scientific reports

Check for updates

OPEN

An enhanced YOLOv10 framework for small-object safety helmet detection on construction sites

Seunghyeon Wang[1], En-Lian Zhang[2], Rong-Lu Hong[2] & Juhyung Kim[2]✉

Safety helmet compliance monitoring remains challenging because helmets often occupy only a few pixels in wide-area surveillance images. This study tackles these small-object difficulties in vision-based safety-helmet compliance monitoring (helmet vs. no-helmet) using wide-area surveillance imagery. An enhanced You Only Look Once version 10 (YOLOv10) detector is proposed by integrating Omni-Dimensional Dynamic Convolution (ODConv) into the backbone, an Efficient Multi-scale Attention–guided Bidirectional Feature Pyramid Network (EMA-BiFPN) for multi-scale feature fusion, a four-head detection scheme, and the Minimum Points Distance Intersection over Union (MPDIoU) regression loss. Across the experiments, the proposed detector reached an mAP50 of 94.28, with AP50 values of 96.55 for the helmet class and 92.01 for the no-helmet class, exceeding the performance of the YOLOv10 baseline and other benchmark detectors. The most notable gains appeared for extremely small and small targets ($\mathbf{AP}_{ES}$ = 86.10, $\mathbf{AP}_{S}$ = 91.55), reflecting improved localization of helmets at long distances. Overall, these findings indicate that the method is a strong candidate for deployment-focused helmet-compliance monitoring in large-scale construction settings, although performance limitations remain most evident in far-field views and highly crowded scenes.

Keywords Safety helmet monitoring, Small object detection, Computer vision, Deep learning, You only look once version 10

Construction sites continue to face serious safety risks even though regulatory frameworks—such as those set by the Occupational Safety and Health Administration (OSHA)—and modern monitoring tools (e.g., sensors, artificial intelligence, and computer vision) are increasingly used. In the United States, construction recorded the largest number of fatal workplace injuries in 2021. The Bureau of Labor Statistics (BLS) reported 1,034 deaths that year, accounting for roughly 21% of total occupational fatalities[1].

Safety helmets are one of the most critical pieces of Personal Protective Equipment (PPE) because they reduce the likelihood of head injuries caused by falling or flying objects. Despite this importance, helmet usage is not consistently enforced in practice, often due to discomfort, insufficient awareness, or lax compliance. Consequently, verifying whether workers are properly wearing helmets is a key requirement for improving site safety. Traditional approaches rely on manual patrols and human review of surveillance footage. These methods demand substantial labor and time, and continuous screen-based monitoring can degrade performance as fatigue accumulates, increasing the chance of missed violations[2]. For these reasons, automated helmet-monitoring solutions have become an active area of development, supported by progress in imaging systems and data-driven analysis.

Deep learning–based object detection is well suited to automated safety-helmet compliance monitoring because it can localize and classify targets in images captured by fixed surveillance cameras or wearable devices. In object detection, models output bounding boxes and corresponding class labels for objects of interest. Among widely used detectors, the You Only Look Once (YOLO) family is popular for real-time applications—including construction safety—because it provides a strong balance between accuracy and inference speed[3–5]. Accordingly, many recent studies have proposed YOLO-based variants to improve helmet-monitoring performance, and successive YOLO versions have introduced architectural refinements that further enhance detection accuracy.

Despite these advances, reliable helmet compliance monitoring in real construction scenes remains difficult, largely because the visual evidence needed to determine compliance is often confined to a small head/helmet region. In distant or wide-angle views, this region may occupy only a few pixels, so cues that distinguish helmet from no-helmet can be weakened after feature-map downsampling. Moreover, these small-scale cues may be

[1]Department of Cognitive Convergence Science, Hanyang University, Seongdong-gu, Seoul 133791, Korea. [2]Department of Architectural Engineering, Hanyang University, Seongdong- gu, Seoul 133791, Korea. ✉email: kcr97jhk@hanyang.ac.kr

diluted during multi-scale feature fusion and further suppressed by post-processing steps (e.g., confidence filtering or non-maximum suppression), particularly in crowded scenes where workers overlap. As a result, feature quality at small scales deteriorates and localization becomes less stable under frequent occlusion[6,7]. However, small-object–oriented architectures specifically designed for helmet compliance monitoring (helmet vs. no-helmet) remain underexplored.

To address these challenges, an enhanced YOLOv10–based framework is developed for vision-based safety-helmet compliance monitoring (helmet vs. no-helmet). Omni-Dimensional Dynamic Convolution (ODConv) is integrated into the backbone to adapt convolutional responses across spatial locations, channels, and kernel configurations, improving sensitivity to small helmets under clutter and scale variation. An Efficient Multi-scale Attention–guided Bidirectional Feature Pyramid Network (EMA-BiFPN) is introduced to strengthen cross-scale feature aggregation: attention highlights informative fine-grained cues, while bidirectional fusion combines shallow spatial details with deeper semantic representations.

For bounding-box regression, Minimum Points Distance Intersection over Union (MPDIoU) is adopted to incorporate geometric distance constraints beyond overlap, stabilizing localization for small and heavily overlapping instances. In addition, a four-head detection scheme is investigated to better cover extremely small, small, medium, and large targets, improving robustness to the wide range of helmet sizes observed in practical imagery. The main contributions are as follows:

1) A YOLOv10-based helmet-monitoring framework is proposed that integrates ODConv, EMA-BiFPN, and MPDIoU, together with a four-head detection strategy.
2) The effects of hyperparameters are analyzed, and sensitivity to stochastic factors—including weight initialization and data-split variability—is quantified.
3) Validation-to-test generalization is evaluated, the impact of image conditions is examined, and an ablation study is conducted to quantify the individual impact of each component. The performance is also assessed on benchmark datasets, and results are compared with those of other representative detectors.

Related work

Existing studies for vision-based safety helmet monitoring

A large body of research has investigated deep-learning-based safety-helmet monitoring on construction sites, most often relying on CNN-driven object detectors. Early efforts commonly used two-stage pipelines. For instance, Fang et al.[8] employed Faster R-CNN and reported strong accuracy (90.1%–98.4%) across multiple scenarios; however, inference took approximately 0.2 s per image, limiting real-time deployment. Gu et al.[9] enhanced Faster R-CNN through multi-scale training and broader anchor configurations, achieving an additional ~ 7% accuracy improvement, although computational speed was not explicitly reported.

Given that practical systems must balance accuracy with runtime efficiency, many later studies moved toward single-stage detectors. Yang et al.[10] integrated an image-pyramid strategy into YOLOv3 (DarkNet53 backbone) to better preserve multi-scale cues, reaching 95.13% accuracy with 0.017 s latency per image and outperforming SSD and Faster R-CNN. Yan and Wang[11] redesigned the YOLOv3 backbone with dense connections, increasing processing speed by 52% and improving accuracy by 5.7% while maintaining efficient inference. Shen et al.[12] presented a hybrid approach in which faces were detected using VGG16, followed by helmet recognition via regression-based region mapping and DenseNet classification. Nath[3] first localized workers using YOLOv3 and then categorized PPE with VGG16, ResNet, and Xception, reporting 71.23% accuracy. Wang et al.[13] compared YOLOv5 variants, showing that YOLOv5x achieved higher accuracy (86.55%), whereas YOLOv5s provided faster processing (0.019 s per frame). Lee et al.[14] proposed a real-time method using YOLACT with MobileNetV3, obtaining 66.4% accuracy for helmet and vest detection. Nguyen et al.[15] tuned YOLOv5 using the Seahorse Optimization algorithm (SHO) and reported an average accuracy of roughly 66% across multiple PPE categories (helmets, masks, gloves, vests, and shoes). Park et al.[16] examined individual augmentation techniques and combined YOLOv8 with transformer backbones to improve non-PPE detection. More recently, Wang et al.[17] introduced YOLOv10-based helmet/no-helmet detectors enhanced with transformer variants. Related studies by Wang[18] addressed non-PPE detection in surveillance and body-worn imagery using benchmark datasets, while Wang[19] explored domain-adaptive Faster R-CNN to alleviate class imbalance, particularly for non-helmet classes. In addition, Wang et al.[20] developed YOLOv10-based methods for detecting multiple categories of PPE non-compliance, with particular attention to multi-scale object variations.

Taken together, prior studies demonstrate that deep-learning detectors can deliver strong performance under their respective experimental settings. Nevertheless, helmets frequently appear as tiny objects in real construction footage, and accuracy may drop due to scale variation, feature downsampling, dense crowds, and occlusion. As a result, approaches specifically tailored to small-object detection—along with evaluations that explicitly report size-aware metrics—remain relatively scarce in the safety-helmet monitoring literature.

Studies on YOLO-based small object detection

A single, universally accepted definition of a "small object" is not available because the apparent size of a target depends on factors such as image resolution, viewing distance, and the evaluation protocol. In many benchmarks, a practical pixel-based convention is used; for example, instances with bounding-box areas smaller than 32×32 pixels are often categorized as small objects[21].

Within the YOLO family, numerous variants have been proposed, and prior efforts to enhance small-object performance in YOLO-style detectors can be broadly grouped into three research directions. First, many studies aim to preserve fine spatial detail by strengthening high-resolution feature pathways and improving multi-scale fusion through pyramid-based neck designs, including Feature Pyramid Network (FPN), Path Aggregation Network (PANet), and Bidirectional Feature Pyramid Network (BiFPN). For instance, Chao et al.[22]

introduced a YOLOv8 variant with a Deep Spatial Pyramid Fusion (DSPF) module that reduces redundancy and computational burden from residual blocks while maintaining effective multi-scale aggregation in lightweight settings.

Second, other work focuses on amplifying representations of tiny targets by incorporating lightweight attention mechanisms—such as Squeeze-and-Excitation (SE) and the Convolutional Block Attention Module (CBAM)—or by adopting dynamic convolution strategies, including ODConv. In this line, He and Wei[23] improved wear-particle recognition by embedding CBAM into a YOLOv5-based detector to emphasize informative features.

Third, localization performance under stricter IoU requirements has been improved through IoU-aware regression objectives, such as Generalized IoU (GIoU), Distance IoU (DIoU), and Complete IoU (CIoU). Su et al.[24] evaluated several IoU-based losses and reported that GIoU, DIoU, and CIoU can outperform the standard IoU loss in terms of localization accuracy, although the magnitude of improvement can depend on the dataset (e.g., PASCAL VOC). More recently, Zou and Liu[25] combined YOLOv10 with ODConv and an MPDIoU-based regression loss to increase precision and recall for irregularly shaped targets in traffic-sign detection.

Despite these advances, detecting small objects remains challenging. Tiny instances often produce weak feature responses after repeated downsampling, may be attenuated during multi-scale fusion, and can be suppressed during post-processing in crowded scenes. This challenge is particularly relevant to worker-centered helmet compliance monitoring, where each worker must be classified as "helmet" or "no-helmet" using head-region cues that may occupy only a few pixels in distant or wide-angle views and are further complicated by occlusion and dense crowds.

To mitigate these issues, the proposed framework builds on YOLOv10 and combines multiple complementary designs: (i) ODConv, which employs four-dimensional dynamic kernels to better capture multi-scale patterns and reduce background interference; (ii) EMA-BiFPN to strengthen multi-scale feature aggregation for small targets; (iii) an MPDIoU-based bounding-box regression loss to improve localization, especially for difficult or irregular cases; and (iv) a four-head detection scheme to enhance high-resolution prediction.

Proposed approach

Overview of the YOLOv10 baseline for helmet/no-helmet detection

YOLOv10 is a one-stage object detector designed for efficient end-to-end detection without relying on Non-Maximum Suppression (NMS)[18]. This is achieved through a consistent dual-assignment strategy, which reduces duplicate predictions while preserving the real-time inference advantage of the YOLO family. As shown in Fig. 1, YOLOv10 follows the standard backbone–neck–head structure commonly used in modern object detectors.

The backbone extracts hierarchical visual representations from the input image through lightweight convolutional modules and attention-based components. Lower-level stages retain relatively rich spatial information, which is important for localizing small objects, whereas higher-level stages provide stronger semantic abstraction. The neck then aggregates multi-resolution features through a feature pyramid–style fusion process, combining high-resolution spatial cues with deeper semantic representations. Finally, the detection head performs classification and bounding-box regression using a lightweight decoupled prediction structure.

