IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 September 2024, accepted 25 September 2024, date of publication 1 October 2024, date of current version 11 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3471690

METHODS

CDF-YOLOv8: City Recognition System Based on Improved YOLOv8

P. LU[1], Y. S. JIA[1], W. X. ZENG[2], AND P. WEI[1]
[1]Maritime College, Tianjin University of Technology, Tianjin 300384, China
[2]School of Electric Engineering and Automation, Tianjin University of Technology, Tianjin 300384, China
Corresponding author: P. Wei (weipeng0925@sina.com)

ABSTRACT To address challenges in urban traffic management, especially detection under low exposure conditions and image quality degradation caused by weather factors, this paper proposes an urban detection algorithm based on the YOLOv8 model. Initially, The idea introduced the Chain-of-Thought Prompt Adaptive Enhancer (CPA-enhancer) to enhance image processing capabilities to cope with unknown image quality degradation. Secondly, the lightweight and efficient dynamic module DySample replaces the original upsampling module, boosting the model's upsampling capability. Furthermore, YOLOv8's Spatial Pyramid Pooling Fast (SPPF) was replaced with FocalModulation to enhance feature processing, particularly for small objects. Finally, experimental results show that compared to the original model, our enhanced algorithm achieved significant improvements in the precision, recall, and map50, with values of 70%, 53%, and 0.6056 respectively. These improvements represent increases of 4.48%, 8.16%, and 8.12%. Our enhanced algorithm surpasses both YOLOv8 and YOLOv10 in recognizing urban traffic imaging degradation.

INDEX TERMS Noise image recognition, deep learning, YOLOv8, CPA-enhancer, DySample, FocalModulation.

I. INTRODUCTION

Due to the continuous development of deep learning technology, increasingly is artificial intelligence (AI) being applied in urban management [1]. As a crucial component of urban management, computer vision-based image recognition systems have a profound impact on the robustness of smart city management systems [2]. These systems face significant challenges, particularly when image quality deteriorates due to factors like day-night transitions and adverse weather conditions. Such degradation can potentially lead to system failures, thereby affecting the efficiency of urban operations and increasing safety risks. To mitigate these risks, it is essential to develop a model that is not only robust to noise but also capable of maintaining high detection performance. However, the variability and unpredictability of noise types and intensities in urban environments pose significant challenges, making it difficult to create a universally applicable model.

Over the past years, YOLOs have emerged as the predominant paradigm in the field of real-time object detection owing to their effective balance between computational cost and detection performance [3]. The YOLO algorithm has undergone continual revisions and updates, such as YOLOv3 [4], YOLOv7 [5], YOLOv8 [6], YOLOv9 [7], and YOLOv10 [3]. YOLO is a universal SOTA in object detection algorithms and was improved by many researchers to apply in various fields. For example, Bi-PAN-FPN and GhostblockV2 are introduced to improve the neck part in YOLOv8-s. The WiseIoU loss is used as bounding box regression to resolve the problem of UAV Aerial Image Recognition [8]. Another Introduced lightweight convolution SEConv and SPPFE modules and used a transformer prediction head to enhance the recognition ability for Remote Sensing Object Detection and Recognition etc [9]. In summary, YOLO is a general object detection algorithm. When faced with different needs, corresponding improvements can be made to achieve better performance in this field.

Therefore, improving existing mature models to better handle image detection tasks in urban management is a promising approach. Numerous studies have demonstrated that enhanced YOLO models are particularly well-suited for image detection tasks in urban and complex environments. For instance, Manhas proposed YOLO NAS for license plate

The associate editor coordinating the review of this manuscript and approving it for publication was Abdel-Hamid Soliman.

recognition, by integrating YOLO with NAS, The NAS which further optimizes the model, the design is adapted specifically for the task of vehicle plate recognition. Achieving higher capability than basic YOLO [10]. Nafiseh Zarei et al. introduced Fast-Yolo-Rec, a detection network based on YOLO combined with an LSTM-based localization prediction network. The incorporation of a Semantic Attention Mechanism, proposed in the Spatial Semantic Attention Module (SSAM), significantly enhances both accuracy and speed, comparable to the latest fast detectors [11]. Wennan Wu and Jizhou Lai have developed a multi-camera positioning technique based on the YOLO object detection algorithm, which utilizes homography transformations for recognizing objects within overlapping areas of camera coverage. This design significantly enhances the positioning accuracy of YOLO object detection [12].

The YOLO algorithm does require a certain level of image quality for optimal performance. Although there have been some studies focused on noise removal, such as the DiffYOLO model proposed by Liu et al., which effectively eliminates specific Gaussian noise [13], image quality degradation is also influenced by factors like lighting conditions, object size and position within the image, and illumination intensity. Additionally, Madhasu and Pande introduced Chrometect GAYO, which utilizes the pix2pix model to convert low-quality images into ones that are more easily recognized by YOLO [14]. However, this approach necessitates the extra training of a pix2pix model and requires its deployment during actual operations, which may affect the image size or cause irreversible damage to the image. Zhou proposed YOLO-NL [15], which introduces the Rep-CSPNet network using a reparameterization method to convert residual convolutions into ghost linear operations, thereby enhancing robustness in challenging scenarios such as dust, dense environments, blurriness, and occlusion. However, it still lacks a universal solution for dealing with unknown image damage.

