Computer Fraud and Security
ISSN (online): 1873-7056

MarkYolo: An Enhanced YOLOv10 Network with Dynamic Convolution and Attention Mechanism for Circular Marker Detection in High-Speed Video Measurement

Ziqi Zhang[1], Zhonghua Hong[1*]
[1]the College of Information Technology, Shanghai Ocean University, Shanghai, China
Corresponding Author's Email: zhhong@shou.edu.cn

Abstract: In high-speed video measurement, accurate detection of circular markers is critical for applications in structural analysis, motion tracking, and industrial automation. Traditional marker detection methods often struggle with challenges such as dynamic occlusion, complex backgrounds, and scale variations. To address these issues, this paper proposes MarkYolo, an enhanced object detection framework based on YOLOv10 tailored for robust circular marker detection. Key innovations include: (1) Omni-dimensional Dynamic Convolution (ODConv) integrated into a novel COD module to capture multi-dimensional contextual features while reducing computational complexity; (2) an Adaptive Fine-Grained Channel Attention (AFGCAttention) mechanism to enhance small object localization by adaptively fusing global and local information; and (3) Normalized Wasserstein Distance (NWD) loss to improve robustness against positional shifts and scale variations by modeling bounding boxes as Gaussian distributions. Experiments on the CME dataset demonstrate that MarkYolo achieves a state-of-the-art AP50-95 of 75.4%, outperforming the baseline YOLOv10 by 4.9% while maintaining real-time efficiency. The model also reduces false positives and missed detections in complex scenarios, offering significant advancements for high-speed photogrammetry applications. Further ablation studies validate the synergistic contributions of each proposed module, highlighting improvements in recall (96.4%), precision (98.3%), and computational efficiency (8.6 GFLOPs). This work provides a practical solution for enhancing marker detection accuracy in dynamic environments and lays a foundation for future lightweight deployments on edge devices.

Keywords: Circular marker detection; high-speed video measurement; YOLOv10; dynamic convolution; attention mechanism; Normalized Wasserstein Distance.

1. Introduction

With the rapid advancement of computer and photogrammetry technologies, high-speed video measurement [1-3] has become an economically efficient technique that captures and processes images to collect spatial information about objects in detail. This technology is widely applied in fields such as civil engineering [4-6], structural inspection and evaluation [7-9], 3D model creation [10-12], and environmental studies [13-15] due to its precision, non-intrusiveness, and non-destructiveness. It also plays a crucial role in various domains like industrial automation [16], mechanical analysis [17-18], and motion measurement [19-20]. To ensure accuracy, artificial markers [21-22] play a pivotal role in tracking the spatial movement of the objects under study. Automating and accurately identifying these markers, which indicate specific points, is crucial for subsequent tasks such as precise camera calibration [23] and displacement tracking [24].

In high-speed video imagery, factors such as background noise, lighting condition changes, and foreground dynamic occlusion interfere, making existing circular marker recognition methods [25-27] often fall short of ideal results. Traditional methods rely on manually designed features and empirical thresholds, which make them less robust in the face of different scenarios and conditions. Given these challenges, there is an urgent need for innovative marker detection algorithms that maintain robust detection accuracy in cluttered visual scenes while demonstrating dynamic environmental adaptability.

Circular marker detection algorithms in high-speed video measurement are mainly divided into two categories: traditional methods based on specific features and classifiers, and deep learning-based methods. Traditional methods rely on manually designed features, such as Haar feature cascade classifiers [28], which use Haar-like features and the AdaBoost algorithm for efficient detection; SIFT [29] algorithm, which ensures scale,

rotation, and lighting invariance by extracting key points and generating feature descriptors; and DPM (Deformable Part Models) [30], which improves accuracy by modeling the parts of the object and their relationships, although it involves higher computational complexity. Additionally, while image segmentation approaches incorporating background modeling frameworks effectively isolate target regions, their performance tends to degrade significantly under conditions of environmental complexity and temporal scene variations. While traditional methods perform well in simple scenes and are technologically mature, their feature expression capabilities are limited and unable to handle complex scenarios, making deep learning methods increasingly popular.

As deep learning techniques continue to mature, they offer fresh insights into the detection of circular markers. Deep neural networks, especially Convolutional Neural Networks (CNN) [31], can automatically learn features from large labeled datasets, showing stronger feature extraction capabilities and target extraction precision. Widely adopted two-stage object detection methods encompass R-CNN [32], Fast R-CNN [33], Faster R-CNN [34], and Mask R-CNN [35]. These region-focused algorithms begin by generating potential bounding regions that may harbor objects. Then, a classification network is applied to determine the object class in each candidate region. Unlike the above networks, YOLO [36-38] is a one-stage regression-based detection method, where the object detection problem is transformed into a regression task, predicting both the object class and the bounding box in a single forward pass. Due to its efficient design, YOLO is highly suitable for real-time video detection tasks, and its speed advantage has made it widely adopted in practical detection tasks.

In the initial phase of object detection studies, conventional techniques produced some promising outcomes, yet their drawbacks have become increasingly obvious. Consider, for instance, the Haar feature cascade classifier: despite its simple algorithm structure and high efficiency, it has limited feature expression ability, capturing only simple edges and textures, and is unable to describe the complex appearance of objects. Moreover, Haar features are less adaptable to object rotation and scale variations, typically requiring multi-scale sliding windows to detect objects of different sizes, and often underperform in cases of object deformation or occlusion. While SIFT shows great promise in feature extraction, it may fail to extract enough key points in areas with few textures, leading to feature matching failure, and its performance significantly declines in scenarios with large lighting variations or complex backgrounds, especially in crowded environments. DPM (Deformable Part Model) can capture structural variations of objects, however, this approach necessitates detecting each object component independently and then combining them under geometric constraints, which imposes a substantial computational burden and hampers real-time performance. In essence, conventional object detection approaches exhibit several major limitations, which can be outlined as follows: manually designed features are unable to fully capture the complex shape of objects; detection performance significantly degrades under lighting changes or in complex backgrounds; many methods have high computational complexity and cannot meet real-time detection needs; and detection performance is suboptimal under changes in object scale, posture, or partial occlusion.

Because deep learning approaches can capture both low-level and high-level features, they have been extensively employed in image classification [39], object detection [40], and image segmentation [41]. For instance, Deep learning architectures employing convolutional neural networks can autonomously learn intricate, high-dimensional representations, thereby notably boosting the accuracy of circular marker recognition across various scales and lighting environments. Moreover, these methods generally demonstrate heightened robustness, effectively adapting to illumination changes and background distractions, making them especially well-suited for dynamic scenes and video-based surveillance. Owing to these strengths, deep learning has emerged as the primary solution for circular marker detection. R-CNN, a region-based convolutional neural network, employs selective search to propose candidate regions, then extracts features and classifies them through CNN. Though it achieves strong detection accuracy, the method's substantial computational overhead and limited real-time efficiency reduce its practical utility. To enhance detection speed and real-time capability, Faster R-CNN introduces a Region Proposal Network (RPN), though it still falls short of meeting high-frame-rate demands. The Single Shot MultiBox Detector (SSD) [42] conducts detection on multi-scale feature maps, balancing speed with accuracy and excelling at identifying small objects. However, its performance deteriorates in more complex backgrounds. By integrating Focal Loss, RetinaNet [43] tackles detection accuracy issues by mitigating the imbalance between foreground and background samples, thus boosting small-object detection. Despite these benefits, its computational load remains high, which impedes real-time performance. YOLO algorithm, due to its end-to-end detection design, has gained widespread attention. YOLO partitions the input into a grid of cells, each of which

simultaneously predicts bounding boxes and object classes, delivering a substantial boost in detection speed. For example, the MOD-YOLO algorithm [44] proposed by Peng et al. enhances multi-scale feature perception and key information capturing through the introduction of MODSConv modules, GRF-SPP-Fast multi-scale fusion structure, and DAF-CA dual-attention mechanisms. However, this model faces challenges in low-contrast detection scenarios with complex background interference, resulting in missed detections due to weak features of small objects. Additionally, the EL-YOLO algorithm [45] introduced by Yang et al. significantly improves the detection accuracy and model lightweight level of marine targets by optimizing the AWIoU loss function, SMFN multi-level feature fusion, and GDFP pruning strategy. While this method improves the robustness of small target detection in complex conditions, it still faces asymmetric computational costs, making it difficult to meet inference latency requirements when processing high-resolution images. Currently, efforts in YOLO-based approaches concentrate on boosting detection accuracy and real-time performance by incorporating more advanced feature extraction networks and attention mechanisms, significantly enhancing the detection of small objects, particularly in complex conditions.