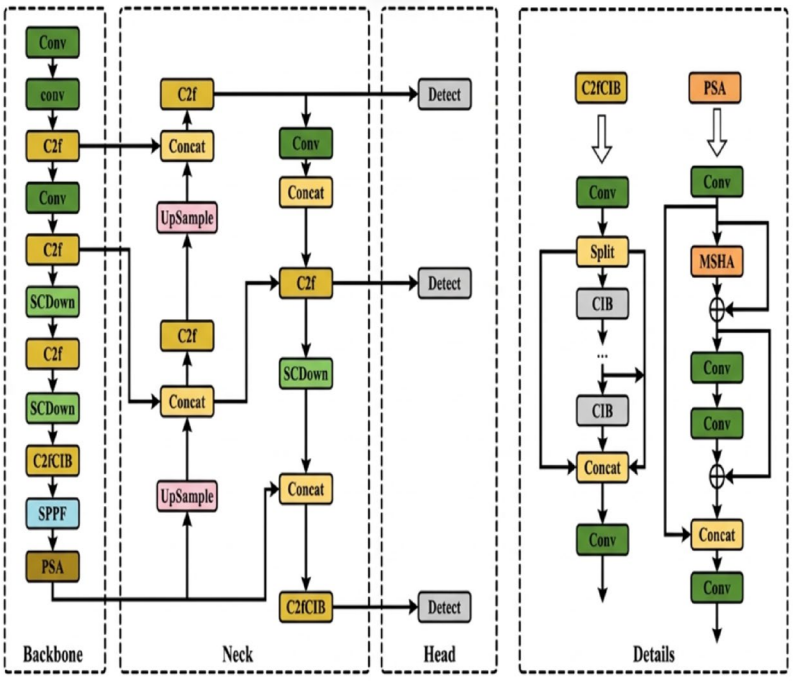


Fig. 1. Architecture of the baseline YOLOv10 model.

In this study, YOLOv10 is adopted as the baseline detector for two-class head-level safety compliance monitoring: "Helmet" and "No-helmet." This formulation directly reflects the practical safety-monitoring task because compliance is determined by whether the worker's head region contains a properly worn helmet. However, in wide-area construction-site surveillance images, helmet and no-helmet regions often appear as very small targets. Their visual cues can be weakened by distance, downsampling, occlusion, illumination variation, motion blur, and background clutter. Therefore, although YOLOv10 provides an efficient detection framework, additional task-specific modifications are required to improve the detection of small, far-field, occluded, and weak-feature helmet/no-helmet instances.

Architectural enhancements of YOLOv10 for head-level compliance monitoring

In real construction-site surveillance imagery, helmet compliance detection is fundamentally a head-level small-object detection problem. Unlike general object detection, the visual evidence needed to distinguish a worker wearing a helmet from a worker without a helmet is usually confined to a very small region around the head. This problem becomes more severe in wide-angle or far-field surveillance views, where a helmet or uncovered head may occupy only a few pixels after image resizing. In such cases, repeated downsampling in the backbone can weaken fine spatial details, multi-scale fusion can dilute small-object cues, and dense worker arrangements can make neighboring head regions overlap or merge. These factors often lead to missed detections, inaccurate localization, or confusion between the Helmet and No-helmet classes.

To address these challenges, an enhanced YOLOv10-based framework is constructed, as illustrated in Fig. 2. The proposed framework preserves the backbone–neck–head structure of YOLOv10 but introduces task-specific modifications at each stage of the detection pipeline. Figure 3 further highlights the architectural differences between the original YOLOv10 and the modified model. Specifically, ODConv is incorporated into the backbone for adaptive weak-feature extraction, EMA-BiFPN is introduced in the neck for attention-guided multi-scale fusion, an additional high-resolution prediction head is added for extremely small objects, and MPDIoU is adopted as the bounding-box regression loss for stable tiny-object localization.

Importantly, the proposed framework is not intended as a simple aggregation of existing modules. Instead, each modification is selected to address a specific failure mode observed in construction-site helmet monitoring. ODConv is used to improve adaptive feature extraction for weak and ambiguous helmet/head cues. EMA-BiFPN is used to strengthen multi-scale feature propagation while preserving fine details for small objects. The additional high-resolution detection head is used to increase sensitivity to extremely small far-field helmets and uncovered heads. MPDIoU is used to provide more stable localization supervision for tiny bounding boxes, where even a small coordinate deviation can substantially change the IoU value. Table 1 summarizes the relationship between the main detection challenges and the corresponding components of the proposed framework.

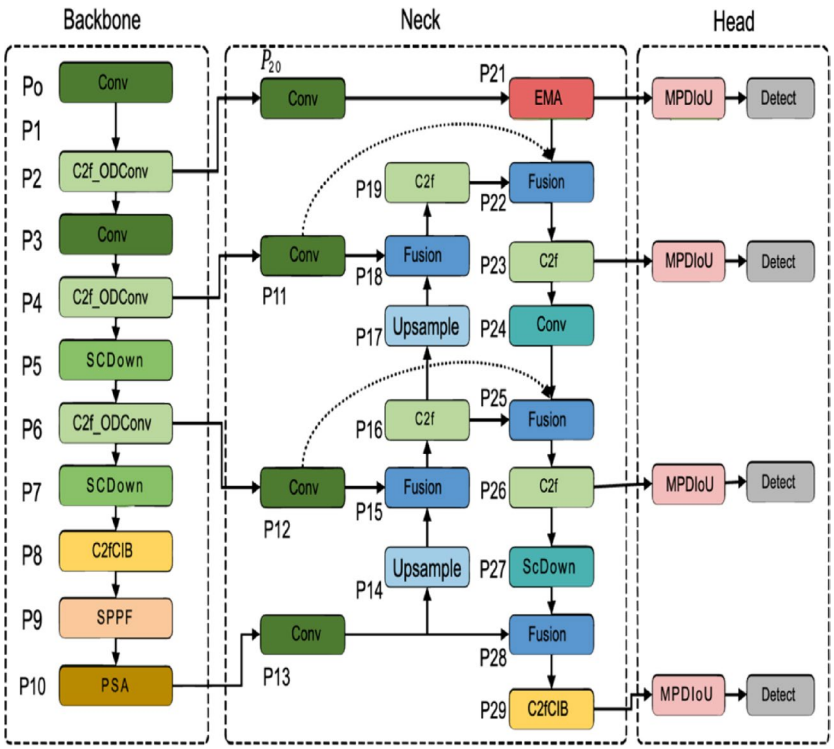


Fig. 2. Structural overview of the proposed modifications to YOLOv10.

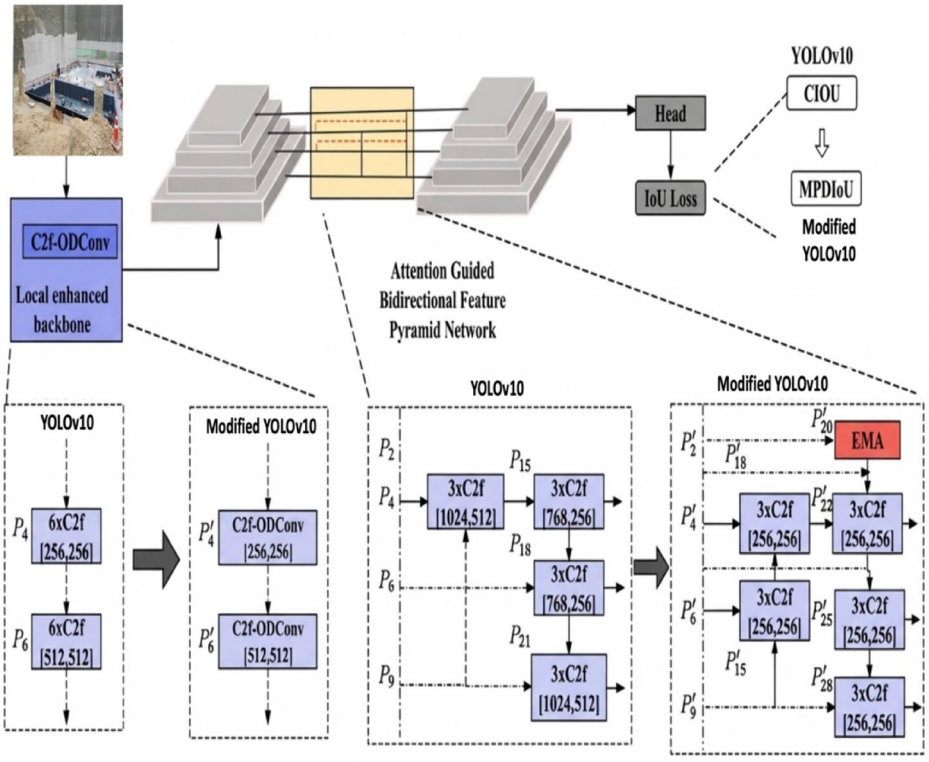


Fig. 3. Architectural differences between the original YOLOv10 and the modified model.

Detection challenge	Effect in construction-site imagery	Proposed component	Task-specific role
Extremely small helmets/heads	Far-field workers may occupy only a few pixels, and head-level details are weakened after downsampling	Four-scale prediction heads	Adds a high-resolution prediction level for extremely small helmet/no-helmet targets
Weak helmet/no-helmet cues	Helmet rims, hair, skin, shadows, and background objects may appear visually similar	ODConv-enhanced backbone	Dynamically adapts convolutional responses to local content and improves discrimination of subtle head-level cues
Feature dilution during fusion	Shallow features preserve spatial detail but lack semantics; deep features provide semantics but lose fine details	EMA-BiFPN	Strengthens bidirectional multi-scale fusion and attention-guided preservation of informative small-object cues
Occlusion and crowding	Nearby workers may overlap, causing merged, shifted, or missed detections	MPDIoU regression loss	Adds geometric distance constraints to improve tiny-box localization stability
Far-field no-helmet detection	Missing an uncovered head is safety-critical, especially when the worker is distant or partially occluded	ODConv + EMA-BiFPN + four-scale head + MPDIoU	Jointly improves weak-feature extraction, multi-scale representation, high-resolution prediction, and localization precision

Table 1. Relationship between helmet-monitoring challenges and proposed components.

ODConv-enhanced backbone for tiny helmet/head cues

Helmet and no-helmet targets in construction scenes often exhibit limited and ambiguous visual patterns (e.g., partial helmet rims, specular highlights, hair/skin textures, and shadow boundaries). Such patterns can be easily confused with background elements such as concrete textures, reflective equipment, bright vests, or complex scaffolding lines. In addition, because the target is small, minor changes in viewpoint and illumination can significantly alter appearance.

To improve robustness to these variations, ODConv is incorporated into the C2f module. ODConv dynamically modulates convolution operations along four dimensions—spatial locations, input channels, output channels, and kernel indices—allowing the effective kernel response to adapt to the input content. Compared with earlier dynamic convolution approaches (e.g., CondConv and DyConv) that typically modulate only part of the convolution process, ODConv provides richer kernel adaptation while maintaining practical computational efficiency for real-time inference[26,27]. This property is well aligned with head-level helmet monitoring, where subtle local cues must be extracted reliably despite clutter and scale changes. Formally, ODConv can be expressed as:

$$y = (a_{w1}W_1 + \cdots + a_{wn}W_n) * x \tag{1}$$

Here, x and y represent the input and output feature maps, respectively; W_n is the n -th convolution kernel, and a_{wn} is its associated weight. With omni-dimensional attention, each kernel is further modulated by multiple attention terms:

$$y = (a_{w1} \odot a_{f1} \odot a_{c1} \odot a_{s1} \odot W_1 + \cdots + a_{wn} \odot a_{fn} \odot a_{cn} \odot a_{sn} \odot W_n) * x \tag{2}$$

where $\odot$ denotes element-wise multiplication and a_{fn}, a_{cn}, and a_{sn} represent attention factors related to feature responses, channels, and spatial locations. This omni-dimensional modulation is intended to improve discrimination between helmet and no-helmet regions by strengthening weak, fine-grained cues in the presence of background confusion.

EMA-BiFPN

Head-level helmet targets appear across a wide range of scales because camera placement and worker distance vary significantly. When the target becomes very small, reliance on high-resolution features increases; however, these features may lack strong semantic context. Conversely, deep features provide semantics but are spatially coarse and may not preserve tiny helmet/head structures after downsampling. Therefore, effective helmet/no-helmet detection benefits from fusion that propagates semantic context to high-resolution maps while preventing small-object cues from being overwhelmed.

To address this, an EMA-BiFPN is constructed by combining bidirectional feature pyramid fusion with EMA[27]. EMA refines features by jointly modeling spatial and channel relationships, thereby emphasizing informative regions related to small head/helmet cues during fusion. BiFPN-style aggregation supports repeated top-down and bottom-up information flow, enabling both high-resolution detail and deep semantic context to be reused more effectively. In addition, shortcut connections between intermediate fusion nodes are included to reduce feature attenuation and to preserve small-scale information across the pyramid[28].

Given an input feature map $X \in \mathbb{R}^{\mathrm{H} \times \mathrm{W} \times \mathrm{C}}$, EMA divides the channels into GGG groups X_g to limit computation. Two lightweight branches (e.g., 1×1 and 3×3 convolutions) capture complementary context. Global average pooling generates channel attention terms:

$$G_{avg}^{1\times 1} = GAP\left(X_g^{1\times 1}\right), \; G_{avg}^{3\times 3} = GAP\left(X_g^{3\times 3}\right) \tag{3}$$

and the group-wise attention-refined output is computed as:

$$X_g^{EMA} = G_{avg}^{1\times 1} \odot X_g^{1\times 1} + G_{avg}^{3\times 3} \odot X_g^{3\times 3} \tag{4}$$

By concatenating all X_g^{EMA} along the channel dimension, we obtain the final refined feature representation. The proposed attention-guided fusion enhances cross-scale contextual understanding without significantly increasing computation, making it suitable for real-time helmet monitoring.