To address the limitations of the aforementioned studies and develop a more general and stable YOLO model. This model needs to handle image quality degradation caused by unknown factors. It is expected to provide a more robust image detection system for smart city management, meeting the demands of daily urban management and monitoring. By doing so, it reduces the reliance on high-resolution urban surveillance equipment, thereby lowering the cost of urban infrastructure and mitigating the impact of weather conditions such as lighting changes, rain, snow, and fog.

We modified the original YOLOv8 architecture by incorporating the CPA-enhancer module, which can handle unknown image degradation for image enhancement, as proposed by Zhang et al. [16], designed to enhance noise resistance. Subsequently, the Dysample module was developed by Liu et al. [17] replaces the original up-sample module of YOLOv8 to enhance model capabilities. Finally, the original SPPF of YOLOv8 was replaced with FocalModulation Networks developed by Yang et al. [18] to further enhance the model's detection capability, especially in low-light and noisy environments, and for small targets. Through experiments on real-world image datasets, our proposed method effectively addresses the challenges posed by degraded images. Compared to the original YOLOv8 model, our approach demonstrates significant improvements in accuracy, recall, F1 score, and mean Average Precision (mAP). These enhancements offer a robust solution to the image detection issues commonly encountered in urban management, providing a powerful new model for urban image detection tasks and supporting the needs of smart city management systems. This model can better handle various challenges in urban environments.

The contributions of this paper are as follows:

1. Introduction of a Novel Modification Strategy: For the first time, we introduce a viable modification strategy to the YOLOv8 model, enabling it to effectively handle unknown image degradation while also improving the recognition of small objects.
2. High-Accuracy Model for Smart City Object Detection: We provide a high-accuracy model specifically designed for object detection in smart cities, addressing the unique challenges of urban environments.
3. Reference Model Fusion Scheme: Our work offers a model fusion scheme that can serve as a reference for other types of scenarios, demonstrating the flexibility and adaptability of our approach.

The rest of this paper is organized as follows: Section II analyzes the CPA-enhancer, DySample, and Focal Modulation modules in conjunction with YOLOv8. Section III provides a detailed introduction to the proposed method. Section IV discusses the experimental setup, and Section V discusses the experimental results. Finally, Section VI briefly summarizes our findings.

II. YOLO8 NETWORK STRUCTURE ANALYSIS

Building on the success of previous YOLO versions, YOLOv8 introduces new features and improvements, further enhancing its performance and flexibility, and achieving high speed and accuracy. YOLOv8 uses a backbone similar to YOLOv5, but with modifications in the CSPLayer, now called the C2f module. The C2f module consists of a two-convolution cross-stage partial bottleneck, which combines high-level features with contextual information, reducing the likelihood of misjudgment background elements and improving detection accuracy.

YOLOv8 employs an anchor-free model with a decoupled head that independently handles object detection, classification, and regression tasks. This design allows each branch to focus on its specific task, thereby improving the overall accuracy of the model. In YOLOv8's output layer, a sigmoid function is used as the activation function for object scores, indicating the probability that an object is present within the bounding box. The softmax function is employed to represent class probabilities, showing the likelihood of the object belonging to each possible class.

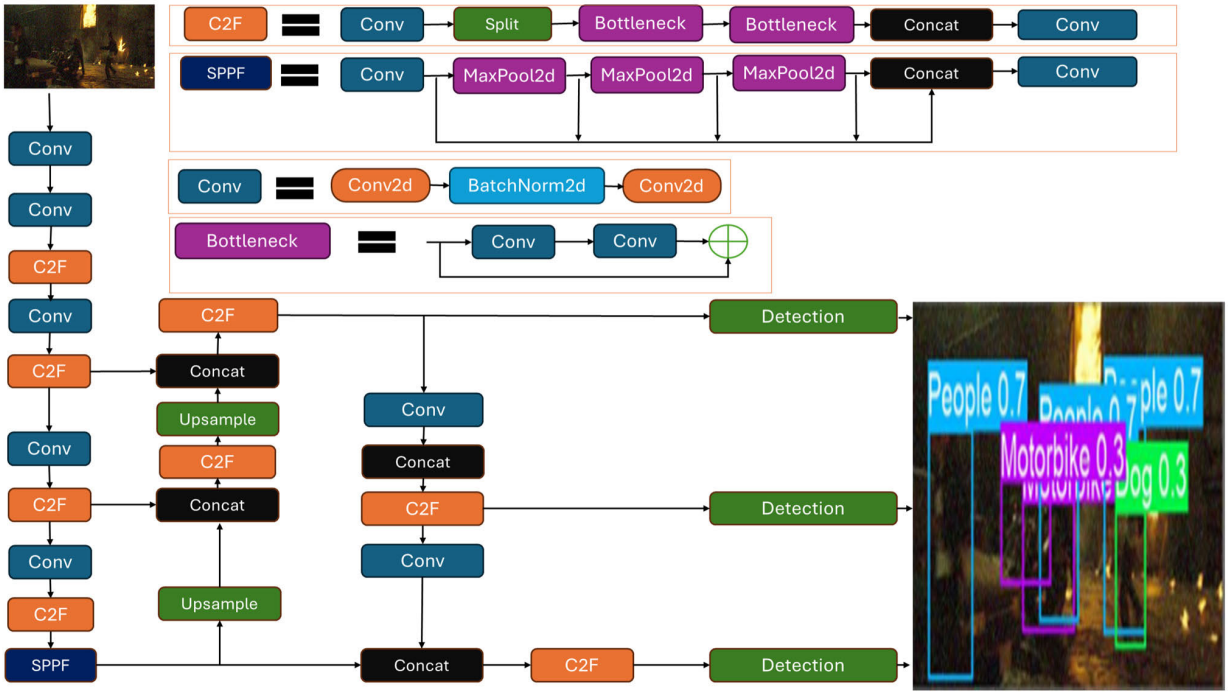

FIGURE 1. YOLOv8 network.