Currently, research on YOLO-based approaches centers on boosting detection precision and real-time efficiency by incorporating deeper feature extraction frameworks and attention mechanisms, thereby enhancing the detection of small targets, particularly in challenging scenarios. Building on these advancements, we present MarkYolo, a YOLOv10-based circular marker detection network specifically designed to tackle low accuracy, missed detections, and false alarms in high-speed photogrammetry. Our method first introduces Omni-dimensional Dynamic Convolution (ODConv) [46], which expands standard convolution operations to capture extensive contextual information across multiple dimensions, reducing both computational overhead and parameter counts while preserving robust feature extraction. Additionally, we develop the C2f_ODConv (COD) module to refine feature fusion, further elevating the network's detection capabilities and adaptability to diverse environments. To suit the circular marker detection task, we add a small object detection layer to streamline the deep network, improving the detection performance of small objects while reducing the network's parameter size and detection efficiency. Next, we integrate the Adaptive Fine-Grained Channel Attention (AFGCAttention) module [47], an attention mechanism that combines global and local information and adaptively adjusts channel weights to optimize feature maps. This results in a more accurate output that improves the target's recognizability and detection accuracy. Lastly, we employ the Normalized Wasserstein Distance (NWD) [48] loss to handle bounding box predictions, enhancing the model's adaptability to targets of varying scales and boosting robustness in shifting conditions. Our contributions are summarized as follows:

- Introducing Omni-dimensional Dynamic Convolution (ODConv): By implementing ODConv in the YOLOv10 object detection algorithm, we significantly enhance the model's feature extraction capabilities. The multi-dimensional dynamic attention mechanism reduces computational complexity and parameter count while improving real-time detection efficiency. This method improves target detection performance while maintaining efficiency.
- Improving Feature Extraction Module Design: By constructing the COD module and introducing a multi-dimensional attention mechanism, we can reduce the parameter count without significantly increasing the computational burden. This improvement enhances the network's ability to extract features while maintaining low computational overhead.
- Integrating AFGCAttention: This mechanism effectively addresses the complexity of target backgrounds and the lack of local details. It heightens the model's sensitivity to smaller targets and fine-grained features, enabling more precise localization of circular markers in complex scenes. This reduces false alarms and missed detections, ultimately boosting overall accuracy.
- Designing the NWD Loss Function: The NWD loss function, used for bounding box calculation, models bounding boxes as 2D Gaussian distributions and computes the Wasserstein distance between two distributions. This method helps the network focus more on the shape and location of the target, rather than just the bounding box size, resulting in more accurate similarity measurements.
- Experimental Validation and Performance Improvement: Experiments on the CME dataset indicate that MarkYOLO achieves an AP50-95 of 75.4%, marking a 4.9% increase compared to the baseline YOLOv10. This enhancement notably strengthens the detection of small objects and effectively tackles false positives and missed detections in complex scenarios.

2. Datasets

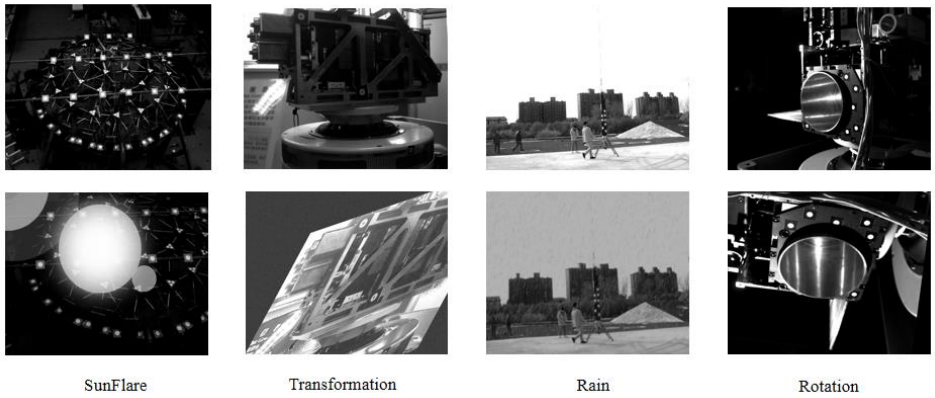

Figure. 1 Circular Marker Data and Data Augmentation Methods.

In this study, the target markers feature a black square backing, a white circular pattern, and a black cross, placed at designated monitoring points on the test object. The dataset consists of circular marker images obtained from a variety of experimental settings, including seismic-resistant structural models, rotating marker mechanisms, and frame-type vibrating screen prototypes. To ensure the proposed method's effectiveness under diverse conditions, data were gathered under multiple illumination levels and experimental scenarios. In total, 3568 raw circular marker images—covering various marker sizes and angles—were collected to form the experimental dataset.

For training the object detection model, the dataset was partitioned into training, testing, and validation sets in a 7:1.5:1.5 ratio. Subsequently, as illustrated in Figure 1, offline augmentation was applied to the training images, employing a range of geometric transformations (scaling, translation, rotation, cropping, and perspective warping) as well as color or noise manipulations (Gaussian noise, brightness/contrast modifications, and simulated rain/snow) to reinforce the model's robustness. Ultimately, 1632 augmented images were appended to the training set, producing the CME dataset.

3. Methodology

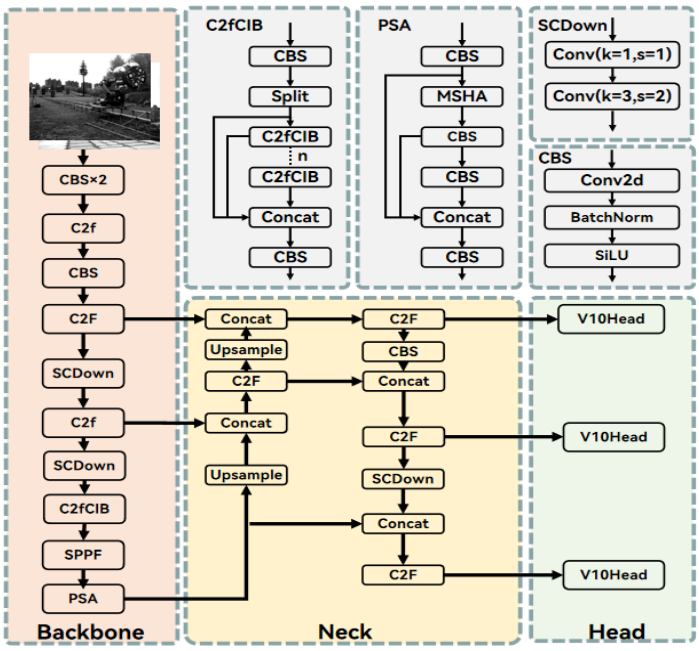

Figure. 2 YOLOv10 Architecture Diagram.