MPDIoU-based regression for stable localization of tiny boxes

Accurate localization is essential for head-level objects because small coordinate shifts can substantially change IoU and can also affect class confidence due to misalignment with the true head/helmet region. For tiny targets, IoU-based losses may become unstable: a 1–2 pixel shift can cause a large drop in overlap, weakening the learning signal for precise regression.

YOLOv10 commonly employs Complete IoU (CIoU)[29]:

$$L_{CIOU} = 1 - IoU\left(B_{gt}, B_{prd}\right) + \frac{\rho^2\left(c_{gt}, c_{prd}\right)}{c^2} + \alpha v \tag{5}$$

Here, $\mathrm{IoU}\left(B_{gt}, B_{prd}\right) = \frac{B_{gt} \cap B_{prd}}{B_{gt} \cup B_{prd}}$, $\rho\left(c_{gt}, c_{prd}\right)$ denotes the Euclidean distance between the two box centers, and c is the diagonal length of the smallest enclosing box. The aspect-ratio term is:

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w_{gt}}{h_{gt}} - \arctan\frac{w_{prd}}{h_{prd}}\right)^2, + \alpha = \frac{v}{\left(1 - \mathrm{IoU}\left(B_{gt}, B_{prd}\right)\right) + v} \tag{6}$$

Although CIoU improves convergence by adding geometric constraints beyond overlap, the training signal may still be weak when overlap is ambiguous for tiny boxes. To strengthen regression for head-level small objects, MPDIoU (Minimum Points Distance IoU) is adopted[30]. MPDIoU introduces additional constraints by penalizing distances between corresponding corner points. For two boxes A and B, the squared distances between matched corners are:

$$d_1^2 = \left(x_1^B - x_1^A\right)^2 + \left(y_1^B - y_1^A\right)^2 \tag{7}$$

$$d_2^2 = \left(x_2^B - x_2^A\right)^2 + \left(y_2^B - y_2^A\right)^2 \tag{8}$$

and MPDIoU is computed as:

$$MPDIoU = \frac{A \cap B}{A \cup B} - \frac{w^2 + h^2}{d_1^2 + w^2 + h^2} - \frac{w^2 + h^2}{d_2^2 + w^2 + h^2} \tag{9}$$

where w and h refer to denote the box width and height in the image. By incorporating corner-distance constraints, MPDIoU can provide a stronger and more stable learning signal for tiny boxes, improving both position and scale estimation.

Four-scale prediction heads for extremely small helmets/heads
Another difficulty is that the original detection heads may not provide sufficient resolution for extremely small head-level objects. In far-field surveillance views, a helmet or uncovered head may be represented by only a few pixels in the input image and by an even smaller number of activations in deeper feature maps. If prediction is performed only on relatively coarse feature levels, extremely small heads can be missed, and adjacent workers in crowded scenes may be assigned to a single merged detection.

To increase sensitivity at the finest scale, the detector is extended to a four-head multi-scale prediction scheme. An additional high-resolution pyramid level is generated from shallower neck features that preserve richer spatial detail. This feature level is refined through bidirectional fusion and connected to an extra detection head. Consequently, predictions are produced at four scales intended to cover extremely small, small, medium, and large head-level objects, respectively. During training, the total loss is computed by summing losses across all heads:

$$L_{Total} = \sum_{s=1}^{4} (L_{Cls}^{(s)} + L_{Obj}^{(s)} + \lambda L_{Reg}^{(s)}) \tag{10}$$

where $s \in \{1, 2, 3, 4\}$ indexes the prediction head from highest resolution (extremely small instances) to lowest resolution (large instances), $L_{Cls}^{(s)}$ and $L_{Obj}^{(s)}$ denote classification and objectness losses, and λ balances the regression term. Regression is defined using MPDIoU:

$$L_{Reg}^{(s)} = 1 - MPDIoU(B_{gt}^{(s)}, B_{prd}^{(s)}) \tag{11}$$

where $B_{gt}^{(s)}$ and $B_{prd}^{(s)}$ are the matched ground-truth and predicted boxes associated with detection head s. The added high-resolution head, together with MPDIoU-based supervision, provides stronger learning signals for extremely small helmets/heads and reduces missed detections in dense scenes while retaining robustness across scales.

Evaluation metrics

Average precision (AP)
Average Precision (AP) is a standard metric for evaluating object detectors. It captures the precision–recall trade-off by integrating precision over the full recall range [0,1] under a given IoU threshold. In this study, AP is computed as:

$$AP = \sum_{i=1}^{n} p(i) \Delta r(i) \tag{12}$$

where n is the number of detections sorted by confidence, $p(i)$ is the precision after the top i detections are considered, and $\Delta r(i)$ is the recall increment from rank i–1 to i. AP50 refers to AP evaluated at IoU = 0.50, allowing relatively tolerant box matching. In contrast, AP50:95 is more demanding and is obtained by averaging AP over IoU thresholds from 0.50 to 0.95 with a step size of 0.05. Because it rewards tighter localization—particularly important in crowded scenes where instances may overlap—both AP50 and AP50:95 are reported.

Size-specific average precision
To examine performance across different object scales, we additionally report size-dependent AP. All images are resized to 640 × 640 using aspect-ratio-preserving letterbox resizing, and bounding-box sizes are measured in this resized coordinate system. For a ground-truth box with width w and height h (in pixels), the size indicator is defined as:

$$s = min(w, h) \tag{13}$$

Based on s, objects are categorized into four groups: Extremely Small (ES) s<8, Small (S) 8≤ s<16, Medium (M) 16≤ s<32, and Large (L) s≥32. For each group, AP_{ES}, AP_{S}, AP_{M}, and AP_{L} are computed by evaluating only instances within that size range, following the same AP procedure as the primary results (AP50) used in this work.

Detection speed
Inference speed is measured as the time needed to process a single image and is reported in Frames per Second (FPS), i.e., the number of images the model can handle per second. This metric indicates computational efficiency and helps assess whether a detector is practical for real-time safety helmet monitoring on construction sites.

Experiments

Data preparation

Original dataset construction
The present study employed an existing construction-site surveillance image dataset that had been established and used in our prior studies[17,18]. No additional image or video data were collected specifically for this work.

The original dataset was captured using 42 Bosch FLEXIDOME IP Starlight 8000i surveillance cameras installed across seven construction sites to provide wide visual coverage of site activities. Representative frames showing workers in the collected surveillance images are provided in Fig. 4.

In the original dataset construction process, each camera continuously recorded video, and still images were generated by sampling the video streams at fixed time intervals. To unify the input format and limit storage requirements, all sampled frames were resized to 640×640 pixels prior to training and evaluation. Because adjacent frames captured from the same viewpoint contained substantial visual overlap, the data were filtered to retain a representative subset reflecting diverse site conditions, including lighting variation, occlusions, cluttered backgrounds, and different viewing angles. The final surveillance-camera dataset included 10,896 images and was used to develop detection models that can generalize across diverse construction environments.

Data split

Images from the shared camera configuration were pooled across all seven sites and randomly divided into training, validation, and test sets using a 60/20/20 ratio. With a total of 10,896 images, this corresponds to 6,538 training images, 2,179 validation images, and 2,179 test images. This random partitioning was used to maintain a consistent and unbiased evaluation procedure.

Annotation

Images of construction sites were annotated by drawing rectangular bounding boxes around each visible instance. Each box was labeled as one of two classes: "Helmet", indicating a worker wearing compliant head protection, or "No-helmet", indicating the absence of a helmet or non-compliant headgear. Regions without relevant head protection were treated as background. Annotations were produced using LabelImg and exported in VOC XML format (Fig. 5).

To promote consistent and reliable labels, four annotators reviewed one another's work and reconciled disagreements through discussion until a consensus was reached. Typical difficulties included cases where helmets were partially occluded, visually unclear, or difficult to distinguish due to illumination changes and varying viewing angles. A summary of the resulting annotations with scale information for each dataset is provided in Table 2.



Fig. 4. Examples of surveillance camera images for safety helmet monitoring.

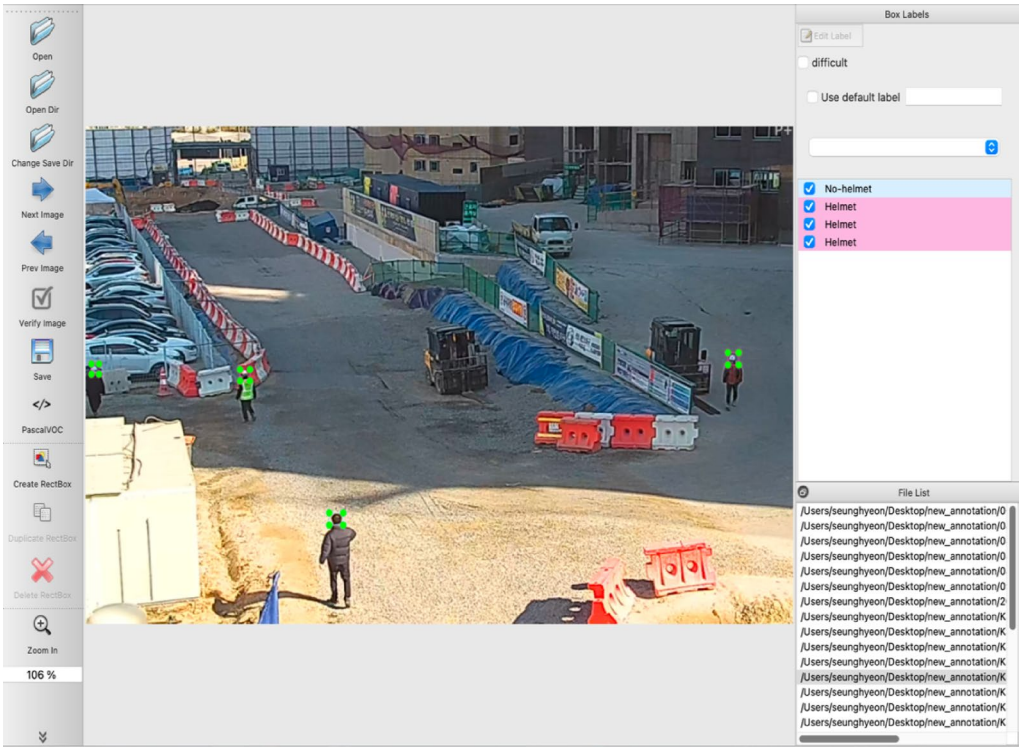


Fig. 5. Example annotations for two classes: "Helmet" and "No-helmet."

Purpose	Number of images	Number of safety helmet	Number of non-safety helmet	Scale			
				AP_{ES}	AP_S	AP_M	AP_L
Training	6,538	16,236	6,988	8,184	8,892	4,534	1,614
Validation	2,179	4,037	4,036	2,843	3,093	1,576	561
Test	2,179	3,991	4,092	2,848	3,095	1,578	562
Total	10,896	24,264	15,116	13,875	15,080	7,688	2,737

Table 2. Distribution of images and instances across the dataset.

Transfer learning with a pre-trained model

Training deep networks from scratch typically requires large labeled datasets, substantial computational resources, and long optimization times. To reduce these demands, transfer learning is employed: each detector is initialized with weights learned on a large reference dataset and then fine-tuned for the target task[31]. COCO-pretrained weights are used for initialization because COCO is a widely adopted benchmark for object detection and provides strong supervision for both classification and localization[32].

Representations learned from diverse object categories and scenes can accelerate convergence and improve generalization when adapting to safety helmet and no-helmet detection, where targets may appear small, densely distributed, and embedded in cluttered backgrounds. During fine-tuning, the pretrained weights are updated using the safety helmet dataset to specialize the feature representations for the target domain.

Comparative methods

To assess the proposed approach under consistent experimental conditions, a series of baseline comparisons was established (Table 3). As a first step, YOLOv10 was used as a common detection framework while varying only the backbone network. The evaluated backbones include CNN-based designs (CSPNet, ConvNeXt, and EfficientNet) as well as transformer-based designs (Swin Transformer, Vision Transformer (ViT), Pyramid Vision Transformer (PVT), MobileViT, and Axial Transformer). With the detection head and training pipeline held constant, observed performance differences largely reflect the influence of backbone choice.

To place the results in a broader context, additional experiments were conducted using representative one-stage, two-stage, and transformer-based detectors (Table 3). The one-stage group includes YOLOv9 evaluated at multiple scales (YOLOv9t–YOLOv9e) and YOLOv8 implemented with several architectural variants (CSPDarknet–DETR, CSPDarknet–T-FPN, and CSPDarknet–CBAM). For two-stage baselines, Faster R-CNN with FPN was tested with ResNet-101, ResNet-152, and MobileNetV3 backbones. Transformer-based baselines consist of DETR with ResNet-50 and ResNet-101, and DINO with ResNet-50 and Swin Transformer backbones.

