbakh M, El Ghmary M, Benjelloun M, Abdelali HA, et al. An efficient real-time moroccan automatic license plate recognition system based on the YOLO object detector. In: 2022 International Conference on Big Data and Internet of Things; 2022 Oct 25–27; Tangier, Morocco. p. 290–302.
9. Birkl R, Wofk D, Müller M. Midas v3.1—a model zoo for robust monocular relative depth estimation. arXiv:2307.14460. 2023.
10. Bhat SF, Alhashim I, Wonka P. Adabins: depth estimation using adaptive bins. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021 Jun 20–25; Nashville, TN, USA. 2021. p. 4009–18.
11. Ming Y, Meng X, Fan C, Yu H. Deep learning for monocular depth estimation: a review. Neurocomputing. 2021;438:14–33. doi:10.1016/j.neucom.2020.12.089.
12. Ouardirhi Z, Amel O, Zbakh M, Mahmoudi SA. FuDensityNet: fusion-based density-enhanced network for occlusion handling. In: Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2024). Rome, Italy; 2024. p. 632–9.
13. Pang S, Morris D, Radha H. CLOCs: camera-LiDAR object candidates fusion for 3D object detection. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Las Vegas, NV, USA; 2020. p. 10386–93.
14. Pandya S, Srivastava G, Jhaveri R, Babu MR, Bhattacharya S, Maddikunta PKR, et al. Federated learning for smart cities: a comprehensive survey. Sustain Energy Technol Assess. 2023;55(5):102987. doi:10.1016/j.seta.2022.102987.
15. Dellermann D, Ebel P, Söllner M, Leimeister JM. Hybrid intelligence. Bus Inf Syst Eng. 2019;61:637–43.

16. He X, Liu Y, Ganesan K, Ahnood A, Beckett P, Eftekhari F, et al. A single sensor based multispectral imaging camera using a narrow spectral band color mosaic integrated on the monochrome CMOS image sensor. APL Photonics. 2020;5(4):046104. doi:10.1063/1.5140215.
17. Jeon HG, Lee JY, Im S, Ha H, Kweon IS. Stereo matching with color and monochrome cameras in low-light conditions. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA. p. 4086–94.
18. Kruegle H. Video technology overview for schools. In: The handbook for school safety and security. 1st ed. Oxford, UK: Butterworth-Heinemann; 2014. p. 195–240.
19. Lee BY, Liew LH, Cheah WS, Wang YC. Occlusion handling in videos object tracking: a survey. IOP Conf Ser: Earth Environ Sci. 2014;18:012020.
20. Nuage D. LiDAR 3D perception and object detection [Internet]; 2024 [cited 2025 Feb 10]. Available from: https://www.digitalnuage.com/lidar-3d-perception-and-object-detection.
21. Moselhi O, Bardareh H, Zhu Z. Automated data acquisition in construction with remote sensing technologies. Appl Sci. 2020;10(8):2846. doi:10.3390/app10082846.
22. Grigorescu S, Trasnea B, Cocias T, Macesanu G. A survey of deep learning techniques for autonomous driving. J Field Robotics. 2020;37(3):362–86. doi:10.1002/rob.21918.
23. Boizard N, El Haddad K, Ravet T, Cresson F, Dutoit T. Deep learning-based stereo camera multi-video synchronization. In: ICASSP 2023—2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2023 Jun 4–10; Rhodes, Greece. p. 1–5.
24. Ghannami MA, Daniel S, Sicot G, Quidu I. A likelihood-based triangulation method for uncertainties in through-water depth mapping. Remote Sens. 2024;16(21):4098. doi:10.3390/rs16214098.
25. Duba PK, Mannam NPB, Rajalakshmi P. Stereo vision based object detection for autonomous navigation in space environments. Acta Astronautica. 2024;218(17):326–9. doi:10.1016/j.actaastro.2024.02.032.
26. Pencer J, Wong FC, Bromley BP, Atfield J, Zeller M. Comparison of WIMS-AECL/DRAGON/RFSP and MCNP results with ZED-2 measurements for control device worth and reactor kinetics. In: International Conference on the Physics of Reactors 2010; 2010 May 9–14; Pittsburgh, PA, USA. p. 327–37.
27. He Y, Chen S. Recent advances in 3D data acquisition and processing by time-of-flight camera. IEEE Access. 2019;7:12495–510. doi:10.1109/access.2019.2891693.
28. Li L. Time-of-flight camera—an introduction. In: Technical white paper; 2024. SLOA190B. [cited 2025 Feb 10]. Available at https://www.ti.com/lit/pdf/sloa190.
29. Sanmartin-Vich N, Calpe J, Pla F. Analyzing the effect of shot noise in indirect Time-of-Flight cameras. Signal Process: Image Commun. 2024;122(4):117089. doi:10.1016/j.image.2023.117089.
30. Yang D, An D, Xu T, Zhang Y, Wang Q, Pan Z, et al. Object pose and surface material recognition using a single-time-of-flight camera. Adv Photonics Nexus. 2024;3(5):056001–1.
31. Elaraby AF, Hamdy A, Rehan M. A kinect-based 3D object detection and recognition system with enhanced depth estimation algorithm. In: 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). Vancouver, BC, Canada; 2018. p. 247–52. doi:10.1109/IEMCON.2018.8615020.
32. Han J, Shao L, Xu D, Shotton J. Enhanced computer vision with microsoft kinect sensor: a review. IEEE Tran Cybern. 2013;43(5):1318–34. doi:10.1109/tcyb.2013.2265378.
33. Du L, Zhang R, Wang X. Overview of two-stage object detection algorithms. J Phys: Conf Ser. 2020;1544:012033.
34. Hnoohom N, Chotivatunyu P, Jitpattanakul A. ACF: an armed CCTV footage dataset for enhancing weapon detection. Sensors. 2022;22(19):7158. doi:10.3390/s22197158.
35. Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA; 2014. p. 580–7.
36. Girshick R. Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision; 2015 Dec 7–13; Santiago, Chile. p. 1440–8.
37. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell. 2016;39(6):1137–49. doi:10.1109/tpami.2016.2577031.