YOLOv10, regarded as a next-generation single-stage detection framework, achieves a truly end-to-end pipeline by integrating an NMS-Free design. Its model architecture is illustrated in Figure 2. YOLOv10 optimizes the Backbone, Neck, and Head of the YOLO model through multiple strategies. The intrinsic rank of a module is used to measure redundancy – the lower the rank, the lower the parameter utilization and the higher the computational redundancy. Therefore, modules with lower intrinsic ranks are replaced with a lightweight Compact Inverted Block (CIB) optimized with large-kernel convolutions, which maintains performance while reducing computational costs and improving parameter utilization.Additionally, YOLOv10 proposes a Spatially-Channel Decoupled Downsampling method (SCDown). This method decouples the traditional downsampling process into two independent steps: spatial downsampling and channel transformation, which maximizes information retention while reducing detection latency. In deeper networks, Partial Self-Attention (PSA) is introduced, applying the self-attention mechanism only to a subset of channels, balancing global modeling capability with computational cost, and enhancing the model's ability to detect small objects and complex scenes.To achieve real-time end-to-end network functionality, YOLOv10 adopts a dual label assignment and consistency matching strategy in the detection head. The dual label assignment introduces a dual-branch detection head, employing a consistent dual assignment strategy: during training, both one-to-many and one-to-one branches are used simultaneously. The former optimizes the model with rich supervisory signals, while the latter directly outputs non-redundant predictions without the need for NMS. This design not only eliminates traditional YOLO's dependency on NMS for post-processing but also significantly improves detection efficiency.

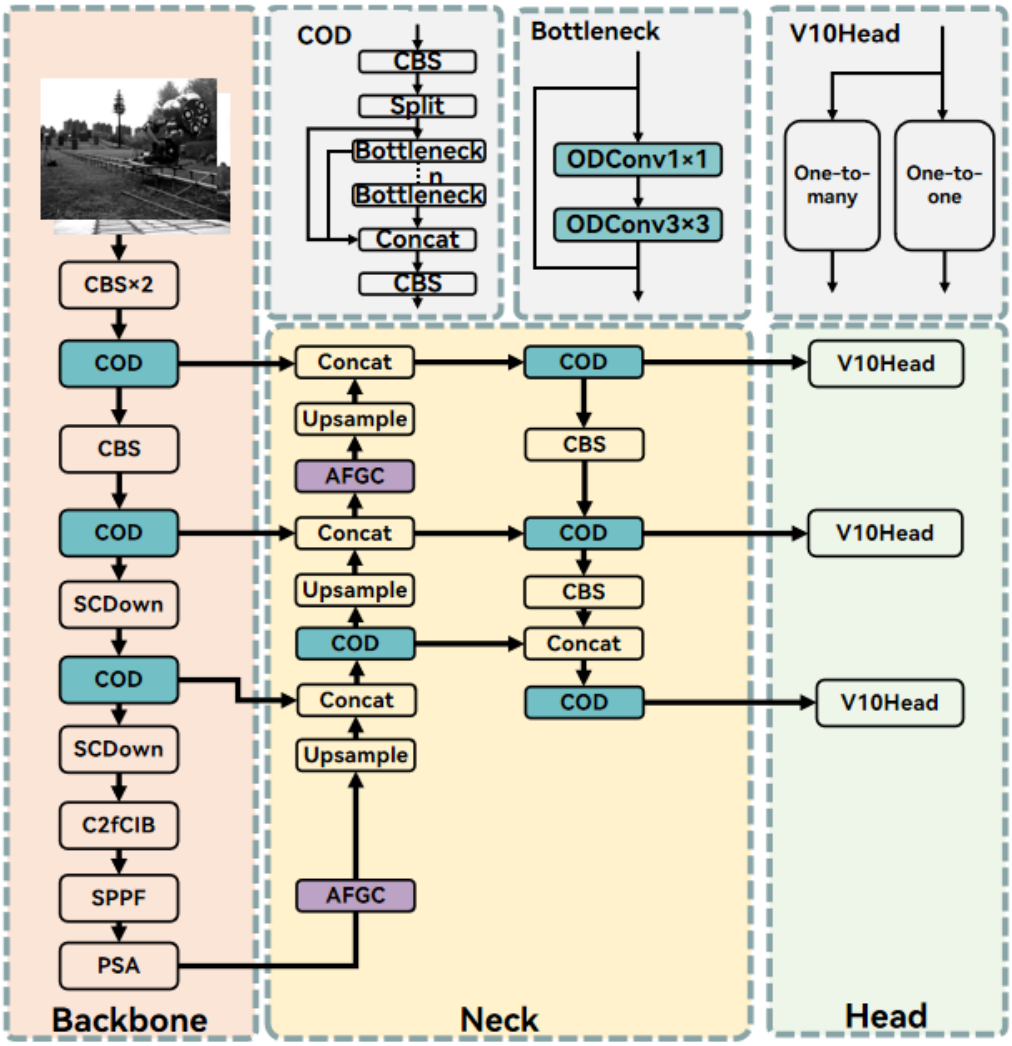

Figure. 3 Architecture of MarkYolo.

The network structure of the MarkYolo detection algorithm is shown in Figure 3. First, taking Yolov10n as the backbone network, in order to detect targets of different scales in an image, this study analyzed the pixel scales of the targets in the CME dataset. Based on the degree of feature map matching, a P2 layer was added to the original P3, P4, and P5 feature layers in the backbone network to detect small-scale targets in the image. Next, to effectively integrate global and local information and optimize the feature enhancement model's ability to perceive small targets and details, the AFGCAttention module—which combines a global information extraction module, a local information interaction module, and an adaptive fusion module—was adopted. Finally, NWD loss was used as the bounding box loss to provide a more stable and reliable measure for small-target detection and reduce the interference of background noise.

3.1 Network Architecture Improvements

3.1.1 Small Target Detection Layer

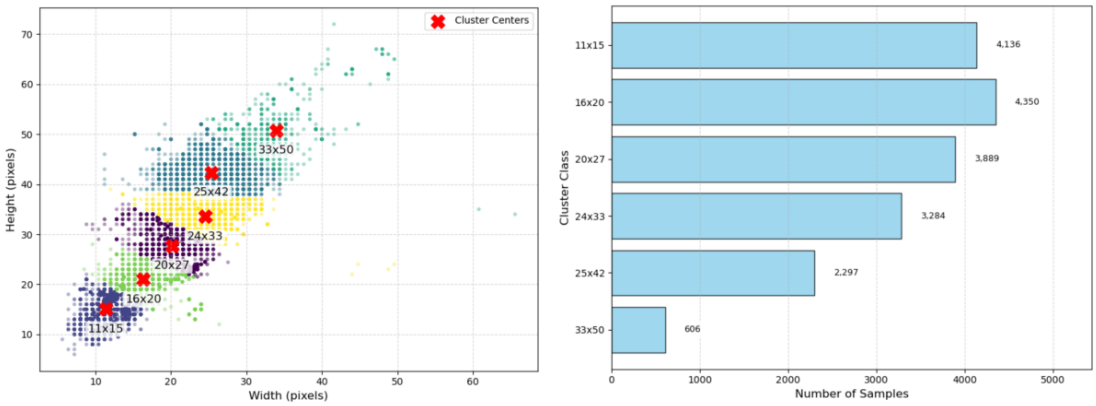

Figure. 4 K-means Clustering Analysis Results. (a) The cluster centers identified through k-means are 11×15, 16×20, 20×27, 24×33, 25×42, and 33×50. Each point on the scatter plot represents a circular landmark target, where the x-axis and y-axis denote the target's width and height, respectively. (b) The distribution of the data points among these cluster centers.