Detector family	Detector	Architecture
One-stage detector	YOLOv10	CSPNet/ConvNeXt/EfficientNet/Swin Transformer/ViT/PVT/MobileViT/Axial Transformer
	YOLOv9	YOLOv9t/YOLOv9s/YOLOv9m/YOLOv9c/YOLOv9e
	YOLOv8	CSPDarknet- DETR/CSPDarknet- T-FPN/CSPDarknet-CBAM
Two-stage detector	Faster R-CNN + FPN	ResNet-101
		ResNet-152
		MobileNetV3
Transformer detector	Deformable DETR	ResNet-50
		ResNet-101
	DINO	ResNet-50
		Swin Transformer

Table 3. Comparison of detection frameworks and architectural components.

Hyperparameter optimization

Detection performance can be affected by hyperparameter choices[33]; therefore, YOLOv10-based models, including the baseline YOLOv10 variants and the proposed method, were optimized using a predefined hyperparameter search space. The search covered batch size {1, 2, 4}, learning rate {0.00025, 0.0005, 0.001}, weight decay {0.0001, 0.0003, 0.0005}, momentum {0.5, 0.7, 0.9}, and maximum training iterations {20,000, 30,000}. The best-performing configuration was selected based on validation-set performance and then evaluated on the held-out test set.

For comparative detectors outside the YOLOv10 framework, the same dataset split, input resolution, class definitions, and evaluation metrics were used to ensure consistency. However, their training followed the official or framework-recommended configurations rather than the full YOLOv10 hyperparameter search space. Specifically, Faster R-CNN, SSD, R-FCN, Deformable DETR, and DINO were implemented using the MMDetection framework, while YOLO-based detectors were implemented using the Ultralytics toolkit. Framework-specific default optimizer and augmentation settings were retained where necessary for compatibility. This protocol was adopted to provide a reproducible comparison while avoiding arbitrary manual tuning of each baseline detector[16,17,34].

Hardware and software environment

All experiments were conducted on a Windows 10 workstation configured with an Intel Core i7-14700 K processor running at 3.80 GHz, an NVIDIA GeForce RTX 4090 GPU with 24 GB of VRAM, and 62 GB of system memory. The implementation was carried out in Python, with PyTorch and TensorFlow used as the main deep-learning frameworks for model training and development. CUDA-enabled GPU acceleration was used for computationally intensive training and inference processes, while the CPU was mainly responsible for data loading, preprocessing, and augmentation.

Results and discussion

Training and validation performance

Effect of hyperparameters

Tables S.1–S.2 summarize detection performance across all tested hyperparameter combinations, while Fig. 6 visualizes the same distributions using boxplots. The results indicate that backbone ranking depends on the evaluation stringency: Swin Transformer achieves the highest baseline mean under AP50 (92.41), whereas PVT slightly leads under AP50:95 (68.30), suggesting different strengths in coarse detection versus precise localization. Notably, transformer performance is not uniform—ViT and MobileViT consistently underperform across both metrics—indicating that gains are architecture-specific rather than inherent to transformer designs. Across backbones, dispersion increases under AP50:95 (e.g., Swin std = 1.87 vs. 0.91 for AP50), implying that hyperparameter choices influence localization quality more strongly than IoU = 0.50 detection success.

The proposed method delivers the best overall performance and shows clearer separation under AP50:95: its mean gain over the strongest baseline increases from approximately + 1.0 AP50 to approximately + 2.0 AP50:95, highlighting improved bounding-box localization rather than only higher detection confidence.

Sensitivity to randomness

To quantify stochastic variation due to random weight initialization and dataset partitioning, each architecture was retrained using its best hyperparameter setting under five different initialization seeds. The impact of data partitioning was further examined by repeating the procedure with five different split seeds, which alter the composition of the training, validation, and test sets. Results are reported in Table 4 as mean ± standard deviation ($\mu \pm \sigma$), and the corresponding variability is illustrated with violin plots in Fig. 7.

Sensitivity to weight initialization Across five weight-initialization seeds, all architectures show limited dispersion, confirming that the learned detector is largely stable once the hyperparameter configuration is fixed (Table 4). This stability is also apparent in the relatively narrow violin widths in Fig. 7, where the distributions remain compact across backbones. Variability remains small not only for AP metrics (AP50 $\sigma = 0.22–0.28$; AP50:95 $\sigma = 0.24–0.34$) but also for precision/recall ($\sigma \leq 0.29$), indicating consistent convergence behavior. The proposed

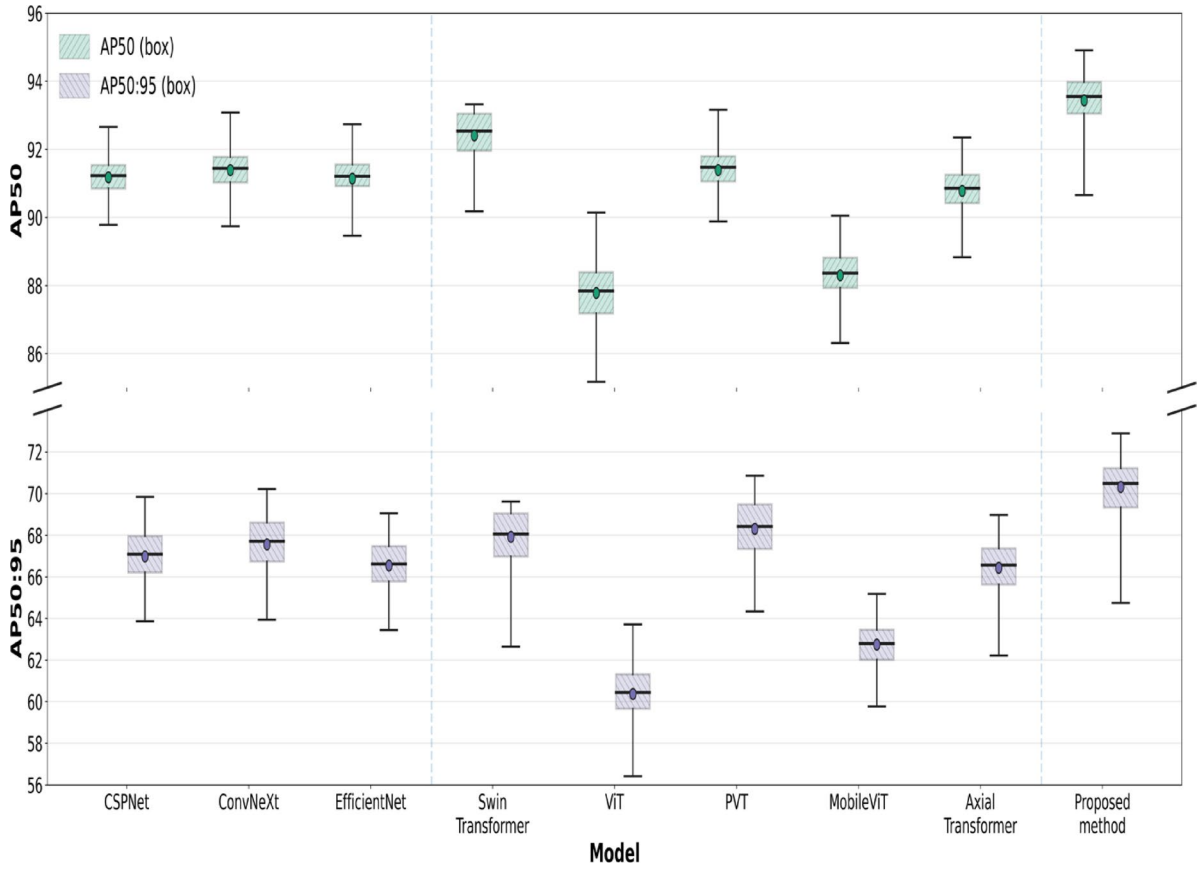


Fig. 6. Performance variability across hyperparameter configurations by architecture (boxplots).

Random factor	Architecture	Precision	Recall	AP50	AP50:95
Weight initialization	CSPNet	94.21 ± 0.16	88.31 ± 0.20	92.41 ± 0.22	68.63 ± 0.25
	ConvNeXt	94.58 ± 0.18	88.70 ± 0.22	92.93 ± 0.25	69.37 ± 0.27
	EfficientNet	94.33 ± 0.15	88.42 ± 0.19	92.56 ± 0.23	68.20 ± 0.24
	Swin Transformer	95.06 ± 0.17	89.46 ± 0.19	93.15 ± 0.26	68.60 ± 0.28
	ViT	91.41 ± 0.29	86.21 ± 0.27	89.96 ± 0.23	62.33 ± 0.34
	PVT	94.66 ± 0.17	88.78 ± 0.18	93.00 ± 0.25	69.63 ± 0.28
	MobileViT	91.62 ± 0.24	86.38 ± 0.25	89.92 ± 0.23	63.68 ± 0.30
	Axial Transformer	94.02 ± 0.19	88.12 ± 0.23	92.21 ± 0.24	67.72 ± 0.26
	Proposed method	96.02 ± 0.20	90.18 ± 0.22	94.73 ± 0.28	70.68 ± 0.30
Dataset split	CSPNet	93.84 ± 0.43	87.81 ± 0.50	92.10 ± 0.44	68.03 ± 0.63
	ConvNeXt	94.18 ± 0.44	88.20 ± 0.52	92.38 ± 0.47	68.63 ± 0.68
	EfficientNet	93.90 ± 0.40	87.96 ± 0.47	91.96 ± 0.43	67.49 ± 0.58
	Swin Transformer	95.10 ± 0.41	89.34 ± 0.45	92.62 ± 0.30	68.40 ± 0.55
	ViT	90.81 ± 0.63	85.42 ± 0.71	89.11 ± 0.64	61.54 ± 0.88
	PVT	94.10 ± 0.46	88.29 ± 0.49	92.41 ± 0.55	68.86 ± 0.61
	MobileViT	91.26 ± 0.51	85.83 ± 0.60	89.06 ± 0.54	62.94 ± 0.69
	Axial Transformer	93.62 ± 0.45	87.66 ± 0.56	91.53 ± 0.48	66.94 ± 0.67
	Proposed method	95.73 ± 0.38	90.02 ± 0.44	94.21 ± 0.41	70.02 ± 0.57

Table 4. YOLOv10 performance across architectural variants over five random seeds (mean ± std).

method achieves the best mean performance (AP50 = 94.73 ± 0.28; AP50:95 = 70.68 ± 0.30) and maintains clear margins over the strongest baselines, with its gains being more pronounced for AP50:95, reflecting improved localization rather than only detection success. Among baselines, PVT provides the strongest localization accuracy (AP50:95 = 69.63 ± 0.28), while Swin attains higher recall (89.46 ± 0.19), suggesting different backbone

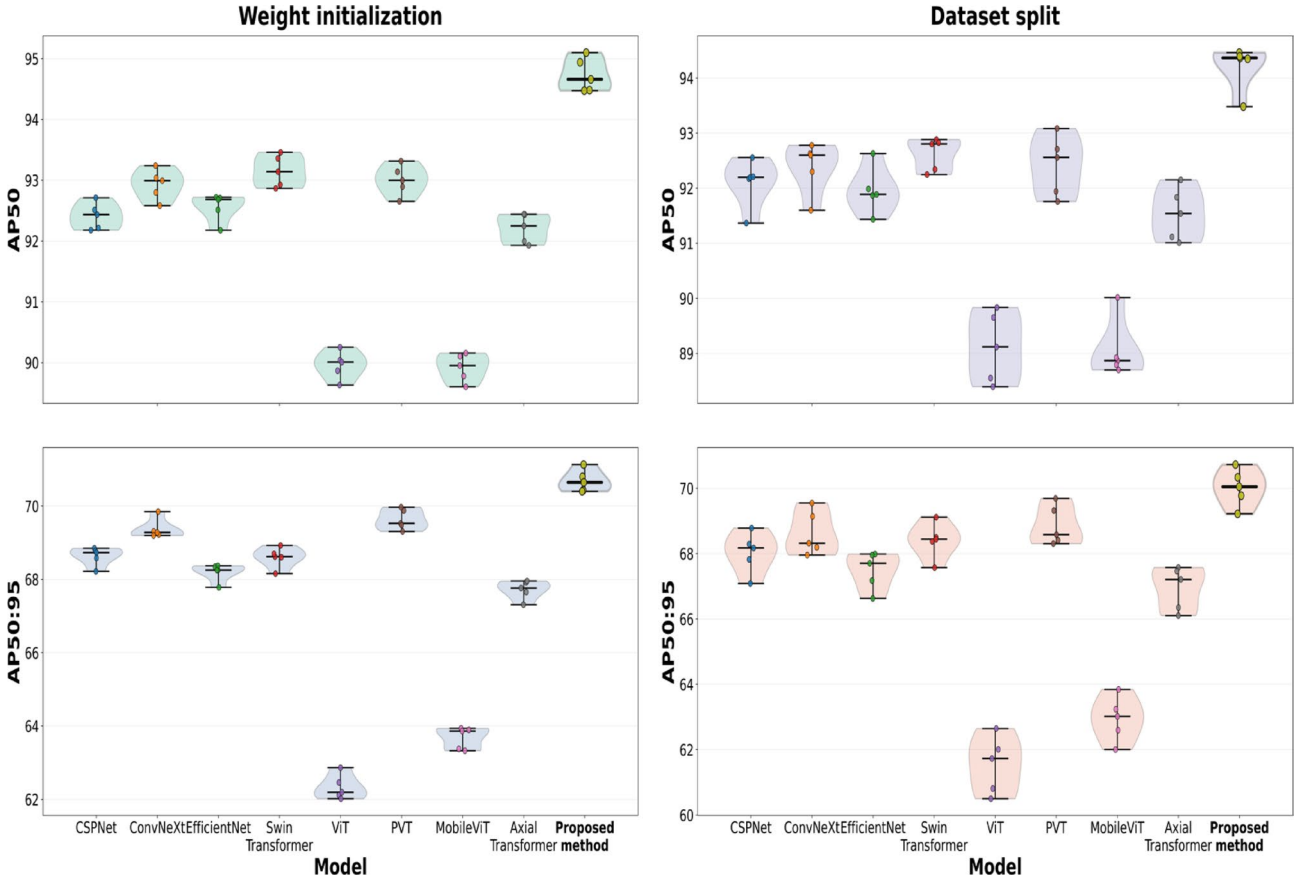


Fig. 7. Model performance across architectural variants under different random seeds.

strengths. In contrast, ViT and MobileViT yield lower means and slightly wider distributions, indicating less reliable optimization under the same training budget.