For bounding box loss, YOLOv8 uses Complete Intersection over Union (CIoU) [19] and Distribution Focal Loss (DFL) [20] functions. For classification loss, it adopts binary cross-entropy. These loss functions enhance object detection performance, especially when dealing with smaller objects. In this study, YOLOv8 is chosen as the baseline model, comprising three key components: the backbone network, the neck network, and the prediction output head.

The backbone network is the core part of the YOLOv8 model, responsible for extracting features from the input RGB color image. If noise is present in the image, this noise will also be extracted and processed. The neck network is situated between the backbone network and the prediction output head. Its primary role is to aggregate and process the features extracted by the backbone network. In YOLOv8, the neck network plays a crucial role in integrating features at different scales. Typically, the neck network combines FPN (Feature Pyramid Network) [21] and PAN (Path Aggregation Network) [22] modules, which perform multi-scale feature fusion through the bidirectional integration of low-level and high-level features, enhancing the detection capability of small objects with a smaller receptive field. However, if noise is extracted by the backbone, it can also affect the integration of these features.

The prediction output head is the topmost part of the YOLOv8 model, responsible for identifying and locating objects within the image. YOLOv8 uses three sets of detectors, each operating at a different scale, to help the model recognize objects of various sizes. The architecture of the YOLOv8 network is illustrated in Figure 1.

III. CDF-YOLOv8

In order to effectively detect road information, a model must possess strong anti-interference capabilities while also being able to accurately identify small objects and those partially obscured. To address the challenges of urban management in object detection, we propose the CDF-YOLOv8 model as Figure 2.

Upon analyzing the existing YOLOv8 model, we identified that its backbone plays a critical role in aggregating and processing extracted features. However, it lacks the capability to handle various performance-impacting factors. In the complex urban environment, it is challenging to determine which specific factors might degrade YOLOv8's performance. Therefore, an enhancement is needed to effectively manage these unknown factors.

To achieve this, we introduce a CPA-enhancer module before the original backbone of YOLOv8. The CPA-enhancer module, guided by chain-of-thought (CoT) prompts, dynamically adjusts its enhancement strategies to handle various degradation factors such as noise, lighting conditions, and weather changes. This addition allows YOLOv8 to better manage these factors, thus improving its robustness. The CPA-enhancer module will be detailed in the subsequent sections.

Furthermore, urban environments present complex scenarios where images used for detection often vary in unknown sizes and scales. After feature extraction by the backbone, there remains a need to further enhance the detection of small objects, particularly those affected by noise. Small objects, due to their limited presence in an image, are more susceptible to degradation, making it crucial to improve the model's

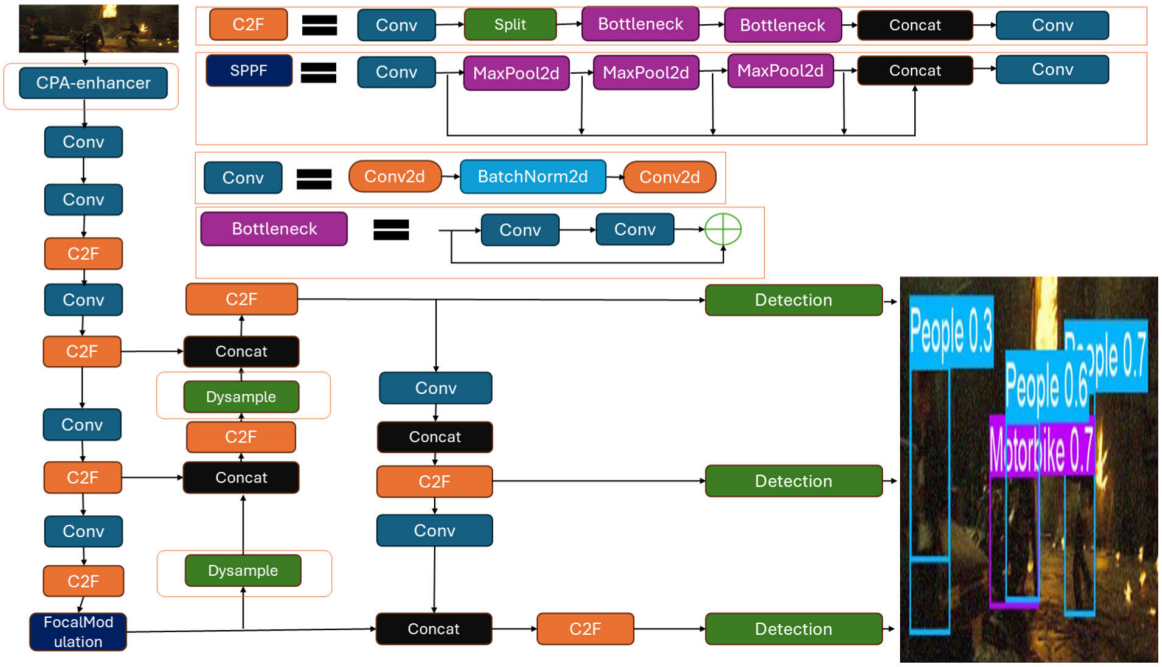

FIGURE 2. The CDF-YOLOv8 network. (Improvements are marked with boxes).

accuracy. The default nearest-neighbor interpolation used in YOLOv8's upsampling process may result in loss of detail and image distortion. DySample, a lightweight and effective dynamic upsampling model, addresses this issue by generating more detailed and smoother edges during the upsampling process, making it an ideal replacement. We will detail the structure and benefits of DySample in the following sections.