Table 1 Clustering Sample Results

Cluster Size	Sample Proportion	Feature Map Layer (Default)	Target Feature Size (Pixels)
11x15	24.7%	P3（1/8 Downsampling）	1.38x1.88
16x20	23.5%	P3（1/8 Downsampling）	2.50x3.38

In deep convolutional neural networks, the breadth of the receptive field is pivotal for visual tasks [49] because the network's output must cover a sufficiently large portion of the image to effectively capture information from larger objects. Research has found that the effective size of the receptive field follows a Gaussian distribution and occupies only a portion of the theoretical receptive field. The receptive field scale can be defined as:

$$R_k = 1 + \sum_{j=1}^{k} (F_j - 1) \prod_{i=0}^{j-1} S_i \tag{1}$$

Here, R_k represents the receptive field size of the k-th convolutional layer, F_j denotes the kernel size of the j-th convolutional layer, and S_i is the stride of the i-th convolutional layer. Shallow network feature maps effectively preserve detailed features such as edge textures and local deformations of targets, making them particularly suitable for tiny object detection. In contrast, deep network feature maps dilute the feature responses of small targets due to their larger receptive fields, which integrate global contextual information. To identify the network layers most suitable for detecting circular markers, we annotated those markers within the CME dataset and conducted a histogram-based statistical analysis to examine their scale characteristics. To explore how target

sizes are distributed, we applied k-means clustering on the experimental dataset. As depicted in Figure 4(a), each point in the scatter plot corresponds to a circular marker, with the x-axis and y-axis signifying the marker's width and height, respectively. Figure 4(b) then illustrates the count distribution generated by the k-means clustering results. The k-means clustering produced six categories: 11×15, 16×20, 20×27, 24×33, 25×42, and 33×50. According to Table 1, targets with sizes of 11×15 (24.7%) and 16×20 (23.5%) are mapped to the P3 feature layer with sizes of 1.38×1.88 and 2.50×3.38 pixels, respectively, which are below the effective detection threshold (typically requiring ≥3×3 pixels). This may result in excessively weak feature responses that gradually vanish during subsequent convolutional operations, leading to missed detection of small targets. Additionally, based on the concept of effective receptive fields, although deep networks have larger theoretical receptive fields, the strong response range in the center of their actual effective receptive fields often exceeds the physical size of small targets, causing target features to be submerged by background noise. Therefore, to address the issue of small-scale targets, which account for 48.2% of the dataset, we propose adjusting the feature map structure: constructing a small target detection head in the P2 (stride=4) layer to increase the mapping size of 11×15 targets to 3.75×5 pixels. Simultaneously, we refine the feature fusion strategy by integrating high-resolution shallow-layer features with semantic representations from deeper layers, thus enhancing the discriminative power of the features while preserving detail sensitivity. Furthermore, since excessively large targets are not present in circular landmark detection, we have removed the P5 layer detection head.

3.1.2 Feature Extraction Module

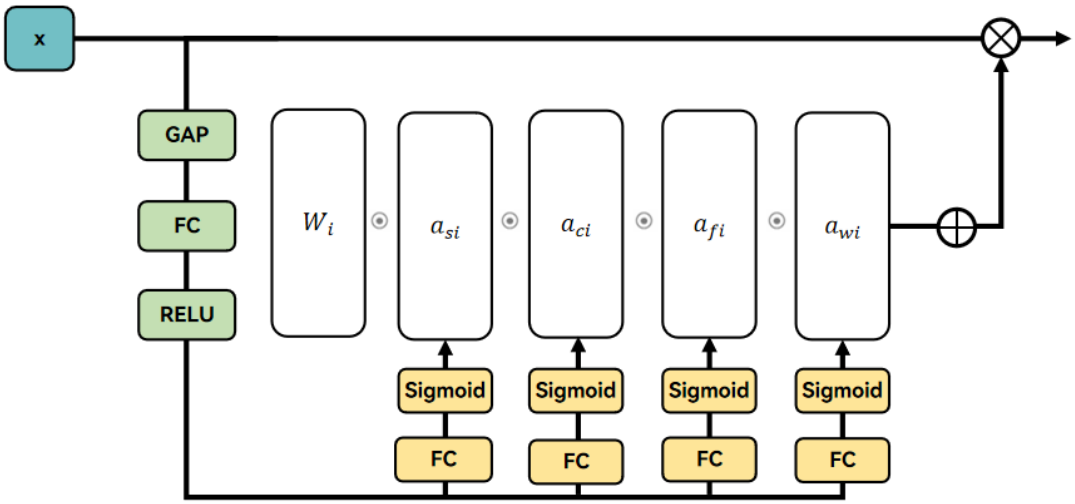

Figure. 5 Schematic Diagram of ODConv Structure.

Conventional convolution is a standard operation widely used in object detection. However, it has limitations, particularly when dealing with complex image features. In conventional convolution, the weights of each kernel are fixed, meaning the kernel's response to the input feature map remains consistent throughout the convolution process. This fixed approach limits the flexibility of the convolution operation, as it cannot automatically adjust kernel weights based on different input data, thus failing to capture diverse features. Conventional convolution applies the same kernel weights across the entire input, lacking contextual awareness of different spatial locations, channels, or output features. Simple convolution operations often fail to fully understand the relationships between different regions or dimensions when handling complex features. To enhance the performance of convolution, it is common to use multiple kernels or larger kernels, which increases the number of model parameters, leading to higher computational costs and memory requirements. This can become a computational bottleneck when processing large images or complex tasks. To address these issues, we introduce Omni-Dimensional Dynamic Convolution (ODConv), whose structure is shown in Figure 5. ODConv employs a multi-dimensional attention mechanism and a parallel strategy to train and learn along the four dimensions of the convolution kernel space, achieving complementarity and thus endowing the convolution kernel with dynamic properties. Therefore, we

designed a COD feature extraction module based on ODConv to extract more critical and discriminative features. The convolution operation in ODConv is expressed as Equation (2).

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

The attention values a_{si}, a_{ci}, a_{fi}, and a_{wi} are computed through an improved SE-type attention module, which employs multiple attention heads. First, the input is compressed into a feature vector using Global Average Pooling (GAP). This vector is then passed through a Fully Connected (FC) layer and routed into four branches, each corresponding to a specific attention type: spatial, channel, filter, and kernel. Each branch outputs the corresponding attention value through a Sigmoid function. These four attention values are applied in a specific sequential order, enabling the convolution operation to adjust based on different aspects of the input data. By applying these attention values sequentially, the model can learn complex contextual information across different dimensions. In this way, ODConv captures rich contextual information across multiple dimensions of the convolution kernel while introducing minimal additional parameters, effectively improving the speed of both training and inference.

3.2 AFGCAttention

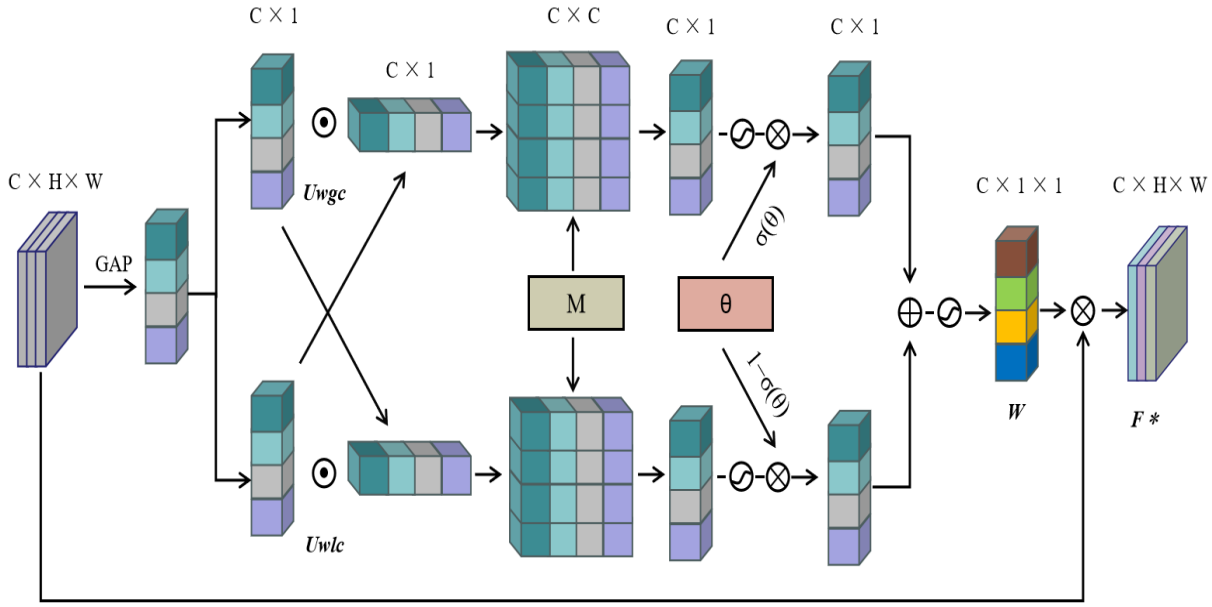

Figure. 6 Structure Diagram of AFGCAttention.