Sensitivity to data split When the train/validation/test partition is varied, performance variability increases markedly across all architectures, indicating that dataset composition is a dominant factor affecting reproducibility (Table 4). This heightened sensitivity is also evident in Fig. 7, where the violins are wider and the seed-level points are more dispersed than in the weight-initialization setting. Standard deviations approximately double relative to initialization-only runs (AP50 $\sigma = 0.30–0.64$; AP50:95 $\sigma = 0.55–0.88$), and the larger spread in AP50:95 suggests that localization quality is particularly sensitive to which scenes and object instances are included in each split. The effect is most pronounced for ViT (AP50:95 = 61.54 ± 0.88) and MobileViT (62.94 ± 0.69), which also show the broadest violin profiles in Fig. 7, indicating limited robustness to split-induced distribution shifts. Notably, Swin exhibits the smallest AP50 variability ($\sigma = 0.30$), whereas PVT remains the strongest baseline under AP50:95 (68.86 ± 0.61), confirming that backbone ranking can depend on the evaluation metric. Despite split variation, the proposed method remains top-ranked (94.21 ± 0.41 AP50; 70.02 ± 0.57 AP50:95), supporting stable superiority under realistic data uncertainty.

Test set evaluation

Validation-to-test generalization

For deployment-oriented evaluation, the best checkpoint for each architecture was selected based on the highest validation performance, and the resulting best-run outcomes are summarized in Table 5. The proposed method achieves the strongest validation accuracy (AP50 = 95.05; AP50:95 = 71.02) while maintaining practical inference speed (35.18 FPS), yielding a favorable accuracy–efficiency trade-off relative to transformer-based baselines and a clear accuracy gain over faster CNN backbones. Notably, the improvement is most pronounced for small objects, as reflected by higher AP_{ES} and AP_{S} values.

Generalization was further assessed by evaluating the selected checkpoint on the held-out test set (Fig. 8). The proposed model retains top performance on test (AP50 = 94.28; AP50:95 = 69.42) with a modest reduction from validation (ΔAP50 = −0.77; ΔAP50:95 = −1.60). The decrease is concentrated in the extremely small and small bins ($\Delta\,\mathrm{AP}_{ES} = -1.52$; $\Delta\,\mathrm{AP}_{S} = -0.89$), whereas medium and large objects remain comparatively stable ($\Delta\,\mathrm{AP}_{M} = -0.73$; $\Delta\,\mathrm{AP}_{L} = -0.53$). This pattern is expected because small instances are more sensitive to viewpoint changes and localization noise, and it indicates that the proposed method generalizes reliably under realistic field conditions, particularly for the most challenging small-object cases.

Architecture	AP50			AP50:95			Scale				FPS
	mAP	Helmet	No-helmet	mAP	Helmet	No-helmet	AP_{ES}	AP_S	AP_M	AP_L	
CSPNet	92.66	94.84	90.48	68.92	71.02	66.82	82.34	89.41	94.86	96.21	44.73
ConvNeXt	93.22	95.39	91.05	69.68	71.64	67.72	83.12	90.26	95.36	96.58	42.96
EfficientNet	92.82	95	90.64	68.48	70.4	66.56	82.39	89.55	95.02	96.31	40.31
Swin Transformer	93.45	95.62	91.28	68.92	70.83	67.01	84.88	91.18	95.92	96.86	36.88
ViT	90.22	92.38	88.06	62.72	64.6	60.84	78.32	86.52	93.06	94.92	35.52
PVT	93.29	95.46	91.12	69.95	71.88	68.02	84.62	90.96	95.72	96.86	37.39
MobileViT	90.18	92.34	88.02	64.02	65.88	62.16	79.46	87.16	93.62	95.22	39.84
Axial Transformer	92.48	94.64	90.32	68.02	69.92	66.12	82.54	89.58	95.02	96.46	40.92
Proposed method	95.05	97.29	92.81	71.02	73.18	68.86	87.62	92.44	96.68	97.45	35.18

Table 5. Optimal run for each architecture based on the highest validation result.

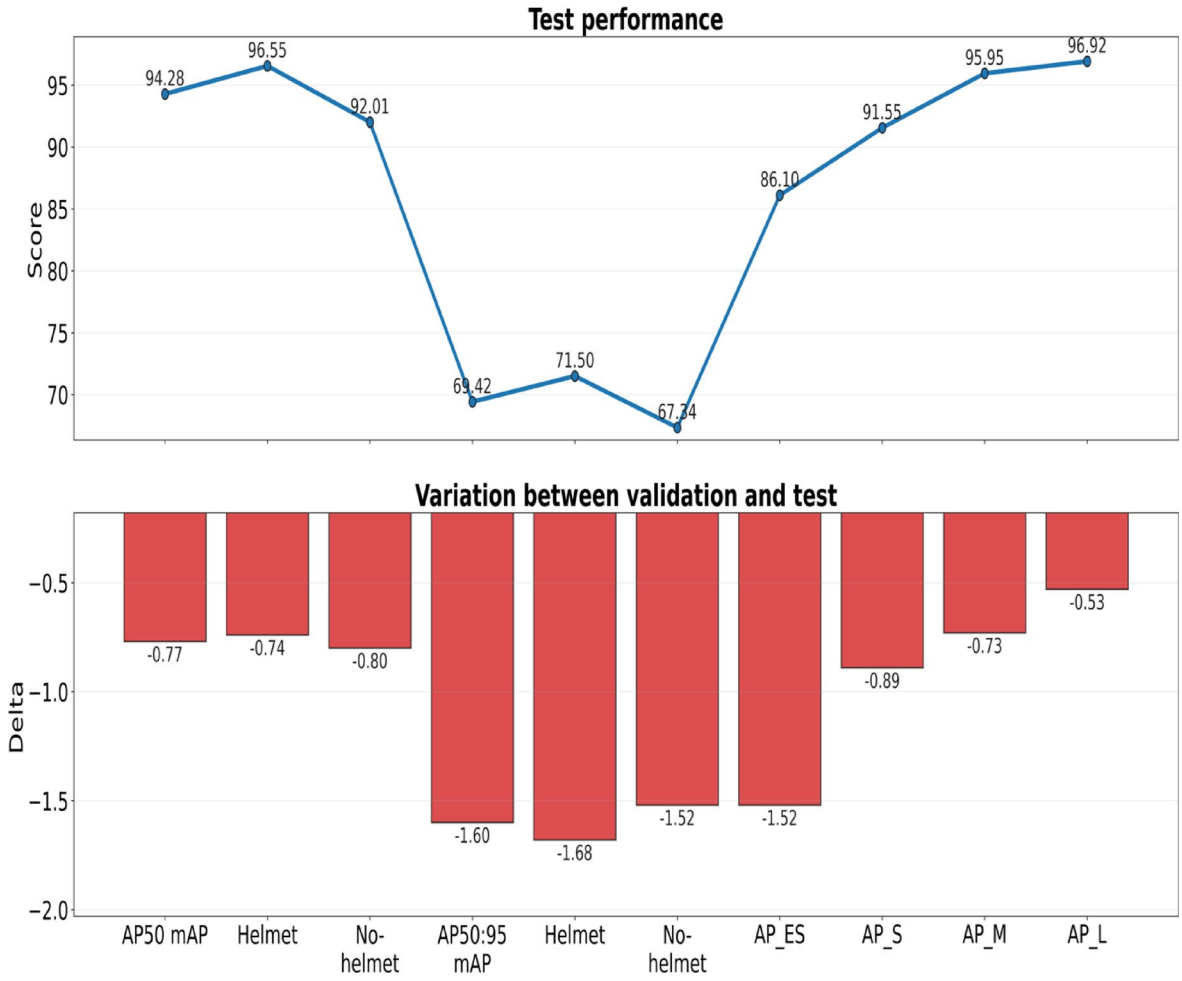


Fig. 8. Generalization performance of the proposed method across validation and test sets.

Effect of image conditions

In this study, test-set performance is stratified by six image-condition factors, with each factor dividing the test set into two subsets. Representative detection examples for each condition are shown in Fig. 9, where textual labels are omitted for clarity. Table 6 reports the corresponding performance results. Because AP is computed by integrating the precision–recall curve over a dataset, AP values calculated for subgroups are not additive and should not be expected to average to the overall test-set AP.

Overall, the performance varies across conditions and consistently declines on the more challenging subsets (low illumination, far-field views, outdoor scenes, overlapping workers, grouped workers, and multiple-worker cases). The drop is typically larger for AP50:95 than for AP50, suggesting that difficult conditions mainly harm



Fig. 9. Sample detection visualizations under multiple scene conditions.

Image condition	Range	AP50			AP50:95			Scale			
		mAP	Safety helmet	No-helmet	mAP	Safety helmet	No-helmet	AP_{ES}	AP_S	AP_M	AP_L
Illumination	High	94.87	97.02	92.72	70.44	72.56	68.32	87.62	92.14	96.22	97.35
	Low	93.33	95.62	91.04	68.58	70.78	66.38	84.71	90.03	95.34	96.98
Visual range	Near-field view	94.79	96.96	92.62	70.33	72.46	68.2	87.41	91.98	96.15	97.28
	Far-field view	93.21	95.54	90.88	68.35	70.62	66.08	84.36	89.62	95.26	96.86
Space	Indoor	94.71	96.92	92.5	70.28	72.4	68.16	87.12	91.72	96.1	97.22
	Outdoor	93.28	95.58	90.98	68.49	70.74	66.24	84.52	89.83	95.31	96.9
Overlap of worker	Non-overlap	94.62	96.78	92.46	70.15	72.22	68.08	86.98	91.54	96.04	97.16
	Overlap	93.12	95.46	90.78	68.21	70.46	65.96	84.2	89.47	95.18	96.79
Density of worker	Scatter	94.36	96.54	92.18	69.92	71.98	67.86	86.66	91.12	95.86	97.01
	Grouped	93.58	95.84	91.32	68.98	71.12	66.84	85.61	90.31	95.54	96.84
Number of workers	Single	94.7	96.88	92.52	70.26	72.34	68.18	87.05	91.63	96.09	97.18
	Multiple	93.17	95.48	90.86	68.36	70.64	66.08	84.4	89.71	95.27	96.86

Table 6. Performance metrics on the test set grouped by image conditions.

Class	Instances	TP	FP	FN	Precision	Recall	FN rate
Helmet	3,991	3,784	152	207	96.14	94.81	5.19
No-helmet	4,092	3,736	289	356	92.82	91.3	8.7

Table 7. Class-specific detection errors on the test set.

Condition	No-helmet GT	No-helmet FN	FN rate
Near-field view	2,150	154	7.16
Far-field view	1,942	202	10.4
High illumination	2,638	200	7.58
Low illumination	1,454	156	10.73
Non-overlap	2,590	196	7.57
Overlap/crowded	1,502	160	10.65
Scattered workers	2,380	184	7.73
Grouped workers	1,712	172	10.05
Single worker	1,406	100	7.11
Multiple workers	2,686	256	9.53

Table 8. Condition-wise false negatives for No-helmet instances.

localization accuracy and bounding-box tightness, rather than merely reducing detection at the looser IoU threshold of 0.50.

