ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 3 - 2026

Design and Optimization of an Automatic Detection System for Chip Defect Recognition Based on YOLOv8

Renjie Yin[1] *and Dong Zhang*[1,*]

[1] School of Mechanical Engineering, shanghai dianji university, Shanghai, 200000, Shanghai, China

SUMMARY: *According to the demands of enterprises, in the current field of chip screening, there are significant loopholes in the traditional manual screening method. It not only takes a lot of time and effort but also leads to a high rate of false detection, which increases the labor cost for enterprises. The traditional chip defect detection work usually adopts the manual visual inspection method. When defective chips are found, these unqualified products are manually removed. However, manual visual inspection has many variable factors and suffers from a series of problems such as low recognition accuracy, poor real-time performance, and high false detection rate. With the gradual increase in chip production speed and the ever-increasing quality requirements for chips, the traditional manual visual inspection method can no longer meet the industrial requirements for real-time chip detection. This system has optimized the detection performance of YOLOv8, enabling it to accurately identify chip defects in complex backgrounds and possess real-time detection capabilities. It is applicable to multiple fields such as industrial inspection. Experimental results show that the system achieves high precision and mAP [2]on the test set, effectively replacing traditional manual inspection methods and significantly enhancing detection efficiency and accuracy.*

KEYWORDS: *Deep Learning; YOLOV8; Chip defect detection; Small target detection; Chip screening*

1 Introduction

As the "digital cornerstone" of modern electronic devices, the quality of chips directly determines the system's computing speed, power efficiency, and long-term reliability. Driven by emerging technologies such as 5G communication, artificial intelligence, and autonomous driving, chip manufacturing technology is approaching its physical limits at a rate of one process node every two years. The 3-nanometer process has entered the mass production stage, and 2-nanometer and below technologies have become a competitive focus for global semiconductor giants. This exponential growth in integration density has led to the number of transistors accommodated on a single chip surface exceeding billions, resulting in a sharp amplification of the impact of nanoscale defects - a particle pollution with a diameter of only 0.1 microns can cause tens of thousands of transistors to fail, resulting in the paralysis of the entire chip function [1, 2].

In the more than 1000 processes of integrated circuit manufacturing, key processes such as photolithography, etching, and chemical mechanical polishing (CMP) may introduce defects: fluctuations in the light source of extreme ultraviolet (EUV) lithography machines

*zhangd@sdju.edu.cn
https://doi.org/10.65102/is20261101 Published: 10 June, 2026

can cause line width deviation, uneven energy of plasma etching can lead to metal wire breakage, and wear of CMP polishing pads may form surface scratches. These defects not only directly reduce the yield rate (70% of the yield loss in advanced processes is due to insufficient testing), but also cause reliability issues in end products - a case study of a certain automotive chip manufacturer shows that leakage of field-effect transistors (MOSFETs) caused by etching defects has led to the failure of a batch of in car controllers in high-temperature environments, resulting in billions of dollars in recall losses [3]. According to the International Semiconductor Industry Association (SEMI), the direct economic losses caused by manufacturing defects in the global semiconductor industry will reach 5.27 billion US dollars in 2023, with advanced processes (below 7 nanometers) accounting for over 65% of defect costs, highlighting the strategic value of defect detection technology at the nanoscale.

The current chip defect detection field is facing three core challenges, which are essentially the conflict between physical limits at the nanoscale and industrial needs [4, 5]: 1) The efficiency bottleneck of manual detection: traditional microscope visual inspection relies on the visual judgment of inspectors, with a speed of only 1-2 wafers per minute, and is affected by factors such as fatigue and experience differences, resulting in a missed detection rate of up to 15% -20%. The production data of a 12-inch wafer fab shows that when using manual inspection, the yield of the 28-nanometer process is only 88%, of which 60% of the yield loss is due to undetected small defects. As the process nodes advance, manual inspection can no longer meet the production needs - the value of wafers in the 3-nanometer process exceeds $20000, and manual missed inspections may result in a loss of over $4000 per wafer. 2) The accuracy limitation of algorithm recognition: Traditional image processing algorithms such as Canny edge detection and SIFT feature matching are based on manually designed feature extractors, which are difficult to adapt to the complex surface structures of advanced processes. In processes below 7 nanometers, the line width/pitch of the metal interconnect layer is reduced to below 40 nanometers, forming a dense periodic grid structure. Normal process fluctuations highly overlap with the signal characteristics of real defects, resulting in a sharp drop in the defect recognition rate of traditional algorithms to below 60%. In addition, the vertical channel structure introduced by 3D stacking technology (such as 3D NAND) expands the defect morphology from two-dimensional plane to three-dimensional space, further increasing the difficulty of algorithm recognition. 3) The real-time contradiction of system integration: Industry 4.0 requires detection systems to have "zero buffering" processing capability, which means that the detection delay must be less than the wafer transfer cycle (usually 2 seconds/wafer). The existing detection systems generally adopt an offline mode of "collection storage processing", which cannot achieve real-time closed-loop control with the production line. For example, the AOI (Automatic Optical Inspection) system of a storage chip manufacturer suffered from delayed processing, resulting in delayed feedback of defect information and repeated defects in five consecutive batches of products, resulting in a direct loss of over 8 million US dollars.

To solve the above problems, this study proposes an intelligent detection framework based on deep learning, which achieves dual breakthroughs in accuracy and speed through algorithm innovation and system optimization: 1) Optimization of small object detection algorithm: for the challenge of defect size less than 0.1% of the image area, the YOLOv8 algorithm is triple improved: ConvNeXt module is introduced in Backbone to reduce computational complexity through depthwise separable convolution; In the Neck section, BiFPN (Bidirectional Feature Pyramid Network) is used to enhance the multi-scale feature fusion capability; Design a dynamic anchor box generation mechanism at the Head layer to adaptively adjust the proportion of detection boxes based on defect size distribution.

Experiments have shown that the improved algorithm is effective in detecting defects at the 0.3-micron level, mAP@0.5 The indicator has reached 96.3%, an increase of 28 percentage points compared to traditional algorithms. 2) Complex background suppression technology: To solve the problem of metal texture interference, a dual stream detection network architecture is proposed: one stream uses U-Net++to learn the statistical features of periodic textures and generate texture suppression maps; The other branch extracts local features of defects through ResNeSt, and after fusion, inputs them into the Transformer decoder for defect classification. In the test set of the 28-nanometer process, this approach reduced the false alarm rate from 25% to 3.8% while maintaining a recall rate of 92%. 3) Industrial grade real-time system implementation: Build a layered detection architecture to meet the detection requirement of 300 pieces per minute: The hardware layer adopts NVIDIA Jetson AGX Orin platform, combined with FPGA to achieve image preprocessing acceleration; The algorithm layer implements model quantization compression (FP32 → INT8) to increase inference speed to 120FPS; The system layer is designed with a circular buffer and pipeline parallel mechanism to control end-to-end latency within 180ms. In the mass production verification of a certain wafer fab, the system improved the detection efficiency by 150 times and saved over 20 million yuan in quality inspection costs per line per year. More noteworthy is that the system achieves self-evolution through continuous learning mechanism (CLS): it automatically collects new defect samples on the production line every month and uses federated learning framework for cross factory model updates, shortening the adaptation cycle of the model to emerging defects from the traditional method of 3 months to 7 days. This closed-loop system of "detection learning optimization" provides technical possibilities for the evolution of semiconductor manufacturing towards the goal of "zero defects".