Lastly, in image detection tasks, different regions of an image contribute differently to the task. Some regions may contain crucial information for classification or detection, while others may have irrelevant backgrounds. FocalModulation replaces traditional self-attention modules by using a focal modulation mechanism to capture long-range dependencies and contextual information within an image. This approach enhances the model's sensitivity to important features and suppresses irrelevant ones, all without adding significant computational overhead. This improvement is particularly beneficial for detecting small objects in urban scenes. The FocalModulation module will also be discussed in detail in the subsequent sections.

A. CPA-EHANCER

The CPA-enhancer is a Chain-of-Thought Prompted Adaptive Enhancement module. It utilizes a method known as Chain-of-Thought (CoT) [23] prompting, which systematically guides the model through a series of prompts to adjust its enhancement strategies based on the types of degradation inferred. The core components of the CPA-Enhancer include the CoT Prompt Generation Module (CGM) and the Content-Driven Prompt Block (CPB). The CGM is responsible for generating CoT prompts that contain information related to the degradations, while the CPB allows interactions between the input features and the prompts, enabling the model to adapt its enhancement strategies under the guidance of these prompts [16].

The CPA-Enhancer initially utilizes Receptive Field Aware Convolution (RFAConv) to obtain low-level image features [24] consequently, these features are input into a four-level hierarchical encoder-decoder architecture, each level containing a 3×3 RFAConv with a stride of 1. The encoder progressively reduces the spatial size through downsampling operations while increasing the channel capacity, generating low-resolution latent features. Then, the decoder gradually restores high-resolution features from these latent representations. Following this, RFAConv is applied to generate enhanced features, which are then added back to the original image to produce the enhanced image.

CPA-enhancer by channel attention and spatial attention mechanisms are mixed sufficiently to fully capture the features. It is particularly well-suited for dealing with unknown image degradation issues, which is highly beneficial for urban surveillance. The complexity and variability of urban environments, especially due to weather conditions and significant changes in lighting, can lead to varying degrees of degradation in surveillance image quality. This makes the CPA-enhancer an ideal choice for enhancing the robustness and reliability of urban monitoring systems under challenging conditions.

The overall structure of the CPA-Enhancer is shown in Figure 3.

B. DySample

DySample eschews dynamic convolution, adopting a point-sampling approach to construct an upsampler that is

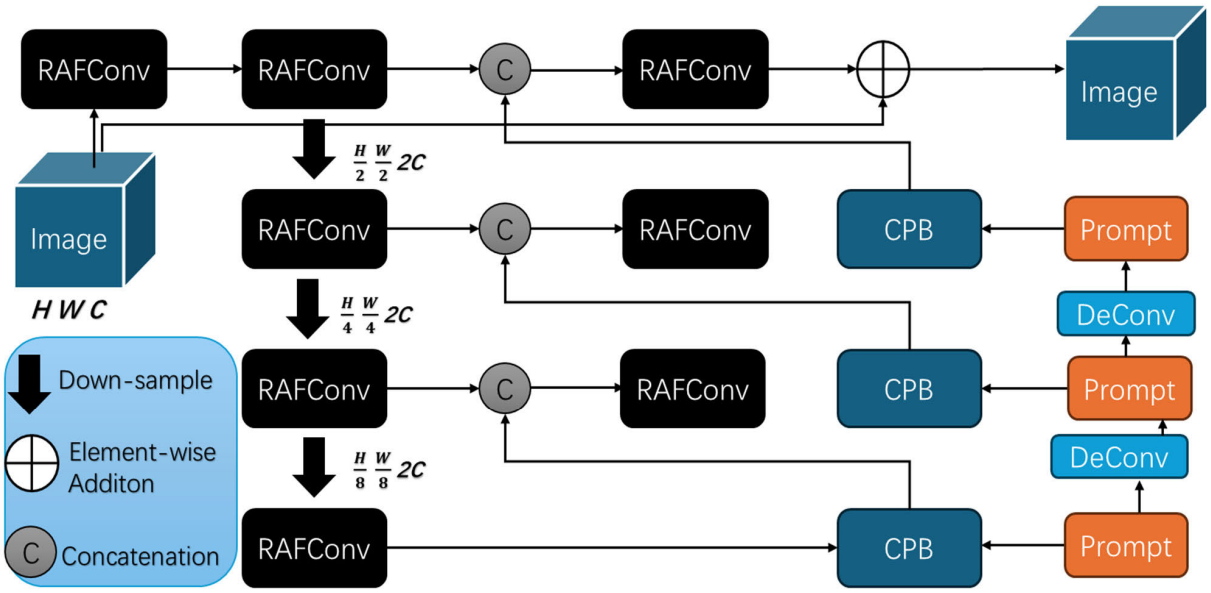

FIGURE 3. The overall structure of the CPA-Enhancer.