To overcome the challenges posed by complex backgrounds and insufficient fine detail, while strengthening the model's capacity to detect small targets, we introduce the AFGCAttention (Adaptive Fine-Grained Channel Attention) module. This component aims to enhance the accuracy of circular marker detection in complicated scenarios. Figure 6 illustrates the module's architecture. By integrating both global and local information and adaptively modulating channel weights, the module fuses its final attention map with the input features on a channel-by-channel basis, thereby improving feature distinguishability and detection precision. The AFGCAttention module is composed of three primary parts: the Global Information Extraction module, the Local Information Interaction module, and the Adaptive Fusion module.

First, to enable the network to focus on global information in the image while reducing computational costs and the risk of overfitting, we perform global aggregation of the image features. Through Global Average Pooling (GAP), the spatial information of each channel is compressed into a scalar, resulting in a channel-level global descriptor. Specifically, given a feature map $F \in R^{C\times H\times W}$, where C is the number of channels, and H and W are the height and width of the feature map, respectively, the spatial features of each channel are aggregated into a scalar U_n through GAP, representing the global information of that channel. The calculation formula is as shown in Equation (3):

$$U_n = \mathrm{GAP}(F_n) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_n(i,j) \tag{3}$$

Through this operation, the feature map F is compressed into a channel descriptor U of size $C \times 1 \times 1$, thereby removing spatial information while retaining the global information of each channel. This approach helps the network better capture the global features of the image and reduces excessive spatial redundancy.

Next, to enhance the model's ability to perceive local details, particularly when dealing with small targets or regions rich in details, we introduce a banded matrix B to model the interactions between local channels. By performing a multiplication operation between the channel descriptor U and the banded matrix B, we can extract the local information U_{lc}. Specifically, the calculation formula for the local information is as shown in Equation (4):

$$U_{lc} = \sum_{i=1}^{k} U \cdot b_i \tag{4}$$

Here, k represents the size of the local neighborhood, and b_i is an element in the banded matrix B. The local information is realized through one-dimensional convolution, which captures the inter-channel relationships within local regions, thereby enhancing the expressive power of local features. This step enables the network to focus more on local details, particularly in the detection of small objects, significantly improving feature representation capabilities.

To further effectively fuse global and local information, the paper proposes using cross-correlation operations to capture the relationship between the two. The global information U_{gc} and the local information U_{lc}^T are combined through Equation (5):

$$M = U_{gc} \cdot U_{lc}^T \tag{5}$$

Here, M is the correlation matrix between global and local information, which captures the interaction of information at different granularities by computing the relationship between the two. The fusion of global and local information aims to achieve effective integration of information, enabling the network to focus on both global and local features at varying granularities, thereby enhancing the overall expressive power of features. This integration of information is significant for improving model performance, especially in complex scenarios.

To further enhance the effectiveness of the fusion of global and local information, the paper proposes an adaptive weighted fusion strategy. This strategy extracts information from the correlation matrix M and its transpose matrix to generate weight vectors for global and local information. Then, using these weights, it dynamically fuses the global and local information to obtain the weighted result. The specific calculation formula is shown in Equation (6):

$$U_{wgc} = \sum_{j=1}^{c} M_{i,j}, U_{wlc} = \sum_{i,j=1}^{c} M_{i,j} \tag{6}$$

This process achieves adaptive weighted fusion through the Sigmoid activation function σ and learnable parameters θ, as shown in Equation (7):

$$W = \sigma\left(\sigma(\theta) \times \sigma\left(U_{wgc}\right) + \left(1 - \sigma(\theta)\right) \times \sigma(U_{wlc})\right) \tag{7}$$

Adaptive weighted fusion dynamically prioritizes the most salient features by adjusting the relative importance of global and local information. This strategy effectively filters out superfluous data, allowing the network to concentrate on critical characteristics and thereby enhancing both flexibility and accuracy in target detection. By employing this adaptive fusion method, the model retains broad contextual understanding while sharpening its attention to local details, ultimately improving detection performance.

Finally, by applying an element-wise multiplication between the weighted fused channel descriptor W and the input feature map F, we obtain the final output feature map F^*, as shown in Equation (8).

$$F^* = W \odot F \tag{8}$$

Here, ⊙ indicates element-wise multiplication, W represents the weighted channel descriptor, and F is the input feature map. The final output feature map, F^*, results from merging the input feature map with the weighted channel descriptor. This fusion effectively combines both global and local information, heightening the network's responsiveness to targets and thereby boosting detection accuracy. Such an approach proves particularly advantageous in complex environments, where the ability to identify fine details and small targets is essential.

3.3 NWDloss

In object detection tasks, the similarity between bounding boxes is typically measured using loss functions based on IoU (Intersection over Union), which is an effective similarity metric in most object detection scenarios. However, in circular marker detection tasks, there are often a large number of small targets, and IoU-based loss is extremely sensitive to changes in the position of bounding boxes. Even if two bounding boxes are highly similar, a slight shift in their relative positions can cause a dramatic fluctuation in the IoU value. This makes IoU unable to provide a sufficiently stable measurement result when the object's position undergoes minor shifts. For example, in the detection of small objects, a slight movement of the bounding box can lead to significant fluctuations in the IoU value, thereby affecting detection accuracy.

Additionally, IoU-based loss is highly sensitive to differences in the scale of bounding boxes. Between objects with large scale differences, the IoU value can vary significantly, especially when comparing larger objects with smaller ones. Even if they overlap considerably, the IoU value may still be relatively low. The impact of this scale difference is particularly pronounced for small objects. To address this challenge, we introduce NWD (Normalized Wasserstein Distance) loss as a measurement tool for bounding boxes. NWD draws on optimal transport theory by leveraging Wasserstein distance to evaluate how closely two bounding boxes match. It models these boxes as two-dimensional Gaussian distributions, thereby focusing on the shape and location of the target rather than solely its size, enabling a more precise measure of similarity. Moreover, NWD exhibits smoother changes than IoU when the object shifts slightly, capturing positional deviations more accurately. As it employs a Gaussian-based approach, NWD can more reliably isolate the object region within the bounding box, remaining less susceptible to interference from the surrounding background—an especially valuable trait for circular marker detection.

The specific steps for applying NWD are as follows:

Initially, the bounding box must be converted into a Gaussian distribution. Consider a horizontal bounding box $R = (cx, cy, w, h)$, where cx and cy denote the box's center coordinates, and w and h specify its width and height. The corresponding equations can be found in (9) and (10).

$$\mu = (cx, cy) \tag{9}$$

$$\Sigma = \begin{pmatrix} \frac{w^2}{4} & 0 \\ 0 & \frac{h^2}{4} \end{pmatrix} \tag{10}$$

In this representation, μ and Σ denote the bounding box's center coordinates and covariance matrix, respectively, collectively characterizing the target's shape.

To compute NWD, the Wasserstein distance is used to measure the difference between two Gaussian distributions. The calculation method is as shown in Equation (11):

$$W_2^2(\mu_1, \mu_2, \Sigma_1, \Sigma_2) = ||\mu_1 - \mu_2||^2 + \mathrm{Tr}(\Sigma_1 + \Sigma_2 - 2\left(\Sigma_1^{\frac{1}{2}}\Sigma_2\Sigma_1^{\frac{1}{2}}\right)^{\frac{1}{2}}) \tag{11}$$

Finally, to transform the Wasserstein distance into a more discriminative similarity metric, NWD normalizes it, yielding the final NWD as shown in Equation (12):

$$NWD(\mu_1, \mu_2, \Sigma_1, \Sigma_2) = \exp\left(-\frac{W_2^2(\mu_1, \mu_2, \Sigma_1, \Sigma_2)}{C}\right) \tag{12}$$

Here, C serves as a normalization parameter, set to 12.3 in this paper.