38. Cai Z, Vasconcelos N. Cascade R-CNN: Delving into high quality object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 6154–62.
39. Qiao S, Chen LC, Yuille A. Detectors: detecting objects with recursive feature pyramid and switchable atrous convolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021 Jun 20–25; Nashville, TN, USA. p. 10213–24.
40. Uijlings JR, Van De Sande KE, Gevers T, Smeulders AW. Selective search for object recognition. Int J Comput Vis. 2013;104:154–71. doi:10.1109/iccv.2011.6126456.
41. Cortes C. Support-vector networks. Mach Learn. 1995;20:273–97.
42. Ren Y, Zhu C, Xiao S. Object detection based on fast/faster RCNN employing fully convolutional architectures. Math Probl Eng. 2018;2018(1):3598316. doi:10.1155/2018/3598316.
43. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA. p. 779–88.
44. Jiang P, Ergu D, Liu F, Cai Y, Ma B. A review of Yolo algorithm developments. Procedia Comput Sci. 2022;199(11):1066–73. doi:10.1016/j.procs.2022.01.135.
45. Wang C, Bochkovskiy A, Liao H. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: 2023 IEEE CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2023 Jun 17–24; Vancouver, BC, Canada. p. 7464–75.
46. Reis D, Kupec J, Hong J, Daoudi A. Real-time flying object detection with YOLOv8. arXiv:2305.09972. 2023.
47. Wang A, Chen H, Liu L, Chen K, Lin Z, Han J, et al. YOLOv10: real-time end-to-end object detection. arXiv:2405.14458. 2024.
48. Tan M, Pang R, Le QV. EfficientDet: scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2020 Jun 13–19; Seattle, WA, USA. p. 10781–90.
49. Wang CY, Bochkovskiy A, Liao HYM. Scaled-yolov4: scaling cross stage partial network. In: Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2021 Jun 20–25; Nashville, TN, USA. p. 13029–38.
50. Xu S, Wang X, Lv W, Chang Q, Cui C, Deng K, et al. PP-YOLOE: an evolved version of YOLO. arXiv:2203.16250. 2022.
51. Redmon J, Farhadi A. YOLO9000: better, faster, stronger. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition; 2017 Jul 21–26; Honolulu, HI, USA. p. 7263–71.
52. Farhadi A, Redmon J. Yolov3: an incremental improvement. In: Computer vision and pattern recognition. Berlin/Heidelberg, Germany: Springer; 2018. Vol. 1804, p. 1–6.
53. Koonce B, Koonce B. EfficientNet. In: Convolutional neural networks with swift for Tensorflow: image recognition and dataset categorization. Berkeley, CA, USA: Apress; 2021. p. 109–23.
54. Huang X, Wang X, Lv W, Bai X, Long X, Deng K, et al. PP-YOLOv2: a practical object detector. arXiv: 2104.10419. 2021.
55. Zhou Y, Tuzel O. Voxelnet: end-to-end learning for point cloud based 3D object detection. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 4490–9.
56. Yan Y, Mao Y, Li B. SECOND: sparsely embedded convolutional detection. Sensors. 2018;18(10):3337. doi:10.3390/s18103337.
57. Qi CR, Su H, Mo K, Guibas LJ. PointNet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition; 2017 Jul 21–26; Honolulu, HI, USA. p. 652–60.
58. Qi CR, Yi L, Su H, Guibas LJ. PointNet++: deep hierarchical feature learning on point sets in a metric space. Adv Neural Inf Process Syst. 2017;30:5099–108.
59. Shi S, Guo C, Jiang L, Wang Z, Shi J, Wang X, et al. PV-RCNN: point-voxel feature set abstraction for 3D object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2020 Jun 13–19; Seattle, WA, USA. p. 10529–38.