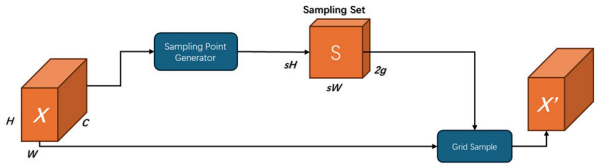

FIGURE 4. Sampling based dynamic upsampling.

more resource-efficient and can be easily implemented using standard built-in functions in PyTorch. This method sidesteps the kernel-based paradigm and returns to point sampling.

Assuming the input feature map is interpolated to a continuous feature map through bilinear interpolation, DySample generates content-aware sampling points to resample this continuous feature map. The point-by-point offsets in DySample are generated through a linear projection and resampled using the grid_sample function in PyTorch. This approach includes improvements such as (i) controlling the initial sampling positions, (ii) adjusting the range of movement for offsets, and (iii) dividing the upsampling process into several independent groups.

Given a feature map X and a sampling set S, grid_sample uses the positions in S to resample the hypothetically bilinear-interpolated X, producing a new feature map X', as shown in Figure 4.

In DySample, given an upsampling scale factor of s and a feature map X, generates offsets O using a linear layer, which is then reshaped by pixel shuffling, an upsampling scale factor of s and a feature map X. First, The inputs and output channel numbers of a linear layer are C and $2s^2$, respectively. Then the sampling set S is the sum of the offset O and the original sampling grid G. Normalization layers ensure that the values of a specific out feature are typically within the range of $[-1,1]$. There could be a large overlap in the walking range of local s^2 sample spots. The overlap would significantly impact the prediction close to boundaries, and these errors would progressively spread and result in output artifacts. Multiplying the offset by 0.25 meets the theoretical marginal requirements between overlap and non-overlap. The dynamic scope is defined as a value in the range [0,0.5] centered at 0.25, using the sigmoid functions and a 0.5 static factor, as follow:

$$O = 0.5 sigmoid(linear1\,(X) \cdot linear2(X)) \tag{1}$$

Finally, through reshaping operations, X' is produced using a grid sample and the sampling set S [17], as shown in Figure 5.

C. FocalModulation

This section outlines the concept of focal modulation, which focuses on enhancing the model's attention mechanism through precise adjustments in focal areas. Focal modulation proves crucial in scenarios where intensified attention to specific regions is necessary. It dynamically adjusts focus during data aggregation, enabling the model to selectively emphasize important details, thus maintaining acuity for local nuances while also enhancing the recognition of overarching structures. Focal modulation offers a refined approach to merging contextual data while preserving detail sensitivity.

The foundation of FocalNets (Focal Modulation Network [18]) is the replacement of conventional self-attention modules with a focal modulation mechanism. This approach aims to capture extensive contextual relationships and details within images more effectively and could focus the model more effectively on important targets, such as detected objects.

Traditional self-attention involves complex interactions and aggregations for each query token, which are computationally intensive. Focal modulation simplifies this by first

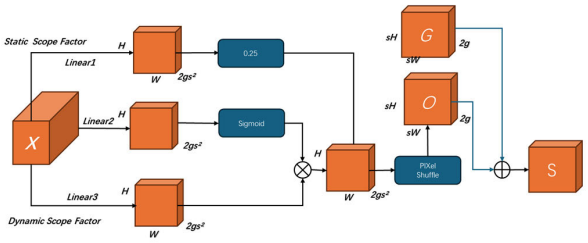

FIGURE 5. Sampling point generator in Dysample.

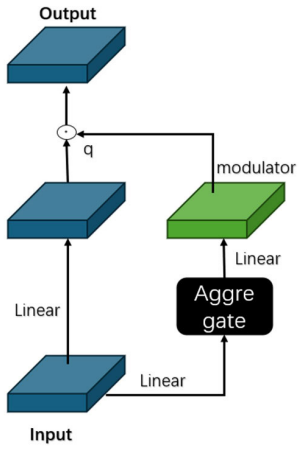

FIGURE 6. The FocalModulation structure.

compiling spatial context at varying levels into modulators [25], like Figure 6, where q is a query projection function.

The Focal Modulation model is implemented through the following steps:

1. Contextual Focus: Using sequential depth convolutional layers to encode visual context at multiple levels.
2. Gate Aggregation: Applying a gating mechanism to selectively integrate contextual information into modulators for each query token.
3. Affine Transformation per Element: Incorporating the gathered modulators into each query token via element-wise affine transformations.

Focal Contextualization is a crucial feature of FocalNets. It employs depth-wise convolutional layers to gather visual context from various distances, thus allowing the network to process image content at different levels of detail. This setup helps preserve local detail sensitivity while broadening the comprehension of larger structures during contextual aggregation.

A gating mechanism, frequently used in deep learning to manage data flow, determines the relevancy of information to be passed or blocked. In LSTM [26] and GRU architectures [27], gates control the timing of information flow in time-series data. For FocalNets, gate aggregation meticulously gathers relevant contextual data for each query token, ensuring focus on pertinent details.