4. Experiments

4.1 Experimental Setup

The model was developed using the PyTorch framework. As outlined in Table 2, both training and inference were conducted on a single NVIDIA GTX 1080 GPU (12 GB), together with an AMD Ryzen 7 2700 CPU and 48 GB of RAM. The optimization process employed Stochastic Gradient Descent (SGD) for 100 epochs, using a batch size of 16 and an initial learning rate of 0.01. No pre-trained weights were used, and the input dimensions were held constant at 640 × 640 pixels for both training and detection.

Table 2 Experimental Configuration

Configuration	Parameter
CPU	AMD Ryzen 7 2700
GPU	Nvidia GeForce GTX 1080Ti
Operating system	Windows 10
Accelerated environment	Pycharm2020
Libraries	Torch1.13.0

4.2 Evaluation Metrics

To assess MarkYolo's detection performance, we employed multiple metrics: AP, Recall, AP50, AP50-95, and GFLOPs. In this context, Precision measures the proportion of true positives among all instances identified as positive, reflecting how accurately the model predicts positive samples. Conversely, Recall gauges the fraction of actual positives that are correctly recognized, indicating the model's overall capacity to identify positive instances. The formulas for these two metrics are provided in Equations (13) and (14).

$$precision = \frac{TP}{TP + FP} \tag{13}$$

$$recall = \frac{TP}{TP + FN} \tag{14}$$

In these equations, a True Positive (TP) refers to correctly identifying a circular marker, while a False Positive (FP) occurs when background or noise is mistakenly classified as a circular marker. A False Negative (FN) arises when an actual circular marker is misidentified as another feature or as background.

By evaluating precision and recall values at multiple thresholds, we can plot the Precision-Recall (P-R) curve, with precision on the vertical axis and recall on the horizontal axis. The area under this curve is defined as Average Precision (AP), as given in Equation (15):

$$AP = \int_0^1 P(R)dR \tag{15}$$

AP50 indicates the average precision when the Intersection over Union (IoU) threshold is set to 50%. IoU gauges how much the predicted bounding box overlaps the ground-truth bounding box, and a detection is deemed correct if their IoU is at least 50%.

AP50-95 extends this measure by calculating average precision across IoU thresholds ranging from 50% to 95%. By evaluating performance under increasingly strict overlap criteria, it provides a more nuanced view of the model's detection capabilities, where a higher AP50-95 score signifies stronger overall performance.

GFLOPs (Giga Floating-point Operations per Second) quantifies the computational load during the model's inference stage, showing how many floating-point operations are carried out each second. Within YOLO-based detection networks, the GFLOP value reflects the system's computational complexity and resource consumption, as a higher GFLOP count typically suggests a more intricate architecture and a greater demand on computing resources. Therefore, GFLOPs serve as a key benchmark for measuring efficiency and performance.

4.3 Comparison Experiment

We conducted a comparative assessment of the baseline YOLOv10n model (employing the C2f module) versus an enhanced YOLOv10n model that integrates the COD module. Both models were evaluated in terms of precision, recall, and the AP50 and AP50-95 metrics. Table 3 presents the outcomes. According to the data analysis, swapping the C2f module with the COD module delivers performance on par with the original, without compromising effectiveness. Specifically, the original YOLOv10n achieved a precision of 95.9%, a recall of 90.2%, an AP50 of 95.7%, and an AP50-95 of 70.5%. With the COD module in place, the recall rose by 0.3%,

and AP50 and AP50-95 each increased by 0.5% and 0.1%, respectively, while GFLOPs decreased by 32%. These findings confirm that the COD module substantially enhances the YOLOv10n model's overall performance while preserving accuracy.

Table 3 Experimental Comparison of Feature Extraction Modules

Model	Precision	Recall	AP50	AP50-95	GFLOPs
YOLOv10n	95.9	90.2	95.7	70.5	6.5
YOLOv10n+COD	95.7	90.5	96.2	70.6	4.4

To validate the effectiveness and parameter requirements of the small target detection layer, we conducted a comparative experiment between the model with the modified detection layer and the original YOLOv10n model. The results are shown in Table 4. The model with the added P2 detection layer and the removed P5 detection layer achieved a 4.3% improvement in recall rate, a 2.3% increase in AP50 accuracy, and a 1.9% enhancement in AP50-95 accuracy. These results indicate that modifying the detection layer significantly improves detection performance, although it also increases the model's computational load.

Table 4 Experimental Comparison of Different Detection Layer Combinations

Model	Precision	Recall	AP50	AP50-95	GFLOPs
YOLOv10n	95.9	90.2	95.7	70.5	6.5
YOLOv10n+P2P3P4	96.2	94.5	98.0	72.4	10.0

To objectively validate the performance of the proposed MarkYolo network, we compared it with the baseline model YOLOv10n and other single-stage detection algorithms such as SSD, YOLOv5n, YOLOv8n, RetinaNet, as well as the two-stage detection algorithm Fast R-CNN, under identical environmental configurations. As shown in Table 5, Fast R-CNN demonstrates high accuracy and low localization error but suffers from slow speed due to the need to classify and regress each candidate box, resulting in high computational complexity. SSD offers fast detection speed, enabling object classification and location regression in a single forward pass, making it suitable for real-time applications and mobile devices. However, it performs poorly in small object detection and images with objects of varying scales. RetinaNet addresses the class imbalance issue through the introduction of Focal Loss, achieving good detection results for both small and large objects but with relatively high computational complexity compared to other methods. YOLOv5 is known for its speed and suitability for real-time object detection, striking a balance between accuracy and performance. However, its computational complexity is relatively low, and it struggles with small object detection. Other YOLO models, including YOLOv8 and YOLOv10, lag significantly behind MarkYolo in terms of model precision, parameter count, computational load, and model size, with the latter demonstrating clear advantages. According to the data in Table 5, MarkYolo excels in precision, recall, and AP values, achieving high levels of 98.6% and 75.4% in AP50 and AP50-95 metrics, respectively. Additionally, MarkYolo's computational complexity (GFLOPs) is slightly higher at 8.6 compared to the original model. When compared to other algorithms, MarkYolo shows significant improvements in precision and overall performance, particularly in AP50 and AP50-95 metrics, where it clearly outperforms other algorithms. Figure 7 showcases some of the detection results.