2 Chip defect data set and YOLOV8 principle

2.1 Chip Defect Dataset

(1) Data Source:

The data set is collected through a variety of ways, including the public data set kaggle (data science community and competition platform), actual chip defect photography, and laboratory simulated defect images (Figure1). To ensure that the data used has rich diversity and good generalization performance, this dataset specifically includes images collected under different materials and diverse background environmental conditions.

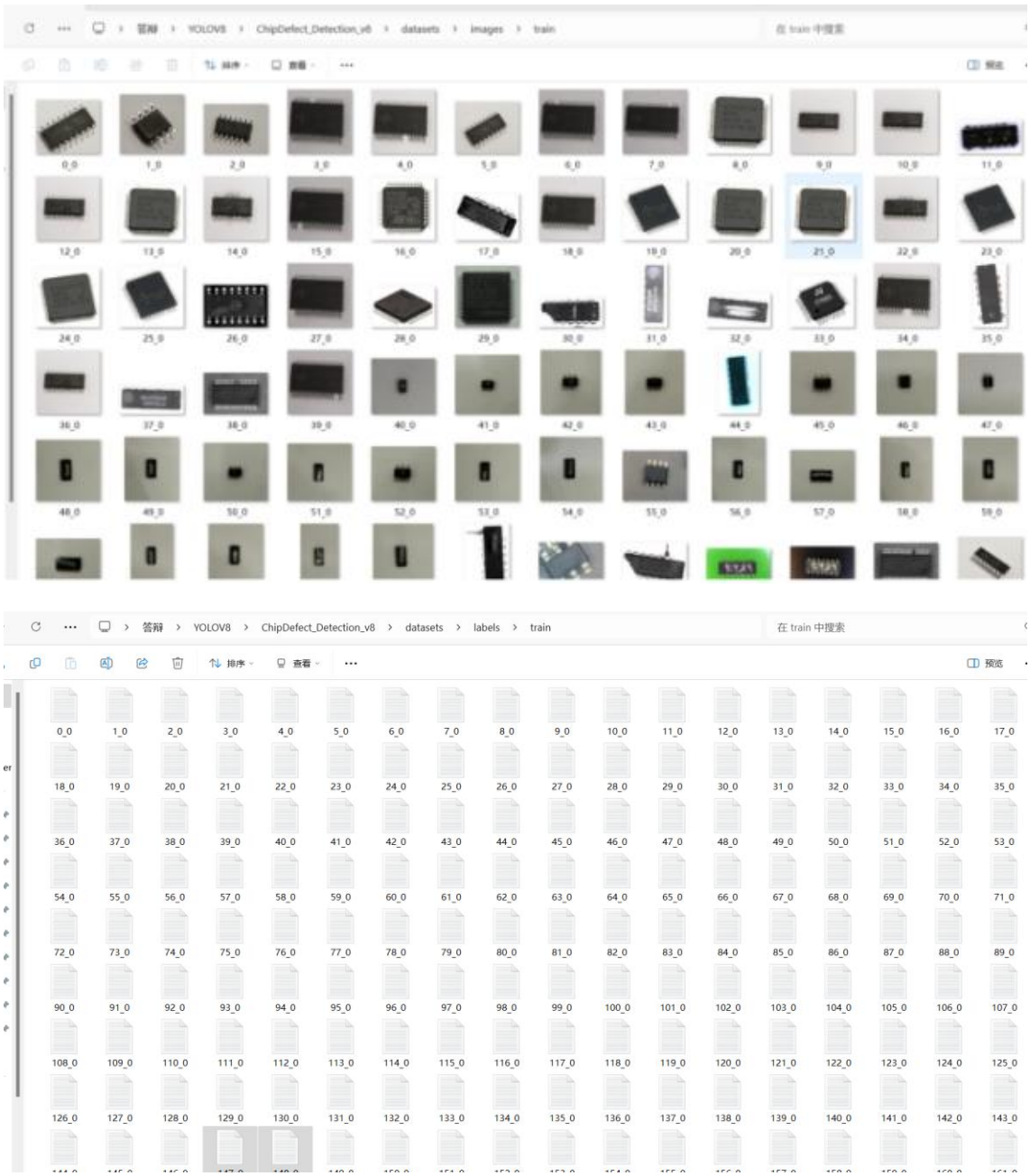

Figure 1: Chip Defect Dataset

(2) Data Annotation:

In each image, the defects present are annotated and explained using bounding boxes. The annotation format used strictly follows the YOLO standard, including category identification (class_id), the horizontal axis of the bounding box center point (*x*_center), the vertical axis of the bounding box center point (*y*_center), the width of the bounding box, and the height of the bounding box.

Annotation tools: LabelImg or CVAT.

(3) Annotation File:

Convert the marked data to YOLO format. The YOLO annotation format is presented in units of each row, with the specific content being:<Target Category><Boundary Box Center Point x Coordinate><Boundary Box Center Point y Coordinate><Boundary Box Width> <Boundary Box Height>. Within this framework, the variable class_id represents a specific

category identifier. The X_center parameter represents the horizontal coordinates of the center point of the bounding box in the image, while y_center specifies its vertical coordinates. The width and height parameters respectively describe the horizontal and vertical ranges of the bounding box, all of which are limited to the range of 0-1. Considering the width and height of the image, the center point coordinates of the bounding box will be normalized. Similarly, the width and height of the bounding box have been normalized relative to the size of the image. It is worth noting that these normalized coordinate values maintain a direct proportional relationship with the true size of the image. In addition, each image is associated with a dedicated. txt file designed to archive the corresponding annotation data.

Characteristics of the dataset:

(4) Diversity: The dataset contains defect images under various materials and background conditions, covering multiple defect types (such as scratches, stains, packaging damage, poor pin connections, etc.).

(5) Challenging: There are various interfering elements in some images, such as complex background environments, dynamic changes in lighting intensity, and mutual occlusion between objects. The introduction of these interfering factors aims to improve the recognition accuracy of the model in complex real-world scenarios.

(6) Balance: The division ratio between the training set, validation set, and test set is scientifically reasonable. This arrangement can ensure that the model fully absorbs data features for learning during the training phase, effectively evaluates its own performance during the validation phase, and demonstrates good generalization ability during the testing process.