Through gate aggregation, FocalNets prioritizes essential data for specific tasks, boosting efficiency and effectiveness by reducing the focus on non-essential information and enhancing attention to key attributes. In visual applications, this can lead to better object detection and image classification outcomes, especially in complex visual settings.

Finally, element-wise affine transformations apply modulators to each element. These transformations, basic linear operations, modify each element.

By applying these affine transformations, the model finally tunes each query token's features based on the contextual information, enabling precise adjustments that improve adaptation to complex visual environments and enhance the network's detail capture capability, thus improving overall performance in visual tasks such as object detection and image classification.

FIGURE 7. The origin dateset.

IV. EXPERIMENT SET

A. DATASET

There are limited publicly accessible datasets for object detection under various degradation conditions. To address this gap, we collected and compiled a custom dataset consisting of 7,000 real-world images with varying degrees of underexposure. Of these, 6,000 images were used to train the model, and 1,000 were used to validate its performance. The dataset contains 12 categories: bicycle, boat, bottle, bus, car, cat, chair, cup, dog, motorcycle, person, and table. These categories encompass the majority of elements needed for urban management detection, including common vehicles, animals, people, and other small objects. Examples from this dataset are shown in Figure 7, illustrating images affected not only by different levels of underexposure but also by various weather conditions, such as rain and snow.

To evaluate the model's robustness under noise and low exposure conditions and domain adaptation [28], a comprehensive assessment was conducted to determine its performance in real-world scenarios. This holistic evaluation provides insight into the model's capabilities under different conditions and can guide further improvements or adjustments needed for specific applications.

To further validate the model's noise resistance and to simulate the impact of temperature on image capture, making the experiment more reflective of real-world conditions, we introduced Gaussian noise of varying degrees into the

TABLE 1. Experiment setup argument.

Parameters	value
Epoch	130
Batch	8
Learning rate	0.01
Optimizer	SGD
Image size	640*640

dataset. Random noise was added to the images, with the relative intensity multiplied by a constant of 255 used as the standard deviation. The intensity of the noise added to each image varied (ranging from 0.15 to 0.65). The effect of this noise addition is shown in Figures 8 and 9.

FIGURE 8. The left image represents the original data, while the right image depicts the dataset (validation set) with added Gaussian noise.

FIGURE 9. Relatively high noise level.

B. EXPERIMENT SETUP

The experimental setup includes a Windows operating system, an NVIDIA GeForce RTX 4060Ti graphics card, PyTorch 2.01, CUDA 10.8, and Python 3.9. Additionally, the detailed configuration of the model training parameters is presented in Table 1.

C. EXPERIMENTAL EVALUATION CRITERION

The experimental evaluation criteria of this work primarily consist of Average Precision (AP), mean Average Precision (mAP), Precision (P), Recall (R), F1 (the harmonic mean of P and R), Parameters, GFLOPs, and model size. Their formulas are shown in blow. In addition, the mAP@0.5 represents the mAP value that is obtained by setting the intersection over union (IOU) to 50%.

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = 2\frac{P * R}{P + P} \tag{4}$$

$$AP = \int_0^1 P\,(Rd)\,r \tag{5}$$

$$mAP = \frac{\sum_{i=0}^{n} AP(i)}{n} \tag{6}$$

Using the metrics discussed, we can effectively evaluate the comprehensive performance of the ADNet-Dynamichead-YOLOv10 model. These metrics typically include accuracy, precision, recall, F1-score, and possibly specific performance indicators like mAP (mean Average Precision) for object detection tasks.

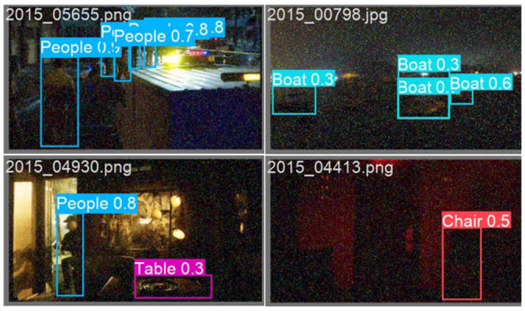

FIGURE 10. YOLOV8 validation set results.

D. EXPERIMENTAL PROCEDURES

To systematically evaluate the enhancements provided by various modules within the YOLO architecture, a comprehensive experimental plan can be established. The plan should start by testing the original YOLOv8 model on a specified dataset to establish baseline metrics such as accuracy, precision, recall, and mean Average Precision (mAP), which are crucial for subsequent comparisons.

Next, integrate the CPA-enhancer module with YOLOv8 and verify this enhanced setup. Monitor changes in performance metrics to assess the impact of the CPA-enhancer's denoising functionality on the overall effectiveness of the model. Then, incorporate the DySample module into the YOLOv8 now equipped with the CPA-enhancer. This step aims to evaluate improvements in sampling efficiency, which could enhance model precision across different object scales. Finally, replace the SPPF module with the FocalModulation module to assess the enhanced model's capability.

Conduct ablation studies by sequentially removing the CPA-enhancer from the CDF-YOLOv8 model, reverting to the original upsampling module in CPv-Dysample-FM-YOLOv8, and finally returning to the original SPPF structure in CPv-Dysample-FM-YOLOv8. Compare the data from these modifications to understand the individual and combined contributions of each module to the detection system. Additionally, to validate the effectiveness and seek optimal solutions for urban surveillance, experiments with the latest YOLOv10 should also be performed.