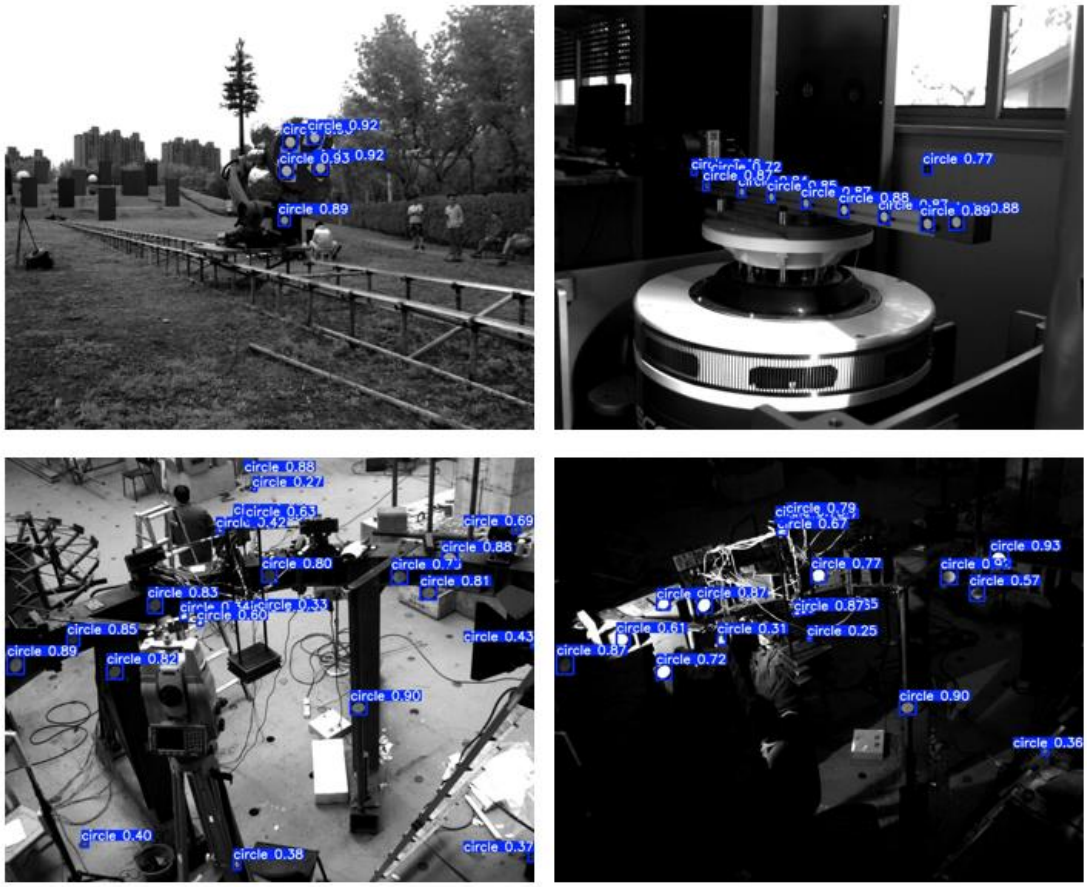

Figure 7. Detection Results of MarkYolo.

Table 5 Comparative Experiment of Different Models

Model	Precision	Recall	AP50	AP50-95	GFLOPs
Fast R-CNN	92.23	87.9	62.7	62.73	206.67
SSD	94.56	46.23	92.46	54.57	87.86
RetinaNet	93.44	84.9	89.57	59.75	194.82
YOLOv5n	94.31	92.8	95.84	67.12	15.85
YOLOv8n	95.98	90.43	95.85	70.46	8.1
YOLOv10n	96.3	90.2	96.7	70.5	6.5
MarkYolo	98.3	96.4	98.6	75.4	8.6

4.3 Ablation Experiment

To validate the performance of the MarkYolo model, ablation experiments were conducted on the YOLOv10n network Architecture Improvements, the AFGCA attention module, and the NWD loss function, as shown in Table 6. The symbol "✓" indicates the use of the corresponding method or module.

Table 6 Model Ablation Experiment

Architecture Improvements	AFGCA	NWD	Precision	Recall	AP50	AP50-95	GFLOPs
✓			95.9	90.2	95.7	70.5	6.5
	✓		97.3	94.5	97.8	72.4	8.6
		✓	96.8	90.8	96.7	71.2	6.5
✓	✓		97.1	92.4	97.1	71.8	6.5
✓		✓	97.9	94.7	98.0	72.6	8.6
	✓	✓	97.5	94.6	97.8	72.7	8.6
✓	✓	✓	96.8	95.3	97.6	73.4	6.5

From Table 6, it can be observed that the MarkYolo algorithm significantly enhances object detection performance while only increasing computational complexity by 2.1 GFLOPs. The recall rate reaches 96.4%, accuracy achieves 98.3%, AP50 is 98.6%, and the AP50-95 metric is 75.4%. Compared to the original algorithm, the accuracy, recall, AP50, and AP50-95 of MarkYolo improved by 2.2%, 6.2%, 1.9%, and 4.9%, respectively. Compared to the standalone network architecture improvement (P2P3P4+COD), the algorithm's accuracy increased by 1.0%, recall by 1.9%, and AP50-95 by 3.0%. Compared to the standalone AFGCA attention module, accuracy, recall, AP50, and AP50-95 improved by 1.5%, 5.6%, 1.0%, and 4.2%, respectively. Additionally, the standalone addition of NWDLoss further enhanced the network's detection performance, with accuracy, recall, AP50, and AP50-95 increasing by 1.1%, 2.2%, 1.0%, and 1.3%, respectively, compared to the original network. These results demonstrate that the incremental integration and improvement of multiple modules significantly enhance the performance of the YOLOv10n algorithm. The synergistic effects among these modules play a crucial role in improving detection accuracy and recall, highlighting the importance of their combined contributions to optimizing algorithm performance.

5. Conclusion

In this work, we introduce MarkYolo, a deep-learning-based method specifically aimed at detecting circular markers in high-speed video measurement contexts. Built on the cutting-edge YOLOv10 framework, we developed a compact detector optimized for high-speed environments. By incorporating full-dimensional dynamic convolution (ODConv) as a feature extraction component, we lowered the model’s computational overhead while maintaining robust feature representation. We also refined the feature layer design to better accommodate small object detection. The AFGCAttention mechanism was integrated to simultaneously improve detection accuracy and processing speed. Furthermore, by employing NWD loss, the model gains greater adaptability to various detection scales. MarkYolo demonstrates excellent results for small circular marker detection, striking an effective balance between performance and accuracy. The model showcases strong resilience in complex high-speed measurement scenarios, as evidenced by experiments on the CME dataset. It achieves a 75.4% AP50-95—a 4.9% increase compared to the original YOLOv10—clearly addressing the challenges of circular marker detection under intricate conditions and providing technological support for high-speed photography applications.

Despite the promising performance of our enhanced YOLOv10n on the CME dataset, certain constraints remain. Most notably, the dataset’s limited range of scene types and imbalanced samples suggest a need for more comprehensive, evenly distributed data to bolster the model’s generalization. Moreover, further refinement is required to streamline the model’s lightweight design for easier deployment on mobile or edge devices, thereby expanding its practical usability.

Data sharing agreement

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Declaration of Conflicting Interests

The authors declared no potential conflicts of interest with respect to the research, author-ship, and publication of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants Nos. 42221002 and 42471502.

References

[1] T. Luhmann (2010) Close range photogrammetry for industrial applications, ISPRS J. Photogramm. Remote Sens., 65:558–569.

[2] J. Baqersad, P. Poozesh, C. Niezrecki and P. Avitabile (2017) Photogrammetry and optical methods in structural dynamics – A review, Mech. Syst. Signal Process., 86:17–34.

[3] I.D. Wallace, N.J. Lawson, A.R. Harvey, J.D. Jones and A.J. Moore (2005) High-speed close-range photogrammetry for dynamic shape measurement, In: Proc. 26th Int. Congr. High-Speed Photogr. Photonics, 358–366.