2.2 YOLOV8 Neural Network

YOLOv8 is the latest development in the YOLO (You Only Look Once) object detection algorithm lineage, carefully crafted by the Ultralytics team (please refer to Figure 2 for comprehensive insights) [6, 7]. Due to its efficient single-stage detection framework, the YOLO series algorithms have been widely used in real-time object detection applications. YOLOv8 represents a cutting-edge algorithm engineered by the Ultralytics team, specifically tailored for real-time object detection tasks, and it signifies the most recent progression within the YOLO algorithm family. This innovative approach incorporates several distinctive characteristics [8]:

(1) Efficient: Utilizing advanced network architecture and optimization techniques to achieve fast inference speed while maintaining high accuracy, meeting real-time detection requirements. For example, real-time chip defect detection [9].

(2) Accuracy: Able to accurately locate and recognize targets in various object detection tasks, demonstrating strong detection performance for complex scenes such as small objects and occluded targets. In industrial quality inspection, it can accurately detect small defects.

(3) Multifunctionality: Suitable for multiple fields such as safety monitoring, autonomous driving, medical image analysis, agricultural monitoring, etc., it has excellent adaptability to various scenarios and target types.

The network architecture adopts [10, 11]:

(1) Backbone network: Adopting CSP (Cross Stage Partial) and other structures, effectively extracting image features, enhancing feature propagation and fusion, improving feature extraction capabilities and model computational efficiency.

(2) Neck network: Utilizing structures such as PAN (Path Aggregation Network) to enhance information exchange between feature maps of different scales, enabling the model to better utilize multi-scale features for object detection.

(3) Head network: Using a decoupled head structure to separate classification and

regression tasks improves the accuracy and flexibility of model detection, enabling more accurate prediction of target categories and positions.

It abandons traditional anchor-based detection methods, directly predicts targets, reduces hyperparameter settings, improves detection speed and accuracy, while reducing model complexity and computational costs.

The object detection task is decomposed into sub tasks such as classification, localization, and regression, each of which is processed independently to make the model learning more targeted and improve detection performance [12, 13]. It more accurately measures the difference between the predicted bounding box and the ground truth bounding box, optimizes the model training process, and improves convergence speed and detection accuracy.

YOLOv8 demonstrates outstanding performance compared to YOLOv5 in terms of both accuracy and speed, with a significant improvement in MAP values over its predecessor and a substantial acceleration in inference speed [14, 15]. It operates efficiently across different hardware platforms, though further enhancement in precision is still required.

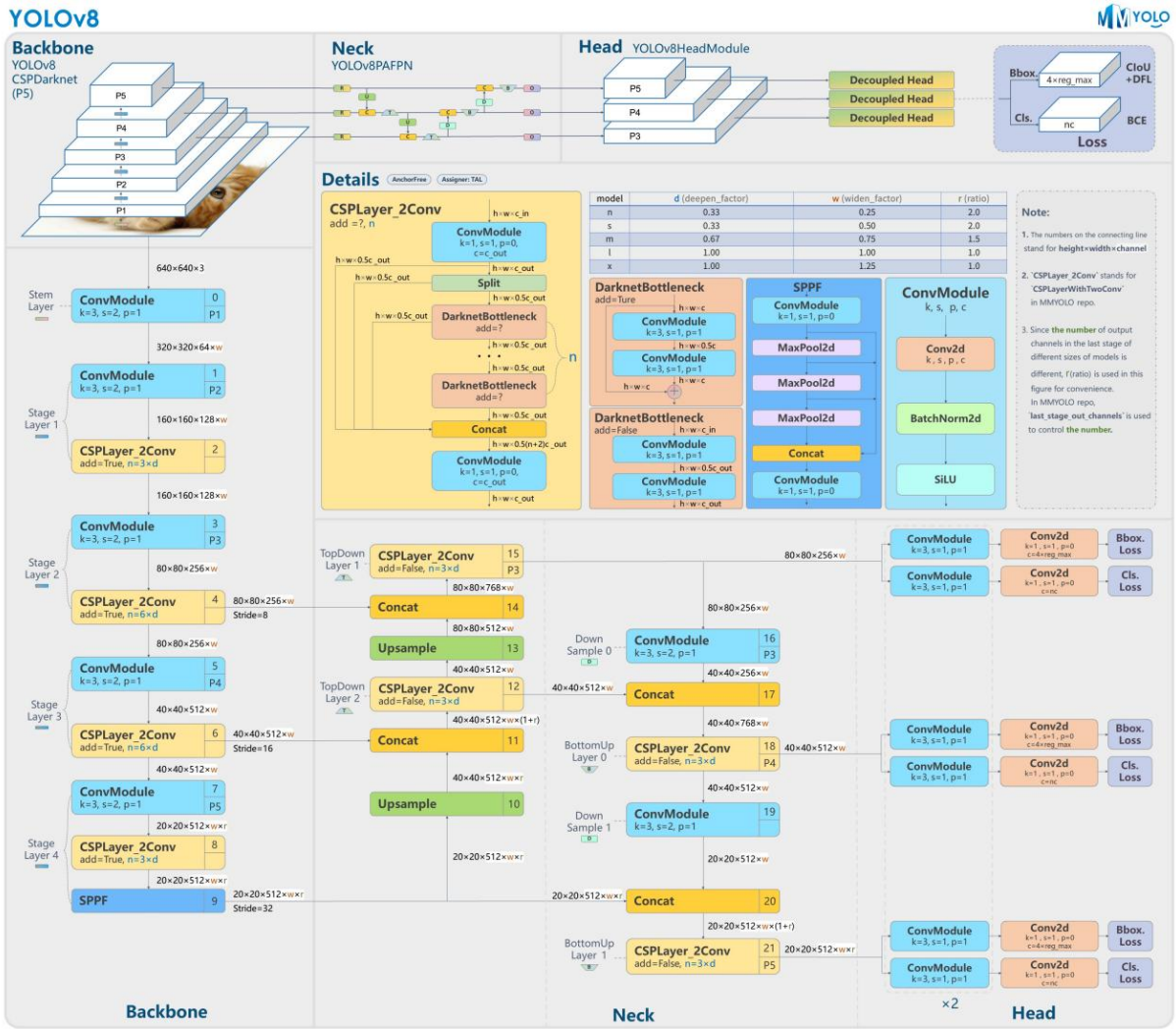

Figure 2: YOLOv8 Network Architecture Diagram

3 Model Optimization

3.1 Data Augmentation

As a widely used technique, the core goal of data augmentation [16, 17] is to improve the accuracy and generalization ability of object detection models. This technology utilizes diverse transformation operations and expansion processing on initial training data to simulate complex and dynamically changing environmental conditions in the real world, thereby enhancing the model's adaptability to changes in factors such as object size, observation perspective, and lighting conditions. Given the unique properties of the YOLO algorithm, the following data augmentation methods have shown particularly significant results [18, 19]:

(1) Random scaling: Performing random enlargement or reduction operations on an image to simulate changes in the distance between the object and the observation point. In the specific implementation process of YOLO, it is necessary to select a random scaling factor, and at the same time, the image and its corresponding object bounding box annotation information should be scaled according to the same scale to ensure that the annotation content matches the transformed image [20].