Across factors, the largest impact is observed for visual range: moving from near-field to far-field views decreases AP50 from 94.79 to 93.21 (− 1.58) and AP50:95 from 70.33 to 68.35 (− 1.98). This reduction is driven mainly by extremely small objects; AP_{ES} drops from 87.41 to 84.36 (− 3.05), whereas medium and large objects remain relatively stable (AP_{M} −0.89; AP_{L} −0.42). Similar trends appear for worker overlap and outdoor scenes, which also yield clear drops in AP50:95 and AP_{ES}, suggesting that occlusion/crowding and scale-related ambiguity are the dominant failure modes. These results confirm that the proposed method is robust across diverse site conditions, but its limitations are concentrated in far-field and crowded scenarios where small instances are hardest to localize accurately.

Safety-oriented error analysis

Because missing a No-helmet case is more critical than missing a Helmet case in safety monitoring, an object-level error analysis was conducted in addition to AP-based evaluation. A detection was counted as a true positive when the predicted class matched the ground-truth class and IoU was ≥0.50. Unmatched predictions were counted as false positives, and unmatched ground-truth boxes were counted as false negatives.

As shown in Table 7, the proposed model achieved high recall for both classes, but the No-helmet class showed a higher false-negative rate than the Helmet class. Specifically, the No-helmet FN rate was 8.70%, compared with 5.19% for Helmet. This indicates that uncovered heads remain more difficult to detect, mainly because they often provide weaker visual cues than helmets and can be confused with hair, shadows, background objects, or partially occluded head regions.

Table 8 further summarizes No-helmet false negatives under representative difficult conditions. Each pair of conditions is analyzed separately according to the corresponding image-condition factor. The FN rate increases in far-field views, low-illumination scenes, overlapping workers, grouped workers, and multiple-worker scenes. The largest error concentration appears in far-field and crowded cases, where uncovered heads occupy only a few pixels or are partially blocked by other workers and site objects. Because helmet/head regions are often extremely small in the original surveillance images, only representative qualitative examples are included in the manuscript for readability. The complete visualization results, including model-wise detection outputs and representative false-positive and false-negative cases, can be made available through a Figshare repository upon reasonable request.

Site-wise generalization evaluation

To further evaluate whether the proposed model generalizes to unseen construction environments, a site-wise hold-out experiment was conducted. Unlike the original random image-level split, the site-wise protocol assigns all images from the same construction site to only one partition. Therefore, images from the test site are never included in the training or validation sets.

As shown in Table 9, the site-wise hold-out setting produces lower performance than the random image-level split. The proposed model decreases from 94.28% to 89.74% in AP50 and from 69.42% to 64.38% in AP50:95. This drop is expected because the site-wise setting reduces visual overlap between training and test images. The degradation is more pronounced for extremely small and small objects, where.

AP_{ES} decreases by 6.24 points and, AP_{S} decreases by 5.13 points. This result indicates that far-field helmet/no-helmet detection is more sensitive to unseen site conditions than medium- or large-object detection.

Split protocol	AP50	AP50:95	Helmet AP50	No-helmet AP50	AP_{ES}	AP_S	AP_M	AP_L	FPS
Random image split	94.28	69.42	96.55	92.01	86.1	91.55	95.95	96.92	35.18
Site-wise hold-out	89.74	64.38	92.43	87.05	79.86	86.42	92.16	94.18	35.18
Difference	−4.54	−5.04	−4.12	−4.96	−6.24	−5.13	−3.79	−2.74	0

Table 9. Comparison between random image split and site-wise hold-out evaluation.

Model	AP50	AP50:95	Helmet AP50	No-helmet AP50	AP_{ES}	AP_S	AP_M	AP_L	FPS
YOLOv10 baseline	87.96	62.55	90.64	85.28	77.92	84.66	90.84	93.01	37.84
Proposed method	89.74	64.38	92.43	87.05	79.86	86.42	92.16	94.18	35.18
Difference	1.78	1.83	1.79	1.77	1.94	1.76	1.32	1.17	−2.66

Table 10. Site-wise hold-out comparison between YOLOv10 baseline and proposed method.

Variant	1	2	3	4	5	6	7	8	9	10
ODConv	–	✓	–	–	✓	✓	–	✓	–	✓
EMA-BiFPN	–	–	✓	–	✓	–	✓	✓	–	✓
MPDIoU	–	–	–	✓	–	✓	✓	✓	–	✓
Four-scale head	–	–	–	–	–	–	–	–	✓	✓
AP50	93.62	93.92	93.86	93.74	94.06	94.02	94.12	94.18	93.83	94.28
AP50:95	68.28	68.61	68.54	68.88	68.93	69.05	69.24	69.31	68.52	69.42
AP_{ES}	84.26	85.01	85.22	84.4	85.46	85.22	85.72	85.78	85.05	86.1
AP_S	90.22	90.61	90.55	90.36	91.12	90.92	91.28	91.34	90.66	91.55
AP_M	95.12	95.28	95.26	95.18	95.7	95.58	95.82	95.88	95.18	95.95
AP_L	96.18	96.28	96.26	96.22	96.68	96.61	96.76	96.85	96.2	96.92
FPS	37.84	36.31	38.22	37.7	35.92	36.14	37.98	35.72	36.95	35.18
GFLOPs	16.2	17	16.8	16.2	17.7	17	16.8	17.3	16.6	17.7
Params (M)	8.1	8.7	8.5	8.1	9.1	8.7	8.5	8.9	8.4	9.1

Table 11. Component-wise ablation results of the proposed method.

Table 10 compares the proposed method with the YOLOv10 baseline under the same site-wise hold-out protocol. Under this stricter evaluation setting, the proposed method still outperforms the baseline across all accuracy metrics. In particular, the proposed model improves AP_{ES} by 1.94 points and AP_S by 1.76 points, suggesting that the proposed architectural modifications remain beneficial for small-object detection even under unseen site conditions.

Ablation study

Component-wise ablation of the proposed method Table 11 summarizes the component-wise ablation results for ODConv, EMA-BiFPN, MPDIoU, and the four-scale detection head. Compared with the baseline YOLOv10 model, each component improves performance through a different mechanism. ODConv provides the largest single-module gain in AP50 (+0.30) and improves AP_{ES} by +0.75, suggesting stronger extraction of weak small-object features. EMA-BiFPN improves both accuracy and measured throughput, indicating that attention-guided feature fusion can enhance multi-scale representation without excessive computational burden. MPDIoU produces the largest single-module gain in AP50:95 (+0.60), confirming its main role in improving localization quality. The four-scale head shows its clearest benefit for extremely small and small targets: Variant 9 improves AP_{ES} and AP_S by +0.79 and +0.44 over the baseline, respectively. More importantly, comparing Variant 8 with Variant 10 isolates its contribution within the full framework, where AP_{ES} and AP_S further increase by +0.32 and +0.21. Overall, the full model achieves the best accuracy with only a moderate increase in complexity.

Effectiveness of MPDIoU Table 12 compares several IoU-based loss functions under an identical YOLOv10 training setting. The proposed MPDIoU loss is evaluated against GIoU[35], DIoU and CIoU[36], and Efficient IoU (EIoU)[37].

MPDIoU achieves the best performance across all reported metrics, indicating more accurate bounding-box regression and improved localization. In particular, MPDIoU yields the highest AP50 (94.28) and AP50:95 (69.42). Compared with the strongest baseline, EIoU (AP50 = 94.12; AP50:95 = 69.05), MPDIoU improves AP50 by +0.16 and AP50:95 by +0.37, with the larger gain under AP50:95 suggesting that its primary benefit

IoU type	AP50			AP50:95			Scale			
	mAP	Safety helmet	No-helmet	mAP	Safety helmet	No-helmet	AP_{ES}	AP_S	AP_M	AP_L
GIoU	93.96	96.22	91.7	68.72	70.94	66.5	85.22	90.92	95.58	96.61
CIoU	94.06	96.34	91.78	68.88	71.1	66.66	85.46	91.12	95.7	96.68
DIoU	93.74	96.01	91.47	68.56	70.78	66.34	84.93	90.63	95.42	96.49
EIoU	94.12	96.4	91.84	69.05	71.24	66.86	85.72	91.28	95.82	96.76
MPDIoU	94.28	96.55	92.01	69.42	71.5	67.34	86.1	91.55	95.95	96.92

Table 12. Comparison of different IOU-based regression losses.

Dataset name	Number of images	Numbers of safety helmet	Numbers of no-helmet
Safety Helmet Wearing-Dataset (SHWD)[38]	832	2,854	921
Safety Helmet Computer Vision Project (SHCVP)[39]	1,321	2,239	1,625
Safety Helmet Detection (SHD)[40]	1,537	1,733	1,011
Hard Hat Workers Dataset (HHWD)[41]	1,135	1,251	837

Table 13. Description and statistics of the benchmark dataset.

Dataset	AP50			AP50:95			Scale			
	mAP	Safety helmet	No-helmet	mAP	Safety helmet	No-helmet	AP_{ES}	AP_S	AP_M	AP_L
Our dataset	94.28	96.55	92.01	69.42	71.5	67.34	86.1	91.55	95.95	96.92
SHWD	86.24	88.72	83.76	62.58	64.91	60.25	77.12	79.48	82.96	85.31
SHCVP	85.06	87.45	82.67	61.72	63.96	59.48	76.05	78.24	81.83	84.1
SHD	84.48	87.09	81.87	61.13	63.42	58.84	75.41	77.56	81.15	83.44
HHWD	81.92	84.48	79.36	58.61	60.94	56.28	73.04	74.86	78.18	80.61

Table 14. Cross-dataset evaluation results on an alternative benchmark dataset.

is improved localization quality under stricter IoU thresholds. The improvement is most evident for extremely small and small objects: MPDIoU increases AP_{ES} from 85.72 to 86.10 (+0.38) and AP_S from 91.28 to 91.55 (+0.27) relative to EIoU, while medium and large objects show smaller but consistent gains (AP_M +0.13; AP_L +0.16). By contrast, DIoU records the lowest AP50 (93.74) and AP50:95 (68.56), supporting the selection of MPDIoU to enhance small-object detection and bounding-box precision in this dataset.

Benchmark dataset evaluation

Benchmark dataset description The models proposed in this study were trained using surveillance camera imagery, which typically exhibits substantial variability and limited environmental control. As a result, distribution differences between the training data and external benchmark datasets can affect evaluation performance. Notably, many widely used benchmark datasets are not composed exclusively of construction-site imagery; instead, they often include images collected from web search engines (e.g., Google or Baidu), leading to scenes that may not accurately represent real-world construction conditions.

As summarized in Table 13, four commonly adopted benchmark datasets are used to evaluate the developed models. These datasets differ in data sources and scene composition, and some contain settings that are less realistic than genuine construction environments.

Comparative analysis Table 14 reports cross-dataset performance when the model trained on our construction-site dataset is evaluated on four widely used benchmark helmet datasets. As expected, the best results are obtained on the in-domain test set (AP50 = 94.28; AP50:95 = 69.42), whereas all benchmark datasets show substantial performance degradation, indicating a pronounced domain shift between real construction imagery and web-curated benchmarks. Among the external datasets, SHWD yields the highest transfer performance (AP50 = 86.24; AP50:95 = 62.58), followed by SHCVP (85.06/61.72) and SHD (84.48/61.13), while HHWD is the most challenging (81.92/58.61). The degradation is consistently larger for AP50:95 than AP50, suggesting that cross-domain differences primarily affect localization quality and bounding-box precision rather than only coarse detection at IoU = 0.50.

Scale-specific results further confirm this trend: compared with our dataset (AP_{ES} = 86.10; AP_S = 91.55), performance drops sharply on benchmarks, particularly for extremely small and small objects (e.g., AP_{ES} = 73.04–77.12; AP_S = 74.86–79.48), while medium and large objects also decline but remain comparatively

higher. Overall, these results highlight that the proposed model generalizes best to datasets that are visually closer to construction settings, and that improving robustness to domain shift—especially for small-object localization—remains an important direction for deployment across heterogeneous data sources.

Comparison with other detectors

Selection of comparative detectors To place the proposed detector's performance in context, comparative evaluations were carried out against representative two-stage, single-stage, and transformer-based object detection methods. The justification for choosing these baselines, along with their associated backbone architectures, is presented in the section "Comparative methods".

For a fair and consistent implementation, baseline detectors were configured and trained using the MMDetection framework[42], while YOLO-based methods were implemented using the Ultralytics toolkit[43].