[4] B.F. Spencer Jr, V. Hoskere and Y. Narazaki (2019) Advances in computer vision-based civil infrastructure inspection and monitoring, Engineering, 5:199–222.
[5] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci and P. Fieguth (2015) A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure, Adv. Eng. Inform., 29:196–210.
[6] Q. Yuan, Y. Shi and M. Li (2024) A review of computer vision-based crack detection methods in civil infrastructure: Progress and challenges, Remote Sens., 16:2910.
[7] X. Tong, S. Gao, S. Liu, Z. Ye, P. Chen, S. Yan, X. Zhao, L. Du, X. Liu and K. Luan (2017) Monitoring a progressive collapse test of a spherical lattice shell using high-speed videogrammetry, Photogramm. Rec., 32:230–254.
[8] X. Liu, X. Tong, W. Lu, S. Liu, B. Huang, P. Tang and T. Guo (2020) High-speed videogrammetric measurement of the deformation of shaking table multi-layer structures, Measurement, 154:107486.
[9] X. Tong, H. Shi, Z. Ye, P. Chen, Z. Liu, Y. Gao, Y. Li, Y. Xu and H. Xie (2024) Liquid-level response measurement using high-speed videogrammetry with robust multiple sphere tracking, Measurement, 228:114290.
[10] X. Liu, X. Tong, X. Yin, X. Gu and Z. Ye (2015) Videogrammetric technique for three-dimensional structural progressive collapse measurement, Measurement, 63:87–99.
[11] J. Herráez, J.C. Martínez, E. Coll, M.T. Martín and J. Rodríguez (2016) 3D modeling by means of videogrammetry and laser scanners for reverse engineering, Measurement, 87:216–227.
[12] Y.H. Jo and S. Hong (2019) Three-dimensional digital documentation of cultural heritage site based on the convergence of terrestrial laser scanning and unmanned aerial vehicle photogrammetry, ISPRS Int. J. Geo-Inf., 8:53.
[13] M. Shahbazi, G. Sohn, J. Théau and P. Menard (2015) Development and evaluation of a UAV-photogrammetry system for precise 3D environmental modeling, Sensors, 15:27493–27524.
[14] J. Butler, S. Lane, J. Chandler and E. Porfiri (2002) Through-water close range digital photogrammetry in flume and field environments, Photogramm. Rec., 17:419–439.
[15] A. Capolupo, S. Pindozzi, C. Okello, N. Fiorentino and L. Boccia (2015) Photogrammetry for environmental monitoring: The use of drones and hydrological models for detection of soil contaminated by copper, Sci. Total Environ., 514:298–306.
[16] F.F. Ahmadi (2017) Integration of industrial videogrammetry and artificial neural networks for monitoring and modeling the deformation or displacement of structures, Neural Comput. Appl., 28:3709–3716.
[17] S. Anweiler and R. Ulbrich (2020) Application of videogrammetry in the mechanics of multi-phase systems, Therm. Sci., 24:3577–3588.
[18] S. Li and T. Xue (2025) Modeling and measurement of 3D velocity for rising bubbles utilizing single-view laser scanning, IEEE Trans. Instrum. Meas., DOI: 10.1109/TIM.2025.XXXXXXX.
[19] H. Hu, J. Liang, Z.-Z. Xiao, Z.-Z. Tang, A.K. Asundi and Y.-X. Wang (2012) A four-camera videogrammetric system for 3-D motion measurement of deformable object, Opt. Lasers Eng., 50:800–811.
[20] J. Leifer, J.T. Black, S.W. Smith, N. Ma and J.K. Lumpp (2007) Measurement of in-plane motion of thin-film structures using videogrammetry, J. Spacecr. Rockets, 44:1317–1325.
[21] J. Han, N. Lu and M. Dong (2008) Design of circular coded target and its application to optical 3D-measurement, In: Proc. 4th Int. Symp. Precis. Mech. Meas., 801–806.
[22] S. Zheng, S. Liu, Z. Ye, X. Ma, B. Wang and J. Zhang (2025) Global automatic detection method employing multi-level constraints for circular markers in high-speed videogrammetry, J. Appl. Remote Sens., 19:016501.
[23] L.M. Galantucci, F. Lavecchia, G. Percoco and S. Raspatelli (2014) New method to calibrate and validate a high-resolution 3D scanner, based on photogrammetry, Precis. Eng., 38:279–291.
[24] H.S. Park, J.Y. Kim, J.G. Kim, S.W. Choi and Y. Kim (2013) A new position measurement system using a motion-capture camera for wind tunnel tests, Sensors, 13:12329–12344.
[25] C.-T. Ho and L.-H. Chen (1995) A fast ellipse/circle detector using geometric symmetry, Pattern Recognit., 28:117–124.
[26] A.Y.-S. Chia, D. Rajan, M.K. Leung and S. Rahardja (2011) Object recognition by discriminative combinations of line segments, ellipses, and appearance features, IEEE Trans. Pattern Anal. Mach. Intell., 34:1758–1772.

[27] D.K. Prasad, M.K. Leung and S.-Y. Cho (2012) Edge curvature and convexity based ellipse detection method, Pattern Recognit., 45:3204–3221.
[28] P. Viola and M. Jones (2001) Rapid object detection using a boosted cascade of simple features, In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 1:I –I.
[29] D.G. Lowe (2004) Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis., 60:91–110.
[30] P.F. Felzenszwalb, R.B. Girshick, D. McAllester and D. Ramanan (2009) Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach. Intell., 32:1627–1645.
[31] M. Szarvas, A. Yoshizawa, M. Yamamoto and J. Ogata (2005) Pedestrian detection with convolutional neural networks, In: Proc. IEEE Intell. Veh. Symp., 224–229.
[32] R. Girshick, J. Donahue, T. Darrell and J. Malik (2014) Rich feature hierarchies for accurate object detection and semantic segmentation, In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 580–587.
[33] R. Girshick (2015) Fast R-CNN, In: Proc. IEEE Int. Conf. Comput. Vis., 1440–1448.
[34] S. Ren, K. He, R. Girshick and J. Sun (2016) Faster R-CNN: Towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell., 39:1137–1149.
[35] K. He, G. Gkioxari, P. Dollár and R. Girshick (2017) Mask R-CNN, In: Proc. IEEE Int. Conf. Comput. Vis., 2961–2969.
[36] J. Redmon (2016) You only look once: Unified, real-time object detection, In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 779–788.
[37] J. Redmon and A. Farhadi (2017) YOLO9000: Better, faster, stronger, In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 7263–7271.
[38] M.-T. Pham, L. Courtrai, C. Friguet, S. Lefèvre and A. Baussard (2020) YOLO-Fine: One-stage detector of small objects under various backgrounds in remote sensing images, Remote Sens., 12:2501.
[39] D. Hong, L. Gao, N. Yokoya, J. Yao, J. Chanussot, Q. Du and B. Zhang (2020) More diverse means better: Multimodal deep learning meets remote-sensing imagery classification, IEEE Trans. Geosci. Remote Sens., 59:4340–4354.
[40] Z. Hong, T. Yang, X. Tong, Y. Zhang, S. Jiang, R. Zhou, Y. Han, J. Wang, S. Yang and S. Liu (2021) Multi-scale ship detection from SAR and optical imagery via a more accurate YOLOv3, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., 14:6083–6101.
[41] G. Cheng, Y. Si, H. Hong, X. Yao and L. Guo (2020) Cross-scale feature fusion for object detection in optical remote sensing images, IEEE Geosci. Remote Sens. Lett., 18:431–435.
[42] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu and A.C. Berg (2016) SSD: Single shot multibox detector, In: Proc. Eur. Conf. Comput. Vis., 21–37.
[43] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie (2017) Focal loss for dense object detection, In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2980–2988.
[44] P. Su, H. Han, M. Liu, T. Yang and S. Liu (2024) MOD-YOLO: Rethinking the YOLO architecture at the level of feature information and applying it to crack detection, Expert Syst. Appl., 237:121346.
[45] D. Yang, M.I. Solihin, I. Ardiyanto, Y. Zhao, W. Li, B. Cai and C. Chen (2024) A streamlined approach for intelligent ship object detection using EL-YOLO algorithm, Sci. Rep., 14:15254.
[46] C. Li, A. Zhou and A. Yao (2022) Omni-dimensional dynamic convolution, arXiv:2209.07947.
[47] H. Sun, Y. Wen, H. Feng, Y. Zheng, Q. Mei, D. Ren and M. Yu (2024) Unsupervised bidirectional contrastive reconstruction and adaptive fine-grained channel attention networks for image dehazing, Neural Netw., 176:106314.
[48] J. Wang, C. Xu, W. Yang and L. Yu (2021) A normalized Gaussian Wasserstein distance for tiny object detection, arXiv:2110.13389.
[49] W. Luo, Y. Li, R. Urtasun and R. Zemel (2016) Understanding the effective receptive field in deep convolutional neural networks, Adv. Neural Inf. Process. Syst., 29.