(2) Random region cropping: Randomly select a sub region from the original image for cropping operation, in order to simulate the possible occurrence of objects in different positions of the image. When using this method, it is necessary to synchronously adjust the bounding box coordinates of the objects within the cropped area to ensure the accuracy of the annotation information.

(3) Random mirror transformation: flipping the image horizontally or vertically with a certain probability to enhance the model's anti-interference ability against changes in object direction. After completing the flipping operation in YOLO, it is necessary to adjust the coordinates of the object bounding box accordingly [21].

(4) Random Rotation: Rotate the image around its center point by a random angle to simulate variations in shooting perspective. When implementing rotation augmentation, it is necessary to accurately calculate the new coordinates of the object's bounding box on the rotated image and update them accordingly.

3.2 Optimized Attention Modules

YOLOv8 departs from the anchor-box-dependent prediction paradigm that characterized prior iterations of the YOLO series, a methodology traditionally employed to ascertain the spatial positioning and dimensions of anchor boxes within imagery. In lieu of this conventional approach, YOLOv8 pioneers an anchor-free detection methodology, enabling direct prediction of both the central coordinates and the aspect ratio of target objects. This innovative strategy substantially diminishes the reliance on a multitude of predefined anchor boxes, thereby streamlining the detection process. Compared to YOLOv5, which adopts a coupling head structure, YOLOv8 innovatively utilizes decoupling head design to separate the head structures corresponding to classification and detection tasks [22, 23]. Specifically, YOLOv8 has removed the subjectivity branch and only retained the classification and regression branches. Moreover, YOLOv8 has made significant changes in object detection methods, shifting from anchor-based methods to anchor free methods. In this anchor free method, the position of the target is uniquely determined by its center point, and the prediction process mainly revolves around estimating the distance from the target center to its boundary.

The attention mechanism equips the model with dynamic adaptability, allowing it to selectively concentrate on pivotal information by assigning feature weights in accordance with the distinctive attributes of the input data. The CBAM (Convolutional Block Attention

Module) represents a streamlined, hybrid attention mechanism that substantially augments the model's feature filtering and extraction capabilities. This enhancement is achieved through the collaborative interaction of dual-path attention mechanisms operating along both channel and spatial dimensions. CBAM has significant core advantages in multiple aspects: from the perspective of computational efficiency, compared to traditional attention modules such as SE Net, it only increases the number of model parameters by about 1.2% [24, 25], while ensuring performance improvement and occupying less computing resources.

As shown in Figure 3, the specific structure of CBAM is presented. In the operation process of the CBAM attention module, the input feature map (i.e. input features) will be first passed into the module system. Subsequently, these input feature maps undergo pooling operations to generate average pooling features (also known as average pooling) and max pooling features (also known as max pooling). Next, the generated features will be further processed using a shared multi-layer perceptron to refine and optimize the feature map. Afterwards, the input feature map is multiplied with the refined feature map to obtain the intermediate feature map. Similarly, f1 in the intermediate feature map will enter another pooling process, during which cross channel average pooling features (average pool) and maximum pooling features (maximum pool) will be generated. Finally, after 7x7 convolution kernel processing, a two-dimensional attention map (labeled as Ms) can be obtained.

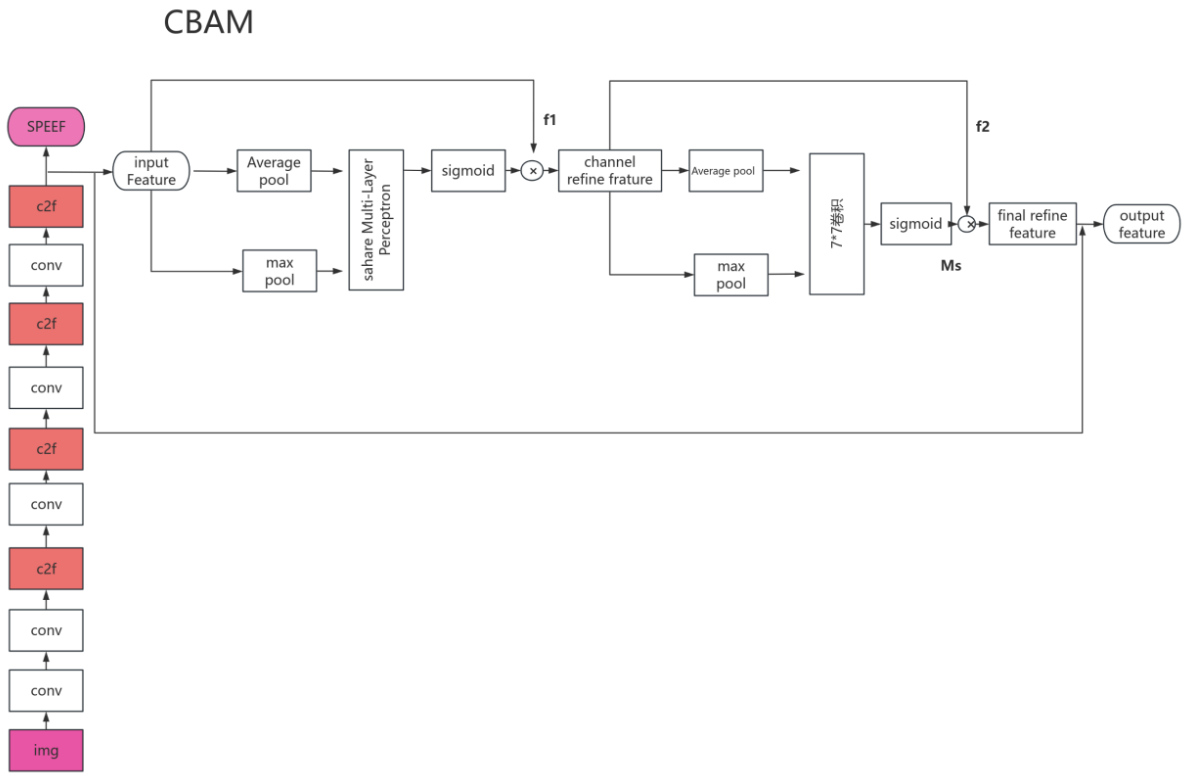

Figure 3: Backbone Network Based on CBAM

In recent years, attention mechanisms have shown outstanding performance optimization results in the field of object detection. When deep neural networks are integrated into the attention module, they can automatically focus on key feature regions in the input image based on the actual situation, and efficiently remove irrelevant background information [26].

This study innovatively proposes an enhancement strategy specifically designed for the YOLOv8 framework, with the key being the clever embedding of advanced attention

mechanism units into the neck structure of the network. Specifically, the Convolutional Block Attention Module (CBAM) has attracted much attention for its outstanding performance in numerous visual recognition tasks, as it strategically connects the outputs of three key C2f components in the network. This unique architecture design fully integrates the advantages of channel attention mechanism and spatial attention mechanism, and constructs a powerful dual attention system. With the synergistic effect of these two attention paths, the model has significantly improved its ability to distinguish features at different scales. As a result, the overall performance indicators of the model have been comprehensively optimized, with particularly significant performance in object detection scenarios.