This structured approach facilitates a detailed comparison of each module's individual contributions and interactions within the detection system, thereby assessing the performance of the effective CDF-YOLOv8 model and validating the solution's efficacy.

TABLE 2. The result of different methods.

Model	CPA-enhancer	dysample	FocalModulation	P	R	F1	MAP50
Yolov8	×	×	×	0.67	0.49	0.5601	0.56
Yolov8	√	×	×	0.69	0.54	0.6098	0.61
Yolov8	×	√	×	0.67	0.49	0.5661	0.57
Yolov8	×	×	√	0.69	0.49	0.5714	0.57
Yolov8	×	√	√	0.62	0.52	0.5667	0.57
Yolov8	√	×	√	0.70	0.53	0.6056	0.61
Yolov8	√	√	×	0.66	0.54	0.5946	0.60
Yolov8	√	√	√	0.70	0.53	0.6056	0.62
Yolov10	×	×	×	0.62	0.48	0.5404	0.54
Yolov10	√	√	√	0.67	0.50	0.5738	0.58

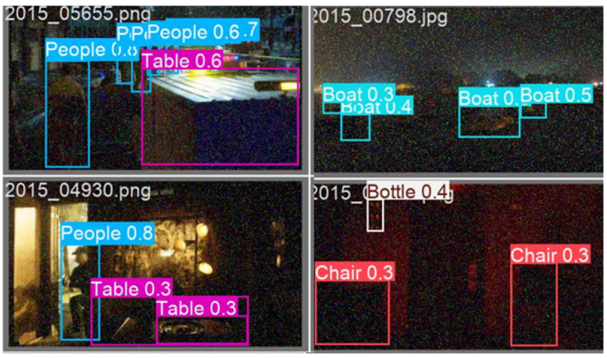

FIGURE 11. CDF-YOLOV8 validation set results.

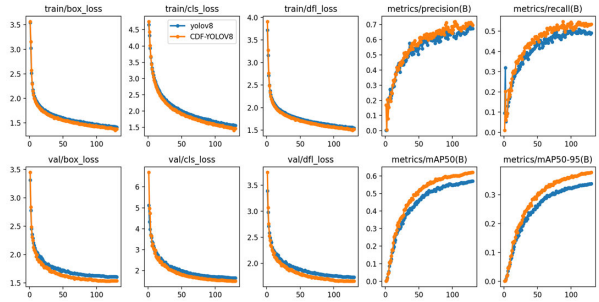

FIGURE 12. YOLOV8 training process.

V. RESULT

The results of our experiments are shown in Figures 10 (original YOLOv8) and 11 (CDF-YOLOv8).

The experimental outcomes demonstrate that our CDF-YOLOv8 model enhances the YOLOv8 framework, effectively addressing image degradation caused by underexposure, added Gaussian noise, and other unknown factors in our dataset. The lightweight DySample improved the upsampling process, while the FocalModulation module enhanced the feature extraction capability. The integration of these techniques significantly improved the robustness of the YOLOv8 algorithm in the robustness criterion [29], particularly in handling noisy datasets, making it an ideal choice for deployment in smart city intelligent detection systems.

The CPA-enhancer module is the core of the CDF-YOLOv8 structure and represents the most substantial improvement to the YOLOv8 algorithm by enhancing the feature extraction from the input images. This effectively increases the model's robustness, as evidenced by the training process comparison shown in Figure 12.

The experimental results also indicate that YOLOv8 outperforms YOLOv10 on our dataset, exhibiting higher performance, which validates our choice of YOLOv8. Our CDF-YOLOv8 model demonstrated significant effectiveness in detecting image degradation caused by adverse lighting conditions, weather effects, and camera hardware failures. This makes the CDF-YOLOV8 model particularly well-suited for the variable and challenging conditions of urban environments, enhancing its potential as an intelligent detection model for urban management. The complete experimental data is presented in Table 2.

VI. CONCLUSION

For target detection and image recognition in smart cities, we have developed the CDF-YOLOv8 model by integrating CPA-enhancer, DySample, and FocalModulation architectures into the base YOLOv8 algorithm.

Due to the required good robustness in urban detection tasks [30]. Compared to the original YOLOv8, the CDF-YOLOv8 exhibits enhanced robustness and is well-suited for urban target detection and object recognition, particularly in environments affected by weather and lighting changes.

Through comparative studies using YOLOv8 and YOLOv10, it has been demonstrated that YOLOv8 is a suitable choice for smart city target detection tasks. Furthermore, the integration of CPA-enhancer, DySample, and FocalModulation modules also effectively enhances performance in the YOLOv10 framework, showing potential for application in other modules. However, overall performance is superior in CDF-YOLOv8.

Our research introduces CDF-YOLOv8 as a viable and effective algorithm for smart city target detection tasks, addressing the issue of decreased robustness due to unknown image degradation. This provides a reference for other target detection algorithms to manage unknown image damages, contributing to the field of smart urban management.

REFERENCES

[1] A. K. Jha, A. Ghimire, S. Thapa, A. M. Jha, and R. Raj, "A review of AI for urban planning: Towards building sustainable smart cities," in *Proc. 6th Int. Conf. Inventive Comput. Technol. (ICICT)*, Jan. 2021, pp. 937–944, doi: 10.1109/ICICT50816.2021.9358548.