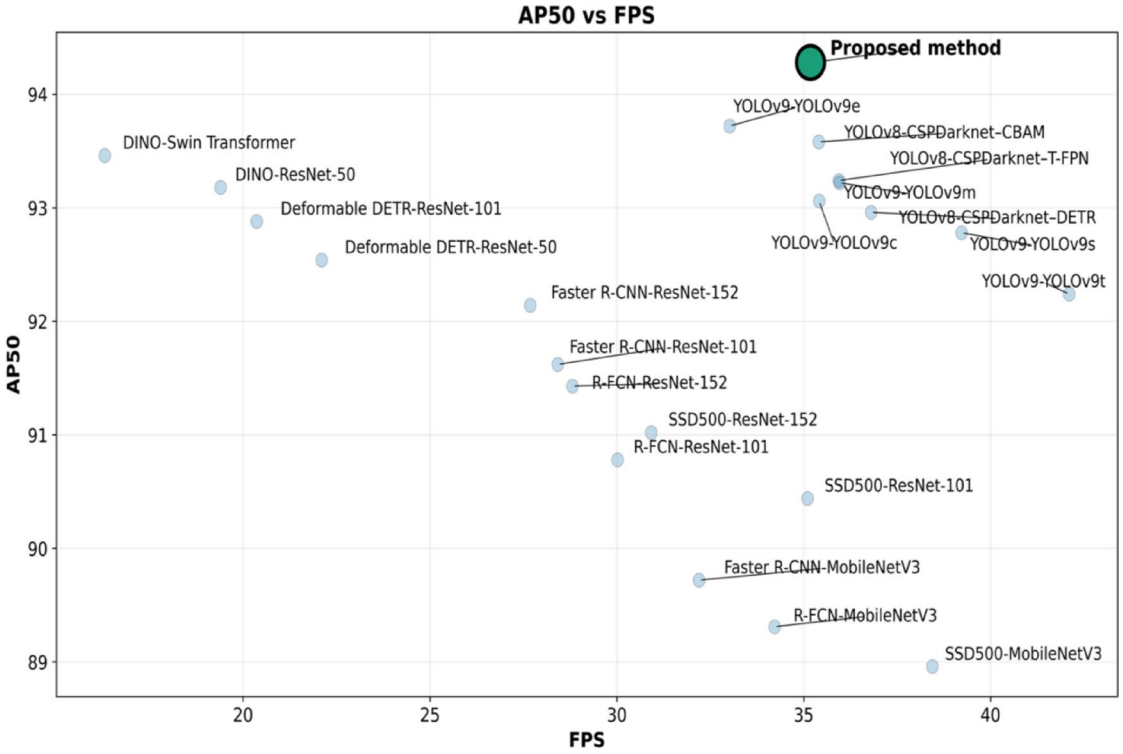
Comparison results Table 15 compares the proposed method with two-stage detectors (Faster R-CNN), classical one-stage detectors (SSD500 and R-FCN), recent YOLO variants (YOLOv8/YOLOv9), and transformer-based detectors (Deformable DETR and DINO) for safety helmet monitoring. Overall, the proposed method achieves the best test-set accuracy, reaching AP50 = 94.28 and AP50:95 = 69.42, while maintaining practical throughput (35.18 FPS). Among the baselines, YOLOv9e is the strongest competitor in terms of accuracy (AP50 = 93.72; AP50:95 = 69.02), but it remains inferior to the proposed method on both metrics and is also slower (33.02 FPS). Transformer-based detectors provide competitive AP50:95 (e.g., DINO with Swin Transformer: 69.12), but their throughput is substantially lower (16.3 FPS), limiting real-time applicability. Deformable DETR variants are also slower (≈ 20–22 FPS) and do not surpass the proposed method in accuracy (AP50:95 ≤ 68.64).

The proposed method's advantage is most evident for extremely small and small instances, achieving the highest $AP_{ES} = 86.10$ and $AP_S = 91.55$. Compared with YOLOv9e, this corresponds to gains of + 1.52 (AP_{ES}) and + 0.43 (AP_S), indicating improved robustness for distant workers or small helmet appearances. Medium and large objects show smaller but consistent improvements ($AP_M = 95.95$, $AP_L = 96.92$), suggesting that performance differences mainly arise under challenging scales rather than easy close-range cases.

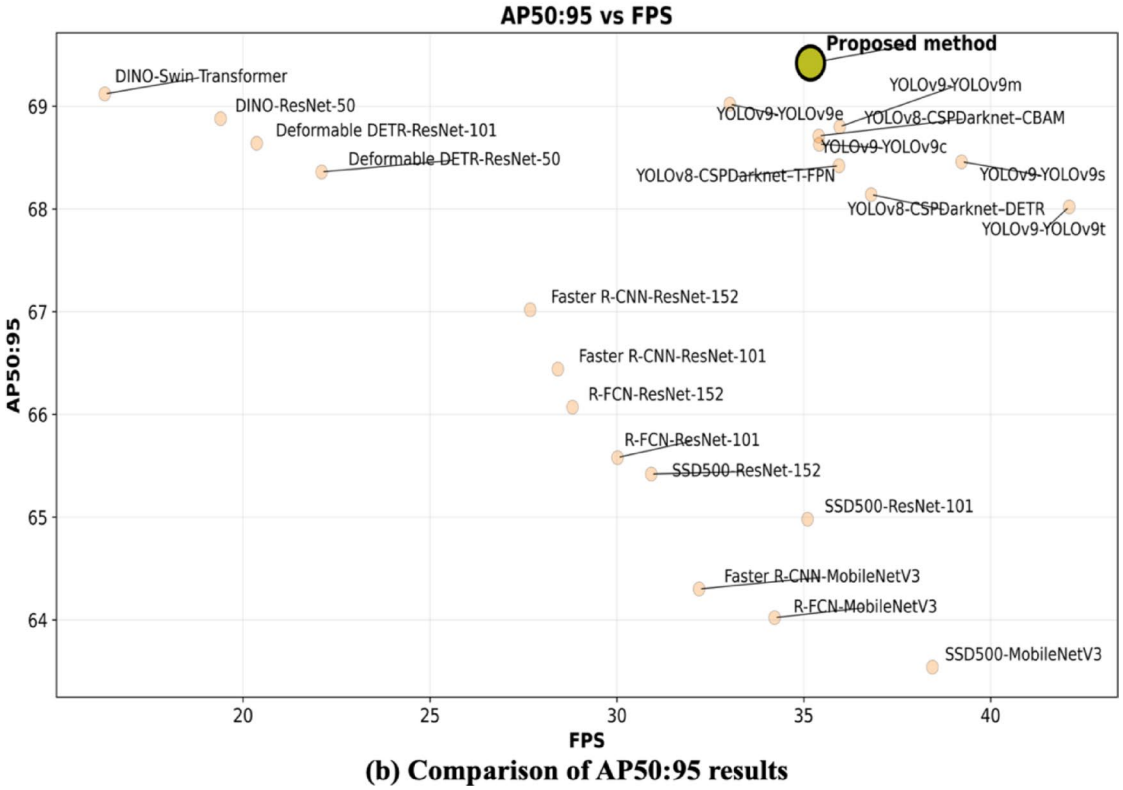
Figure 10 visualizes the accuracy–speed trade-off using AP50 vs. FPS and AP50:95 vs. FPS. The proposed method lies in a favorable region, combining top-tier accuracy with moderate speed. It outperforms faster YOLO variants (e.g., YOLOv9t at 42.1 FPS but AP50:95 = 68.02) in accuracy, while delivering substantially higher throughput than transformer detectors (DINO/Deformable DETR). Compared with two-stage detectors, it provides markedly higher AP with higher or comparable FPS, supporting its suitability for deployment-oriented, real-time helmet compliance monitoring on construction sites.

Detector	Architecture	AP50			AP50:95			Scale				FPS
		mAP	Safety helmet	No-helmet	mAP	Safety helmet	No-helmet	AP_{ES}	AP_S	AP_M	AP_L	
Faster R-CNN	ResNet-101	91.62	93.82	89.42	66.44	68.58	64.3	78.92	87.46	93.12	95.11	28.42
	ResNet-152	92.14	94.26	90.02	67.02	69.14	64.9	79.44	87.96	93.48	95.28	27.68
	MobileNetV3	89.72	91.98	87.46	64.3	66.55	62.05	76.12	84.78	91.62	93.88	32.2
SSD500	ResNet-101	90.44	92.69	88.19	64.98	67.23	62.73	77.26	85.86	92.24	94.22	35.1
	ResNet-152	91.02	93.2	88.84	65.42	67.71	63.13	77.82	86.31	92.58	94.41	30.92
	MobileNetV3	88.96	91.18	86.74	63.54	65.78	61.3	74.92	83.61	90.98	93.12	38.44
R-FCN	ResNet-101	90.78	93.01	88.55	65.58	67.81	63.35	77.68	86.14	92.39	94.46	30.02
	ResNet-152	91.43	93.59	89.27	66.07	68.26	63.88	78.24	86.62	92.74	94.65	28.81
	MobileNetV3	89.31	91.5	87.12	64.02	66.24	61.8	75.68	84.34	91.47	93.69	34.22
YOLOv8	CSPDarknet–DETR	92.96	95.2	90.72	68.14	70.31	65.97	83.62	90.36	94.96	96.21	36.8
	CSPDarknet–T-FPN	93.24	95.5	90.98	68.42	70.61	66.23	83.98	90.72	95.18	96.36	35.94
	CSPDarknet–CBAM	93.58	95.84	91.32	68.71	70.92	66.5	84.44	91.05	95.39	96.48	35.4
YOLOv9	YOLOv9t	92.24	94.46	90.02	68.02	70.18	65.86	83.18	89.88	94.64	95.98	42.1
	YOLOv9s	92.78	95.02	90.54	68.46	70.65	66.27	83.72	90.42	95.01	96.22	39.22
	YOLOv9m	93.22	95.46	90.98	68.8	71.02	66.58	84.16	90.78	95.22	96.37	35.96
	YOLOv9c	93.06	95.28	90.84	68.63	70.84	66.42	83.96	90.66	95.17	96.31	35.41
	YOLOv9e	93.72	95.96	91.48	69.02	71.24	66.8	84.58	91.12	95.48	96.55	33.02
Deformable DETR	ResNet-50	92.54	94.8	90.28	68.36	70.56	66.16	83.44	90.18	95.02	96.28	22.1
	ResNet-101	92.88	95.14	90.62	68.64	70.85	66.43	83.76	90.46	95.18	96.41	20.36
DINO	ResNet-50	93.18	95.4	90.96	68.88	71.09	66.67	84.12	90.72	95.36	96.52	19.4
	Swin Transformer	93.46	95.71	91.21	69.12	71.34	66.9	84.46	90.98	95.54	96.66	16.3
Proposed method		94.28	96.55	92.01	69.42	71.5	67.34	86.1	91.55	95.95	96.92	35.18

Table 15. Comparative test results for safety helmet monitoring using different detector–backbone combinations.



(a) Comparison of AP50 results



(b) Comparison of AP50:95 results

Fig. 10. Pareto comparison of detection accuracy and inference speed: (**a**) AP50 vs. FPS and (**b**) AP50:95 vs. FPS.

Practical applications and deployment considerations

The proposed detector can serve as a decision-support tool for construction-site safety-helmet compliance monitoring. In practical use, images or video frames from fixed surveillance cameras can be processed to identify Helmet and No-helmet cases, helping safety managers prioritize potential non-compliance events instead of manually reviewing all camera feeds. This is particularly useful in wide-area scenes, where distant workers and small head regions are difficult to monitor consistently.

The final model also maintains a practical accuracy–efficiency balance. At an input size of 640 × 640, it requires 17.7 GFLOPs and 9.1 M parameters while achieving 35.18 FPS under the workstation environment described in the section "Hardware and software environment". Although the proposed components increase computational cost compared with the baseline, the increase is moderate relative to the gains for extremely small and small objects.

Several deployment limitations remain. Simultaneous multi-camera monitoring may require frame sampling, batched inference, edge acceleration, or multi-GPU processing. In addition, low illumination, motion blur, far-field views, occlusion, and crowded worker arrangements can reduce helmet/head visibility and increase missed detections. Therefore, the model should assist, rather than replace, safety managers, especially for uncertain No-helmet cases.

Conclusions

This study investigated vision-based safety-helmet compliance monitoring (helmet vs. no-helmet) in wide-area construction surveillance imagery, where helmets frequently appear as extremely small targets under clutter, scale variation, and occlusion. To address these challenges, an enhanced YOLOv10 framework was developed by integrating ODConv for stronger small-target representation, an EMA-BiFPN for improved cross-scale feature fusion, a four-head detection scheme with an additional high-resolution head, and the MPDIoU regression loss to stabilize localization under strict intersection-over-union criteria.

Experiments on a large real-world dataset collected from 42 cameras across seven construction sites (10,896 images) demonstrated consistent gains over strong baselines. The proposed detector achieved AP50 = 94.28 and AP50:95 = 69.42 at 35.18 FPS, outperforming the YOLOv10 baseline and other representative detectors, with the largest improvements concentrated on extremely small and small instances (e.g., AP_{ES} = 86.10 and AP_{S} = 91.55). Robustness analyses showed that performance is largely stable to random weight initialization (AP50 = 94.73 ± 0.28, AP50:95 = 70.68 ± 0.30) but more sensitive to dataset partitioning (AP50 = 94.21 ± 0.41, AP50:95 = 70.02 ± 0.57), indicating that data composition is a key driver of reproducibility. Condition-based evaluation further confirmed that adverse environments primarily degrade localization: for example, near-field to far-field views reduced AP50 by 1.58 and AP50:95 by 1.98, with a larger impact on extremely small objects (AP_{ES} −3.05). Ablation results verified complementary benefits of the proposed modules, and MPDIoU provided the strongest localization improvement among IoU-based losses (e.g., +0.37 AP50:95 over EIoU).