In the dual attention system, channel attention mainly undertakes the task of readjusting the weight allocation of feature channels, while spatial attention focuses on accurately locating the most recognizable spatial regions. The two complement each other, allowing the network to maintain stable and efficient object detection performance even in complex scenarios.

$$F1=MC(F\ input)\times F\ input \tag{1}$$

$$F2=Ms(F1)\times F1 \tag{2}$$

Among them, F1 is the channel optimization feature map, MC (F input) is the channel attention feature map, F2 is the final output feature map, and Ms is the two-dimensional space attention map.

After completing the nonlinear transformation, the Sigmoid (σ) activation function is used to perform element level summation and normalization on the output results, ultimately successfully obtaining the channel attention weights. The mathematical expression corresponding to this process is presented as follows [27]:

$$\mathrm{Mc} = \mathrm{MLP}(\mathrm{GMP}(\mathrm{F}))] + \sigma[\mathrm{MLP}(\mathrm{GAP}(\mathrm{F})) \tag{3}$$

Within the scope of spatial attention mechanism, CBAM utilizes the operation methods of global average pooling (GAP) and global maximum pooling (GMP) along the channel axis direction to extract spatial level feature information from input data. After this operation, two feature maps containing key region information can be obtained. Next, these feature maps are concatenated along the channel dimension and then input into a 7×7 convolutional layer for feature fusion [28]. Finally, the fused features are processed using the Sigmoid (σ) activation function to generate spatial attention weights. The mathematical expression corresponding to this process is presented as follows:

$$\mathrm{Ms}(\mathrm{F}) = \sigma[\mathrm{f}^{(7*7)}(\mathrm{GMP}(\mathrm{F}), \mathrm{GAP}(\mathrm{F}))] \tag{4}$$

4 Comparative Experimental Analysis

4.1 Data Processing

We allocated the dataset using random partitioning, with the training set accounting for 70%, validation set accounting for 20%, and test set accounting for 10%. In terms of specific quantity, the training set contains 7000 images, accounting for 70.0% of the overall dataset; The validation set consists of 2000 images, accounting for 20.0%; The test set contains 1000 images, accounting for 10.0%. Due to the narrow range of brightness changes in the chip surface images in the training set, if the model is trained solely based on these images, its

performance in predicting chip surface images under different brightness conditions may not be satisfactory. At the same time, there is also an issue of imbalanced class distribution in the dataset. To effectively address the above issues, we have decided to use data augmentation techniques to expand the size of the training set, in order to enhance the model's generalization ability and processing ability for different categories of data.

4.2 Target Performance Metrics

In deep learning detection, the performance index of target detection is extremely important, which reflects the accuracy of model detection. When conducting performance evaluation of object detection, a series of indicators are generally considered according to the following process:

(1) Accuracy, recall

The confusion matrix, as a crucial analytical tool, can provide a comprehensive and systematic analysis of the prediction results of classification problems. It provides a comprehensive summary and scientific conclusion about the prediction results by accurately counting the number of correct and incorrect predictions in each category. This matrix has unique advantages in accurately indicating which categories are prone to confusion when predicting in classification models. From the structural composition of the matrix, it adopts an intuitive presentation method: rows (corresponding to the y-axis direction) represent the predicted categories made by the model, and columns (corresponding to the x-axis direction) represent the actual categories to which the samples belong. The following is the specific introduction content:

Thereinto:

A designation of TP (True Positives) signifies that the model yields positive forecasts, and the real-world sample counts align with true instances. In essence, this metric quantifies the count of positive samples that the model has correctly pinpointed.

FN (False Negatives) denotes the quantity of situations in which the model outputs negative predictions, despite the actual samples being positive. This indicates that the model has erroneously categorized positive samples as negative ones.

FP (False Positives) pertains to the number of instances where the model projects a positive outcome, yet the actual sample is negative. It captures the scenario where the model mistakenly labels negative samples as positive.

TN (True Negatives) indicates the frequency with which the model predicts negative results, and the actual sample is indeed negative. In other words, this metric corresponds to the number of negative samples that the model has accurately predicted.

	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

When evaluating the performance of classification models, accuracy and recall are two commonly used metrics, calculated as follows:

Precision:

formula:Precision = TP / (TP + FP)

Explanation: Precision is the proportion of all samples predicted by the model as positive (Positive) that are actually positive. It measures the accuracy of the model in predicting positive cases.

Recall:

formula:Recall = TP / (TP + FN)

Explanation: Recall is the percentage of all samples that are actually positive that the model successfully predicts as positive. It measures the model's ability to recognize positive examples.

The use of a confusion matrix helps to intuitively understand the types of errors in a classification model, particularly whether the model confuses two different categories or misclassifies one category as another [29]. This detailed analysis helps overcome the limitations of relying solely on classification accuracy, as illustrated in Figure 4, which clearly and intuitively shows the model's prediction accuracy.

(2) To comprehensively and systematically evaluate the performance of different algorithms, the concept of F1 value is introduced based on the comprehensive consideration of precision and recall, aiming to achieve a collaborative evaluation of precision and recall at the overall level. The definition of F1 value is as follows:

$$F1 = 2 * [(recall * precision)/(recall + precision)] \quad (5)$$

$$F1 = 2TP/(2TP + FN + FP) \quad (6)$$

In the field of performance evaluation for multi class classification problems, F1 curve is an extremely commonly used and practical tool, and has a wide range of applications in various competition scenarios. The curve is based on the F1 score, which is essentially the harmonic mean of precision and recall, with a numerical range between 0 and 1. Among them, a value of 1 represents that the model has reached its optimal performance state, while a value of 0 means that the model's performance is at its worst level.