60. Ding Z, Han X, Niethammer M. VoteNet: a deep learning label fusion method for multi-atlas segmentation. In: Medical Image Computing and Computer Assisted Intervention-MICCAI 2019: 22nd International Conference; 2019 Oct 13–17; Shenzhen, China. p. 202–10.
61. Li Z, Wang W, Li H, Xie E, Sima C, Lu T, et al. BEVFormer: learning bird's-eye-view representation from LiDAR-camera via spatiotemporal transformers. IEEE Trans Pattern Anal Mach Intell. 2024;47(3):2020–36. doi:10.1109/tpami.2024.3515454.
62. Liu Y, Wang T, Zhang X, Sun J. Position embedding transformation for multi-view 3D object detection. In: 2022 European Conference on Computer Vision; 2022 Oct 23–27; Tel Aviv, Israel. p. 531–48.
63. Wang Y, Guizilini VC, Zhang T, Wang Y, Zhao H, Solomon J. DETR3D: 3D object detection from multi-view images via 3D-to-2D queries. In: 5th Conference on Robot Learning (CoRL 2021). London, UK; 2022. p. 180–91.
64. Bai X, Hu Z, Zhu X, Huang Q, Chen Y, Fu H, et al. Transfusion: robust lidar-camera fusion for 3D object detection with transformers. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun. 18–24; New Orleans, LA, USA. p. 1090–9.
65. Hu C, Zheng H, Li K, Xu J, Mao W, Luo M, et al. FusionFormer: a multi-sensory fusion in bird's-eye-view and temporal consistent transformer for 3D object detection. arXiv:2309.05257. 2023.
66. StereoLabs. StereoLabs developers—release resources for ZED cameras [Internet]. 2024 [cited 2024 Dec 2]. Available from: https://www.stereolabs.com/en-be/developers/release.
67. Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition (CVPR). Providence, RI, USA; 2012. p. 3354–61.
68. Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, et al. Microsoft COCO: common objects in context [Internet]; 2014 [cited 2025 May 15]. Available from: https://cocodataset.org/#home.
69. Xiao J, Owens A, Torralba A. RGB-D object dataset [Internet]. 2023 [cited 2025 May 15]. Available from: https://rgbd.cs.princeton.edu/.
70. Sharma P, Gupta S, Vyas S, Shabaz M. Retracted: object detection and recognition using deep learning-based techniques. IET Commun. 2023;17(13):1589–99. doi:10.1049/cmu2.12513.
71. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016 Jun 27–30; Las Vegas, NV, USA. p. 770–8.
72. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. Mobilenetv2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 4510–20.
73. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556. 2014.
74. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: single shot MultiBox detector. In: European Conference on Computer Vision; 2016 Oct 11–14; Amsterdam, The Netherlands. p. 21–37.
75. Lin TY, Goyal P, Girshick R, He K, Dollár P. Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision; 2017 Oct 22–29; Venice, Italy. p. 2980–8.
76. Sozzi M, Cantalamessa S, Cogato A, Kayad A, Marinello F. Automatic bunch detection in white grape varieties using YOLOv3, YOLOv4, and YOLOv5 deep learning algorithms. Agronomy. 2022;12(2):319. doi:10.3390/agronomy12020319.
77. Li C, Li L, Jiang H, Weng K, Geng Y, Li L, et al. YOLOv6: a single-stage object detection framework for industrial applications. arXiv:2209.02976. 2022.
78. Huang Z, Li L, Krizek GC, Sun L. Research on traffic sign detection based on improved YOLOv8. J Comput Commun. 2023;11(7):226–32. doi:10.4236/jcc.2023.117014.
79. Radosavovic I, Kosaraju RP, Girshick R, He K, Dollár P. Designing network design spaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2020 Jun 13–19; Seattle, WA, USA. p. 10428–36.
80. Zhu X, Ma Y, Wang T, Xu Y, Shi J, Lin D. SSN: shape signature networks for multi-class object detection from point clouds. In: Computer Vision-ECCV 2020: 16th European Conference; 2020 Aug 23–28; Glasgow, UK. p. 581–97.

81. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-cam: visual explanations from deep networks via gradient-based localization. In: Proceedings of the 2017 IEEE International Conference on Computer Vision; 2017 Oct 22–29; Venice, Italy. p. 618–26.

82. Muhammad MB, Yeasin M. Eigen-cam: class activation map using principal components. In: 2020 International Joint Conference on Neural Networks (IJCNN); 2020 Jul 19–24; Glasgow, UK. p. 1–7.

83. Mahmoudi SA, Gloesener M, Benkedadra M, Lerat JS. Edge AI system for real-time and explainable forest fire detection using compressed deep learning models. In: Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025); 2025 Feb 26–28; Porto, Portugal. p. 847–54.

84. Lerat JS, Mahmoudi SA. Scalable deep learning for Industry 4.0: speedup with distributed deep learning and environmental sustainability considerations. In: 2023 International Conference of Cloud Computing Technologies and Applications; 2023 Nov 21–23; Marrakesh, Morocco. 2024. p. 182–204.

85. Infrabel. Infrabel—Gestionnaire de l'infrastructure ferroviaire belge [Internet]. [cited 2025 Jan 23]. Available from: https://infrabel.be/fr.

86. Amel O, Siebert X, Mahmoudi SA. Comparison analysis of multimodal fusion for dangerous action recognition in railway construction sites. Electronics. 2024;13(12):2294. doi:10.3390/electronics13122294.

87. Li Z, Chen Z, Liu X, Jiang J. Depthformer: exploiting long-range correlation and local information for accurate monocular depth estimation. Mach Intell Res. 2023;20(6):837–54. doi:10.1007/s11633-023-1458-0.

88. Lin Z, Liu Z, Xia Z, Wang X, Wang Y, Qi S, et al. RCBEVDet: radar-camera fusion in bird's eye view for 3D object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2024 Jun 16–22; Seattle, WA, USA. p. 14928–37.

89. Jiang X, Hou Y, Tian H, Zhu L. Mirror complementary transformer network for RGB-thermal salient object detection. IET Comput Vis. 2024;18:15–32.

90. Liu H, Xue W, Chen Y, Chen D, Zhao X, Wang K, et al. A survey on hallucination in large vision-language models. arXiv:2402.00253. 2024.

91. Liu Y, Zhang K, Li Y, Yan Z, Gao C, Chen R, et al. Sora: a review on background, technology, limitations, and opportunities of large vision models. arXiv:2402.17177. 2024.