Despite strong in-domain performance, cross-dataset evaluation showed clear domain shift when the model was transferred from real construction-site imagery to web-curated helmet benchmarks, with AP50:95 decreasing to 62.58 on SHWD and 58.61 on HHWD. The largest degradation occurred for small-object localization, indicating that far-field helmet/no-helmet detection remains sensitive to dataset and site differences. Future work should therefore improve robustness across unseen sites, cameras, and visual conditions through domain adaptation, camera-wise evaluation, targeted training for far-field and crowded scenes, model compression, edge acceleration, and temporal information from video.

Although this study compared the proposed method with representative one-stage, two-stage, and transformer-based detectors, benchmarking against the latest YOLO families, such as YOLOv12, YOLOv13, and YOLO26, remains future work. Such comparison requires careful control of implementations, pretrained weights, augmentation, label assignment, optimizers, schedules, and random seeds to ensure fair evaluation.

Data availability

The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Appendix

See Tables 16, 17.

Category	Architecture	Min	25%	Median	Mean	75%	Max	Std
CNN based architectures	CSPNet	89.78	90.85	91.23	91.18	91.54	92.66	0.6
	ConvNeXt	89.74	91.03	91.44	91.39	91.78	93.08	0.64
	EfficientNet	89.46	90.92	91.21	91.14	91.56	92.74	0.61
Transformer based architectures	Swin Transformer	90.18	91.96	92.54	92.41	93.05	93.32	0.91
	ViT	85.17	87.18	87.84	87.78	88.39	90.14	0.86
	PVT	89.88	91.06	91.47	91.39	91.8	93.16	0.69
	MobileViT	86.31	87.93	88.36	88.3	88.82	90.05	0.77
	Axial Transformer	88.83	90.42	90.86	90.78	91.26	92.35	0.74
Proposed method		90.66	93.05	93.55	93.44	93.98	94.91	0.82

Table 16. Summary statistics of model performance measured by AP50.

Category	Architecture	Min	25%	Median	Mean	75%	Max	Std
CNN based architectures	CSPNet	63.86	66.21	67.08	66.98	67.96	69.84	1.36
	ConvNeXt	63.94	66.73	67.7	67.56	68.61	70.22	1.47
	EfficientNet	63.44	65.78	66.61	66.54	67.48	69.05	1.29
Transformer based architectures	Swin Transformer	62.64	66.98	68.06	67.92	69.05	69.62	1.87
	ViT	56.41	59.66	60.44	60.37	61.32	63.71	1.59
	PVT	64.33	67.34	68.42	68.3	69.49	70.86	1.7
	MobileViT	59.77	62.01	62.79	62.74	63.46	65.18	1.21
	Axial Transformer	62.22	65.63	66.56	66.43	67.37	68.98	1.46
Proposed method		64.74	69.33	70.49	70.32	71.22	72.88	1.66

Table 17. Summary statistics of model performance measured by AP50:95.

Received: 23 May 2026; Accepted: 29 June 2026
Published online: 08 July 2026

References

1. Gillman, D. & Waring, C. Modernizing data management at the US Bureau of Labor Statistics, IASSIST Q. (2023). https://doi.org/10.29173/iq1038
2. Fang, W., Ding, L., Luo, H. & Love, P. E. D. Falls from heights: A computer vision-based approach for safety harness detection. *Autom. Constr.* https://doi.org/10.1016/j.autcon.2018.02.018 (2018).
3. Nath, N. D., Behzadan, A. H. & Paal, S. G. Deep learning for site safety: Real-time detection of personal protective equipment. *Autom. Constr.* https://doi.org/10.1016/j.autcon.2020.103085 (2020).
4. Mokayed, H., Alsayed, G., Lodin, F., Hagner, O. & Backe, B. Enhancing Object Detection in Snowy Conditions: Evaluating YOLO v9 Models with Augmentation Techniques. In: *2024 11th Int. Conf. Internet Things Syst. Manag. Secur.*, 198–203 (IEEE, 2024).
5. Wang, S., Eum, I., Park, S. & Kim, J. A labelled dataset for rebar counting inspection on construction sites using unmanned aerial vehicles. *Data Br.* https://doi.org/10.1016/j.dib.2024.110720 (2024).
6. Xu, Z., Jha, N., Mehadi, S. & Mandal, M. Multiscale object detection on complex architectural floor plans. *Autom. Constr.* **165**, 105486. https://doi.org/10.1016/j.autcon.2024.105486 (2024).
7. Yin, Y. & Liu, X. Small target worker detection based on improved YOLOv12 for large construction scenes. *Autom. Constr.* **182**, 106722. https://doi.org/10.1016/j.autcon.2025.106722 (2026).
8. Fang, Q. et al. Detecting non-hardhat-use by a deep learning method from far-field surveillance videos. *Autom. Constr.* https://doi.org/10.1016/j.autcon.2017.09.018 (2018).
9. Gu, Y., Xu, S., Wang, Y. & Shi, L. An advanced deep learning approach for safety helmet wearing detection. In *Proc. – 2019 IEEE Int. Congr. Cybermatics 12th IEEE Int. Conf. Internet Things, 15th IEEE Int. Conf. Green Comput. Commun. 12th IEEE Int. Conf. Cyber, Phys. So.* (2019). https://doi.org/10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00128
10. Yang, W. et al. Safety helmet wearing detection based on an improved YOLOV3 scheme. *Int. J. Innov. Comput. Inf. Control* https://doi.org/10.24507/ijicic.18.03.973 (2022).
11. Yan, D. & Wang, L. Improved YOLOv3 Helmet Detection Algorithm. In *2021 4th Int. Conf. Robot. Control Autom. Eng.* RCAE 2021 (2021). https://doi.org/10.1109/RCAE53607.2021.9638803
12. Shen, J. et al. Detecting safety helmet wearing on construction sites with bounding-box regression and deep transfer learning. *Comput. Civ. Infrastruct. Eng.* https://doi.org/10.1111/mice.12579 (2021).
13. Wang, Z. et al. Fast personal protective equipment detection for real construction sites using deep learning approaches. *Sensors* https://doi.org/10.3390/s21103478 (2021).
14. Lee, Y. R., Jung, S. H., Kang, K. S., Ryu, H. C. & Ryu, H. G. Deep learning-based framework for monitoring wearing personal protective equipment on construction sites. *J. Comput. Des. Eng.* https://doi.org/10.1093/jcde/qwad019 (2023).
15. Nguyen, N. T., Tran, Q., Dao, C. H., Nguyen, D. A. & Tran, D. H. Automatic detection of personal protective equipment in construction sites using metaheuristic optimized YOLOv5. *Arab. J. Sci. Eng.* https://doi.org/10.1007/s13369-023-08700-0 (2024).
16. Park, S., Kim, J., Wang, S. & Kim, J. Effectiveness of image augmentation techniques on non-protective personal equipment detection using YOLOv8. *Appl. Sci.* https://doi.org/10.3390/app15052631 (2025).
17. Wang, S., Park, S., Kim, J. & Kim, J. Safety helmet monitoring on construction sites using YOLOv10 and Advanced transformer architectures with surveillance and body-worn cameras. *J. Constr. Eng. Manag* https://doi.org/10.1061/JCEMD4/COENG-16760 (2025).
18. Wang, S. Automated non-PPE detection on construction sites using YOLOv10 and transformer architectures for surveillance and body worn cameras with benchmark datasets. *Sci. Rep.* **15**27043. https://doi.org/10.1038/s41598-025-12468-8 (2025).
19. Wang, S. Domain-adaptive faster R-CNN for non-PPE identification on construction sites from body-worn and general images. *Sci. Rep.* https://doi.org/10.1038/s41598-026-35148-7 (2026).
20. Wang, S., Kim, H., Yeo, J. & Kim, J. YOLOv10-based multi-scale variant object detection for multi-category PPE non-compliance monitoring on construction sites. *Sci. Rep.* https://doi.org/10.1038/s41598-026-53052-y (2026).
21. Hua, W. & Chen, Q. A survey of small object detection based on deep learning in aerial images. *Artif. Intell. Rev.* **58**, 162. https://doi.org/10.1007/s10462-025-11150-9 (2025).
22. Chao, M. et al. A lightweight small object detection model for UAV images based on deep semantic integration. *Sci. Rep.* **15**31888. https://doi.org/10.1038/s41598-025-16878-6 (2025).
23. He, L. & Wei, H. CBAM-YOLOv5: A promising network model for wear particle recognition. *Wirel. Commun. Mob. Comput.* **2023**, 2520933 (2023).
24. Su, K. et al. N-IoU: better IoU-based bounding box regression loss for object detection. *Neural Comput. Appl.* **36**, 3049–3063. https://doi.org/10.1007/s00521-023-09133-4 (2024).
25. Zou, Y. & Liu, S. Small object detection algorithm based on improved YOLOv10 for traffic sign. *Transp. Res. Interdiscip Perspect.* **32**, 101501. https://doi.org/10.1016/j.trip.2025.101501 (2025).
26. Yang, B., Bender, G., Le, Q. V. & Ngiam, J. Condconv: Conditionally parameterized convolutions for efficient inference. In *Adv. Neural Inf. Process. Syst.* 32 (2019).
27. Ouyang, D. et al. Efficient multi-scale attention module with cross-spatial learning. In *ICASSP 2023–2023 IEEE Int. Conf. Acoust. Speech Signal Process.*, 1–5 (IEEE, 2023).

28. Wang, S. Domain adaptation using transformer models for automated detection of exterior cladding materials in street view images. *Sci. Rep.* **16**2696. https://doi.org/10.1038/s41598-025-32524-7 (2025).
29. Zheng, Z. et al. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **52**, 8574–8586 (2021).
30. Ma, S. & Xu, Y. Mpdiou: a loss for efficient and accurate bounding box regression. ArXiv Prepr. ArXiv2307.07662 (2023).
31. Wang, S. Empirical comparison of sequence-aware augmentation strategies for real operational AHU AFDD under data scarcity and class imbalance. *Expert Syst. Appl.* **326**, 132818. https://doi.org/10.1016/j.eswa.2026.132818 (2026).
32. Zoph, B. et al. Rethinking pre-training and self-training. *Adv. Neural Inf. Process. Syst.* **33**, 3833–3845 (2020).
33. Wang, S. Graph neural network–driven text classification for fire-door defect inspection in pre-completion construction. *Sci. Rep.* **15**, 44382. https://doi.org/10.1038/s41598-025-28120-4 (2025).
34. Eum, I., Kim, J., Wang, S. & Kim, J. Heavy equipment detection on construction sites using you only look once (YOLO-version 10) with transformer architectures. *Appl. Sci.* https://doi.org/10.3390/app15052320 (2025).
35. Rezatofighi, H. et al. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 658–666 (2019).
36. Zheng, Z. et al. Distance-IoU loss: Faster and better learning for bounding box regression. In *Proc. AAAI Conf. Artif. Intell.*, 12993–13000 (2020).
37. Zhang, Y. F. et al. Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* **506**, 146–157 (2022).
38. Safety Helmet Wearing-Dataset (accessed December 4, 2024). https://github.com/njvisionpower/Safety-Helmet-Wearing-Dataset?tab=readme-ov-file (2019).
39. S. H. C. V. Project, Safety Helmet Computer Vision Project (accessed December 4, 2024). https://universe.roboflow.com/digitalimage-114q6/safety-helmet-q3b8o (2024).
40. Safety Helmet Detection. (accessed December 4, 2024). https://www.kaggle.com/datasets/andrewmvd/hard-hat-detection (2020).
41. Hard Hat Workers Dataset. (accessed December 4, 2024). https://public.roboflow.com/object-detection/hard-hat-workers (2022).
42. Chen, K. et al. MMDetection: Open MMLab Detection Toolbox and Benchmark. http://arxiv.org/abs/1906.07155 (2019).
43. Rath, S. YOLOv8 Ultralytics: State-of-the-Art YOLO Models. https://Learnopencv.Com/Ultralytics-Yolov8/ (2023).

Author contributions

Seunghyeon Wang: Formal analysis; Methodology; Project administration; Validation; Visualization; Writing – original draft; Writing – review and editing. En-Lian Zhang: Data curation; Visualization. Rong-Lu Hong: Project administration; Resources; Validation; Writing – original draft. Juhyung Kim: Funding acquisition; Data curation; Formal analysis; Investigation; Supervision.

Funding

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant RS-2026-25540027).

Declarations

Competing interests

The authors declare no competing interests.

Ethical approval

This study used an existing construction-site surveillance-camera image dataset that had been established and used in previous studies. No new images or videos were collected specifically for this work, and no direct contact with human participants occurred. The dataset was used only for safety helmet and no-helmet object detection, and no personally identifiable information was analyzed. The publicly accessible benchmark datasets were used in accordance with their respective licenses and terms of use.

Additional information

Correspondence and requests for materials should be addressed to J.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2026