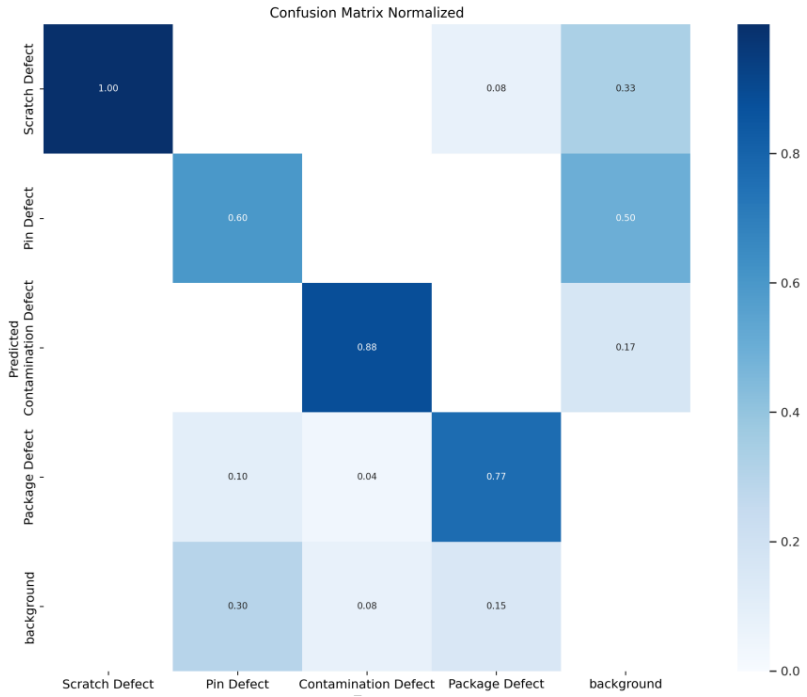

Figure 4: Confusion Matrix

In practical operation, we usually observe the dynamic changes of the F1 curve by adjusting the confidence threshold (i.e. the probability value corresponding to the model's judgment of the sample as a specific category). When the set confidence threshold is at a low level, the model is prone to mistakenly judge a large number of samples with low confidence as true. In this case, the recall rate will be improved, but the accuracy will correspondingly decrease. On the contrary, when the confidence threshold is set high, only those samples with extremely high confidence will be judged as true by the model. In this way, the model's classification of categories will be more accurate, and the precision will also be improved [30].

From an ideal perspective, taking the F1 curve presented in Figure 5 as an example, when the confidence interval is within the range of 0.4-0.6, the model can obtain a more ideal F1 score. This fully demonstrates that within this confidence interval, the model exhibits good performance in balancing accuracy and recall.

(3) P_curve

In the PCC diagram, the horizontal axis (*x*-coordinate) represents the confidence level set by the detector, while the vertical axis (*y*-coordinate) corresponds to accuracy (or recall). The shape and position of the curve can intuitively reflect the performance of the detector when facing different confidence levels.

In terms of the curve characteristics in the PCC graph, if the curve shows an upward and leftward bending trend, it means that the detector can still maintain high accuracy under relatively low confidence level conditions. This further indicates that the detector can control the false alarm rate at a lower level while ensuring a high recall rate, which means that the recognition accuracy of the target is better.

On the contrary, if the curve shows a downward and rightward bending trend, it means that the detector can only achieve higher accuracy under higher confidence conditions. Although such situations may lead to an improvement in detection rate, overall performance highlights the unsatisfactory performance of the detector.

From this, it can be seen that PCC plots can provide valuable information for evaluating the performance of detectors at different confidence levels. In Figure 6, under normal circumstances, the detector should present an upward and leftward curved curve, while a downward and rightward curved curve indicates that there is room for optimization and improvement in the detector.

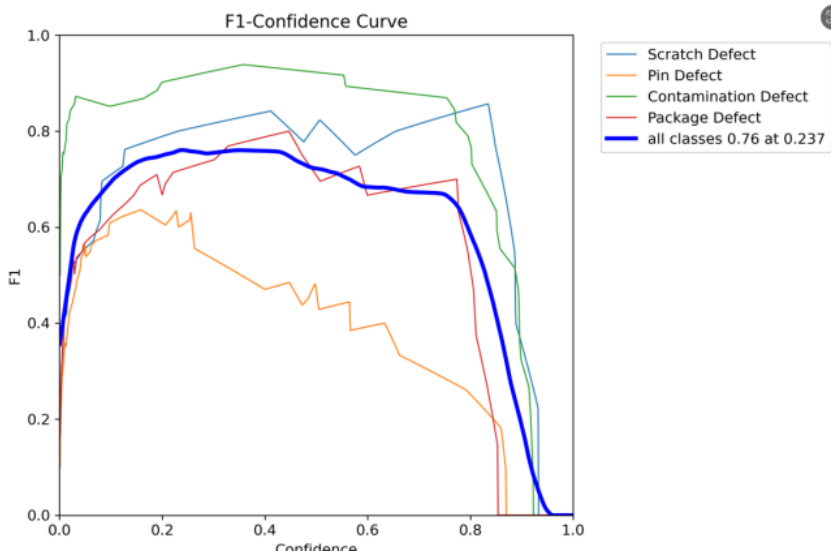

Figure 5: F1_curve

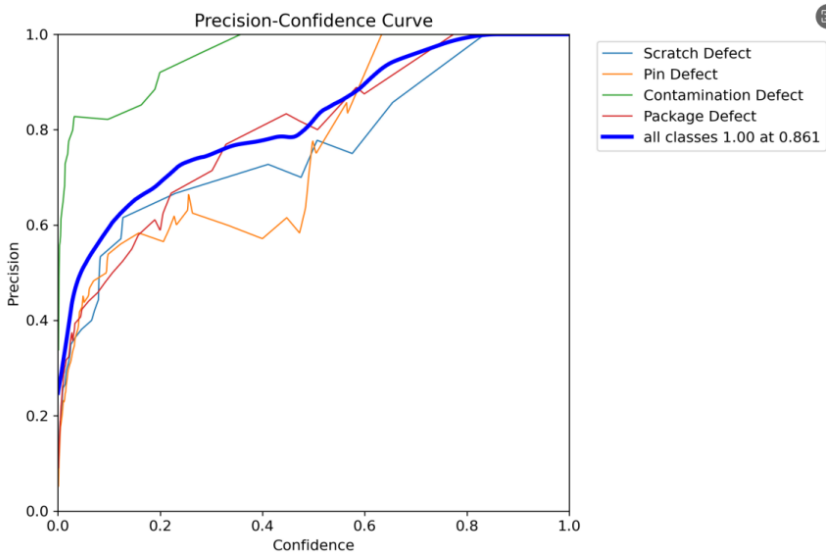

Figure 6: P_curve.png

(4) R_curve.png

It must be recognized that the gradient or steepness of the curve in the RCC graph is intricately related to the performance of the model. More precisely, the steeper slope of the curve corresponds to a higher recall achieved after eliminating prediction boxes with low confidence scores, greatly improving the detection capability of the model.

As shown in the figure, the closer the curve (as shown in Figure 7) is to the upper right quadrant, the better the performance of the model. When the curve asymptotically approaches the upper right corner of the graph, it means that the model can maintain high recall while ensuring high accuracy. Therefore, the RCC graph has become a powerful tool for overall evaluation of model performance and facilitating the selection of the optimal threshold to achieve a harmonious balance between recall and accuracy.