[2] D. Kothadiya, A. Chaudhari, R. Macwan, K. Patel, and C. Bhatt, "The convergence of deep learning and computer vision: Smart city applications and research challenges," in *Proc. Atlantis Highlights Comput. Sci.*, 2021, pp. 14–22.
[3] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, "YOLOv10: Real-time end-to-end object detection," 2024, *arXiv:2405.14458*.
[4] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
[6] G. Jocher, A. Chaurasia, and J. Qiu. (2023). *Ultralytics YOLO (Version 8.0.0) [Computer Software]*. [Online]. Available: https://github.com/ultralytics/ultralytics8
[7] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," 2024, *arXiv:2402.13616*.
[8] Y. Li, Q. Fan, H. Huang, Z. Han, and Q. Gu, "A modified YOLOv8 detection network for UAV aerial image recognition," *Drones*, vol. 7, no. 5, p. 304, May 2023.
[9] T. Wu and Y. Dong, "YOLO-SE: Improved YOLOv8 for remote sensing object detection and recognition," *Appl. Sci.*, vol. 13, no. 24, p. 12977, Dec. 2023.
[10] V. Manhas and Poonam, "Enhancing smart city surveillance: Vehicle number plate detection with YOLO NAS," in *Proc. MIT Art, Design Technol. School Comput. Int. Conf.*, Apr. 2024, pp. 1–6.
[11] N. Zarei, P. Moallem, and M. Shams, "Fast-YOLO-rec: Incorporating YOLO-base detection and recurrent-base prediction networks for fast vehicle detection in consecutive images," *IEEE Access*, vol. 10, pp. 120592–120605, 2022, doi: 10.1109/ACCESS.2022.3221942.
[12] W. Wu and J. Lai, "Multi camera localization handover based on YOLO object detection algorithm in complex environments," *IEEE Access*, vol. 12, pp. 15236–15250, 2024, doi: 10.1109/ACCESS.2024.3357519.
[13] Y. Liu, H. Zhang, and D. Gao, "DiffYOLO: Object detection for anti-noise via YOLO and diffusion models," 2401, *arXiv:2401.01659*.
[14] N. Madhasu and S. D. Pande, "Chrometect GAYO: Classification and colorization using PIX2PIX and YOLOV8," in *Proc. 7th Int. Conf. Comput. Syst. Inf. Technol. Sustain. Solutions (CSITSS)*, Nov. 2023, pp. 1–6.
[15] Y. Zhou, "A YOLO-NL object detector for real-time detection," *Expert Syst. Appl.*, vol. 238, Mar. 2024, Art. no. 122256.
[16] Y. Zhang, Y. Wu, Y. Liu, and X. Peng, "CPA-enhancer: Chain-of-thought prompted adaptive enhancer for object detection under unknown degradations," 2024, *arXiv:2403.11220*.
[17] W. Liu, H. Lu, H. Fu, and Z. Cao, "Learning to upsample by learning to sample," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 6027–6037.
[18] J. Yang, C. Li, X. Dai, and J. Gao, "Focal modulation networks," in *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, vol. 35, 2022, pp. 4203–4217.
[19] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," 2019, *arXiv:1911.08287*.
[20] X. Li, "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," in *Proc. Annu. Conf. Neural Inf. Process. Syst. (NIPS)*, Dec. 2020, pp. 21002–21012.
[21] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[22] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768.
[23] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, and E. Chi, "Chain-of-thought prompting elicits reasoning in large language models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 24824–24837.
[24] X. Zhang, C. Liu, D. Yang, T. Song, Y. Ye, K. Li, and Y. Song, "RFAConv: Innovating spatial attention and standard convolutional operation," 2023, *arXiv:2304.03198*.
[25] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," 2018, *arXiv:1803.02155*.
[26] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019.
[27] S. Nosouhian, F. Nosouhian, and A. K. Khoshouei, "A review of recurrent neural network architecture for sequence learning: Comparison between LSTM and GRU," Tech. Rep., 2021.
[28] A. Banitalebi-Dehkordi, A. Amirkhani, and A. Mohammadinasab, "EBCDet: Energy-based curriculum for robust domain adaptive object detection," *IEEE Access*, vol. 11, pp. 77810–77825, 2023.
[29] A. Amirkhani, A. Khosravian, M. Masih-Tehrani, and H. Kashiani, "Robust semantic segmentation with multi-teacher knowledge distillation," *IEEE Access*, vol. 9, pp. 119049–119066, 2021.
[30] A. Amirkhani, M. P. Karimi, and A. Banitalebi-Dehkordi, "A survey on adversarial attacks and defenses for object detection and their applications in autonomous vehicles," *Vis. Comput.*, vol. 39, no. 11, pp. 5293–5307, Nov. 2023.

P. LU received the bachelor's degree from Tianjin University of Technology. He is currently pursuing the degree with Hiroshima University, with a focus on machine learning and image processing.

Y. S. JIA received the bachelor's degree from Tianjin University of Technology, where he is currently pursuing the degree, with research interests include deep learning and signal processing.

W. X. ZENG is currently pursuing the bachelor's degree with Tianjin University of Technology, with research interest includes machine learning.

P. WEI is a Lecturer at the Maritime College, Tianjin University of Technology, a Senior Chief Engineer, and a Researcher with China Maritime Think Tank. His research interests include maritime navigation, automated intelligent engineering, and green environmental protection in shipping.