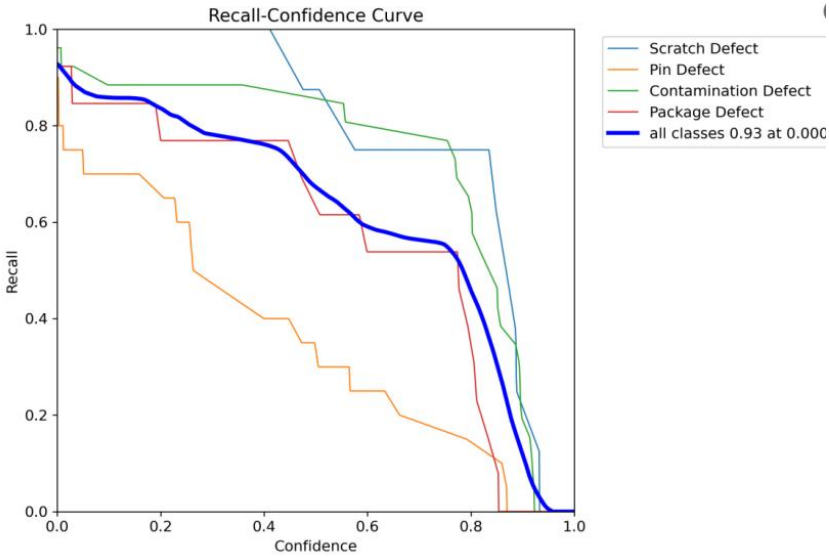

Figure 7: R_curve.png

(5) PR_curve.png

The precision recall (PR) curve is a graphical tool used to illustrate the relationship between precision and recall indicators. In this case, accuracy is defined as the proportion of truly positive samples among the samples predicted as positive by the model. On the contrary, recall rate represents the proportion of actual positive samples correctly identified by the model as positive.

Within the coordinate framework of the PR curve, recall is plotted along the horizontal axis (x-axis), while precision is represented along the vertical axis (y-axis). Usually, there is an inverse relationship between these two indicators: as the recall rate increases, accuracy often decreases, and vice versa. Therefore, the PR curve visually summarizes the inherent trade-off between accuracy and recall.

The positioning of the PR curve in the graph provides valuable insights into the performance of the model. The curve near the upper right corner of the graph indicates that the model can achieve both high accuracy and high recall in the given task, demonstrating that its predictions are both accurate and reliable. In sharp contrast, the curve located near the bottom left corner means that the model struggles to balance high accuracy with high recall, reflecting lower prediction accuracy.

In practical applications, the PR curve (as shown in Figure 8) is often used in conjunction with the ROC curve. Through this comprehensive analysis, the performance of the classification model can be evaluated more comprehensively and deeply. The PR curve can provide a more detailed and detailed analysis perspective for the performance of the model in different task scenarios.

(6) results.png

In the field of object detection tasks, the loss function plays a crucial role. Its main responsibility is to quantify the difference between the predicted values of the model and the actual ground truth. The magnitude of this difference has a direct and substantial impact on the performance of the model.

It is crucial to closely monitor fluctuations in accuracy and recall during model training. In addition, for example mAP@0.5 A comprehensive evaluation of the training results should be considered using mAP @ [.5:95]. These indicators can provide valuable insights into model performance and generalization ability, promoting better optimization and improvement of the model. The experimental results are shown in Figure 9.

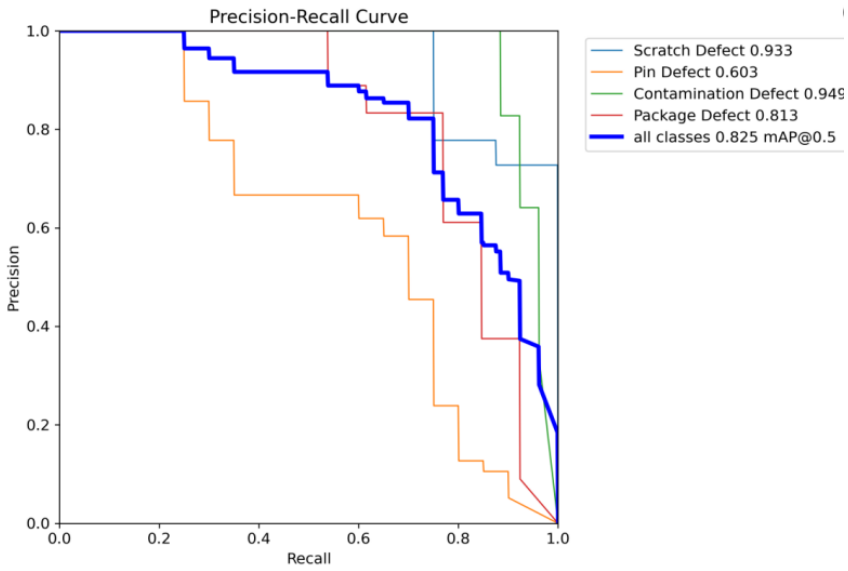

Figure 8: PR_curve.png

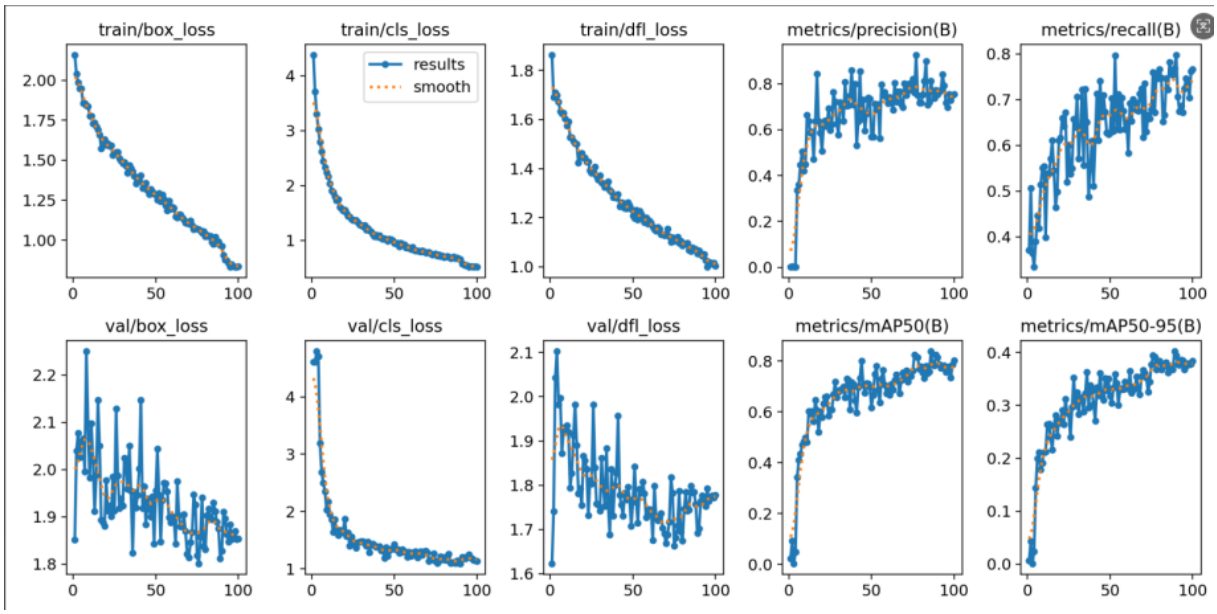

Figure 9: Loss Function

5 Conclusion

This system enhances the detection performance of YOLOv8 by incorporating CBAM (Convolutional Block Attention Module) on top of YOLOv8, enabling it to accurately identify chip defects in complex backgrounds. It achieves a real-time detection accuracy of 98.5% in complex industrial environments, improving mAP by approximately 20% compared to traditional methods. The system also supports real-time detection and is suitable for various industrial inspection applications. However, the stability of defect classification under extreme lighting conditions still needs improvement. Future plans include integrating self-supervised pre-training techniques to further reduce the need for labeled data. This research provides a practical technical pathway for the intelligent upgrade of the semiconductor industry. The study offers an efficient and reliable solution for automated quality inspection in chip manufacturing, and its methodology can serve as a reference for other high-precision industrial vision tasks.

References

[1] Ragab M G, Abdulkadir S J, Muneer A, et al. A comprehensive systematic review of YOLO for medical object detection (2018 to 2023)[J]. IEEE Access, 2024, 12: 57815-57836.

[2] Kang S, Hu Z, Liu L, et al. Object detection YOLO algorithms and their industrial applications: Overview and comparative analysis[J]. Electronics, 2025, 14(6): 1104.

[3] Kang M, Ting C M, Ting F F, et al. ASF-YOLO: A novel YOLO model with attentional scale sequence fusion for cell instance segmentation[J]. Image and Vision Computing, 2024, 147: 105057.

[4] Sapkota R, Flores-Calero M, Qureshi R, et al. YOLO advances to its genesis: a decadal and comprehensive review of the You Only Look Once (YOLO) series[J]. Artificial

Intelligence Review, 2025, 58(9): 1-83.

[5] Badgujar C M, Poulose A, Gan H. Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review[J]. Computers and Electronics in Agriculture, 2024, 223: 109090.

[6] Ali M L, Zhang Z. The YOLO framework: A comprehensive review of evolution, applications, and benchmarks in object detection[J]. Computers, 2024, 13(12): 336.

[7] Su P, Han H, Liu M, et al. MOD-YOLO: Rethinking the YOLO architecture at the level of feature information and applying it to crack detection[J]. Expert Systems with Applications, 2024, 237: 121346.

[8] Chen Y, Yuan X, Wang J, et al. YOLO-MS: rethinking multi-scale representation learning for real-time object detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2025.

[9] Flores-Calero M, Astudillo C A, Guevara D, et al. Traffic sign detection and recognition using YOLO object detection algorithm: A systematic review[J]. Mathematics, 2024, 12(2): 297.

[10] Bai Y, Yu J, Yang S, et al. An improved YOLO algorithm for detecting flowers and fruits on strawberry seedlings[J]. Biosystems Engineering, 2024, 237: 1-12.

[11] Zhang Y, Zhang H, Huang Q, et al. DsP-YOLO: An anchor-free network with DsPAN for small object detection of multiscale defects[J]. Expert Systems with Applications, 2024, 241: 122669.

[12] Liu Z, Abeyrathna R M R D, Sampurno R M, et al. Faster-YOLO-AP: A lightweight apple detection algorithm based on improved YOLOv8 with a new efficient PDWConv in orchard[J]. Computers and Electronics in Agriculture, 2024, 223: 109118.

[13] Sangaiah A K, Yu F N, Lin Y B, et al. UAV T-YOLO-rice: An enhanced tiny YOLO networks for rice leaves diseases detection in paddy agronomy[J]. IEEE Transactions on Network Science and Engineering, 2024, 11(6): 5201-5216.

[14] Zhu L, Li X, Sun H, et al. Research on CBF-YOLO detection model for common soybean pests in complex environment[J]. Computers and Electronics in Agriculture, 2024, 216: 108515.

[15] Wang S, Li Y, Qiao S. ALF-YOLO: Enhanced YOLOv8 based on multiscale attention feature fusion for ship detection[J]. Ocean Engineering, 2024, 308: 118233.

[16] Moussaoui H, Akkad N E, Benslimane M, et al. Enhancing automated vehicle identification by integrating YOLO v8 and OCR techniques for high-precision license plate detection and recognition[J]. Scientific Reports, 2024, 14(1): 14389.

[17] Ahmed A, Imran A S, Manaf A, et al. Enhancing wrist abnormality detection with yolo: Analysis of state-of-the-art single-stage detection models[J]. Biomedical Signal Processing and Control, 2024, 93: 106144.

[18] Ahmed A, Imran A S, Manaf A, et al. Enhancing wrist abnormality detection with yolo: Analysis of state-of-the-art single-stage detection models[J]. Biomedical Signal Processing and Control, 2024, 93: 106144.

[19] Han Y, Guo J, Yang H, et al. SSMA-YOLO: a lightweight YOLO model with enhanced feature extraction and Fusion capabilities for drone-aerial ship image detection[J]. Drones, 2024, 8(4): 145.

[20] Zhang Z, Chen P, Huang Y, et al. Railway obstacle intrusion warning mechanism integrating YOLO-based detection and risk assessment[J]. Journal of Industrial Information Integration, 2024, 38: 100571.

[21] Park S, Kim J, Wang S, et al. Effectiveness of image augmentation techniques on non-protective personal equipment detection using YOLOv8[J]. Applied Sciences, 2025, 15(5): 2631.

[22] Murat A A, Kiran M S. A comprehensive review on YOLO versions for object detection[J]. Engineering Science and Technology, an International Journal, 2025, 70: 102161.

[23] Stefenon S F, Seman L O, Klaar A C R, et al. Hypertuned-YOLO for interpretable distribution power grid fault location based on EigenCAM[J]. Ain Shams Engineering Journal, 2024, 15(6): 102722.

[24] Eum I, Kim J, Wang S, et al. Heavy equipment detection on construction sites using you only look once (YOLO-Version 10) with transformer architectures[J]. Applied Sciences, 2025, 15(5): 2320.

[25] Shi Y, Li S, Liu Z, et al. MTP-YOLO: You only look once based maritime tiny person detector for emergency rescue[J]. Journal of Marine Science and Engineering, 2024, 12(4): 669.

[26] Li Y, Yin C, Lei Y, et al. RDD-YOLO: Road damage detection algorithm based on improved You Only Look Once version 8[J]. Applied Sciences, 2024, 14(8): 3360.

[27] Karimi N, Mishra M, Lourenço P B. Automated surface crack detection in historical constructions with various materials using deep learning-based YOLO network[J]. International Journal of Architectural Heritage, 2025, 19(5): 581-597.

[28] Zhou Z, Hu Y, Yang X, et al. YOLO-based marine organism detection using two-terminal attention mechanism and difficult-sample resampling[J]. Applied Soft Computing, 2024, 153: 111291.

[29] Kang M, Ting C M, Ting F F, et al. Bgf-yolo: Enhanced yolov8 with multiscale attentional feature fusion for brain tumor detection[C]//International Conference on Medical Image Computing and Computer-Assisted Intervention. Cham: Springer Nature Switzerland, 2024: 35-45.

[30] Giri K J. SO-YOLOv8: A novel deep learning-based approach for small object detection with YOLO beyond COCO[J]. Expert Systems with Applications, 2025, 280: 127447.