Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.DOI

YOLOv8-AM: YOLOv8 Based on Effective Attention Mechanisms for Pediatric Wrist Fracture Detection

CHUN-TSE CHIEN[1], RUI-YANG JU[2], (Student Member, IEEE), KUANG-YI CHOU[3], ENKAER XIEERKE[4], JEN-SHIUN CHIANG[1], (Member, IEEE)

[1]Department of Electrical and Computer Engineering, Tamkang University, New Taipei City, 251301, Taiwan
[2]Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei City 106335, Taiwan
[3]School of Nursing, National Taipei University of Nursing and Health Sciences, Taipei City, 112303, Taiwan
[4]College of Energy and Mechanical Engineering, Shanghai University of Electric Power, Shanghai, 201306, China

Corresponding author: Jen-Shiun Chiang (e-mail: jsken.chiang@gmail.com).

This paper is an expanded paper from International Conference on Neural Information Processing (ICONIP) held on December 2-6, 2024 in Auckland, New Zealand. This work is supported by National Science and Technology Council of Taiwan, under Grant Number: NSTC 112-2221-E-032-037-MY2.

ABSTRACT Wrist trauma and even fractures occur frequently in daily life, particularly among children who account for a significant proportion of fracture cases. Before performing surgery, surgeons often request patients to undergo X-ray imaging first and prepare for it based on the analysis of the radiologist. With the development of neural networks, You Only Look Once (YOLO) series models have been widely used in fracture detection as computer-assisted diagnosis (CAD). In 2023, Ultralytics presented the latest version of the YOLO models, which has been employed for detecting fractures across various parts of the body. Attention mechanism is one of the hottest methods to improve the model performance. This research work proposes YOLOv8-AM, which incorporates the attention mechanism into the original YOLOv8 architecture. Specifically, we respectively employ four attention modules, Convolutional Block Attention Module (CBAM), Global Attention Mechanism (GAM), Efficient Channel Attention (ECA), and Shuffle Attention (SA), to design the improved models and train them on GRAZPEDWRI-DX dataset. Experimental results demonstrate that the mean Average Precision at IoU 50 (mAP 50) of the YOLOv8-AM model based on ResBlock + CBAM (ResCBAM) increased from 63.6% to 65.8%, which achieves the state-of-the-art (SOTA) performance. Conversely, YOLOv8-AM model incorporating GAM obtains the mAP 50 value of 64.2%, which is not a satisfactory enhancement. Therefore, we combine ResBlock and GAM, introducing ResGAM to design another new YOLOv8-AM model, whose mAP 50 value is increased to 65.0%. The implementation code for this study is available on GitHub at https://github.com/RuiyangJu/Fracture_Detection_Improved_YOLOv8.

INDEX TERMS Computer Vision, Deep Learning, Fracture Detection, Medical Image Diagnostics, Medical Image Processing, Object Detection, Radiology, X-ray Imaging, You Only Look Once (YOLO).

I. INTRODUCTION

Wrist fractures are one of the most common fractures, particularly among the elderly and children [1], [2]. Fractures typically occur in the distal 2 cm of the radius near the joint. Failure to provide timely treatment may result in deformities of the wrist joint, restricted joint motion, and joint pain for the patients [3]. In children, a misdiagnosis would lead to a lifelong inconvenience [4].

In cases of pediatric wrist fractures, surgeons inquire about the circumstances leading to the fracture and conduct a preoperative examination. Presently, fracture examinations mainly utilize three types of medical imaging equipment: Magnetic Resonance Imaging (MRI), Computed Tomography (CT), and X-ray. Among these, X-ray is the preferred choice for most patients due to its cost-effectiveness [5]. In hospitals providing advanced medical care, radiologists are required to follow the Health Level 7 (HL7) and Digital Imaging and Communications in Medicine (DICOM) international standards for the archival and transfer of X-ray images [6]. Nevertheless, the scarcity of radiologists in

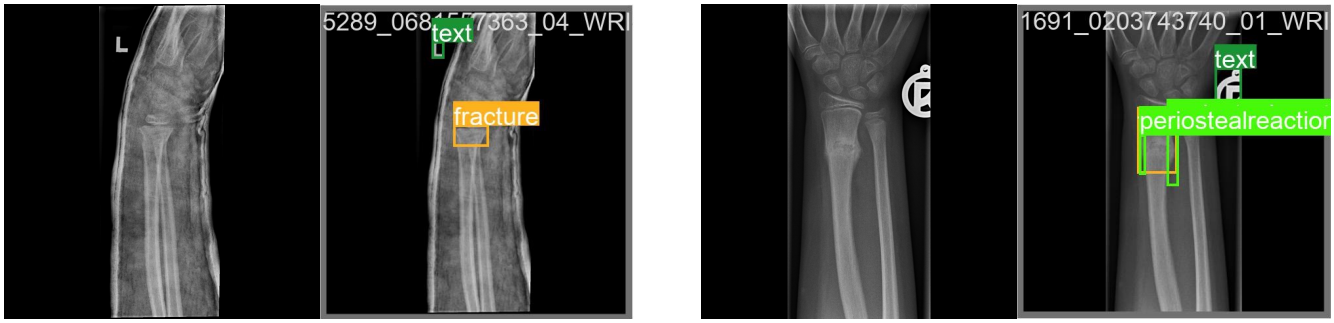

FIGURE 1: Example diagrams illustrating the pediatric wrist fracture prediction task conducted in this work, with the input image displayed on the left of each group and the prediction result shown on the right of each group.

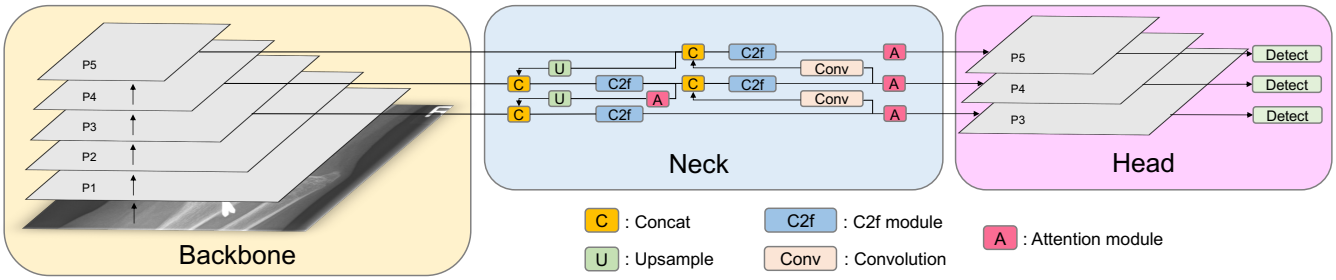

FIGURE 2: The architecture of the YOLOv8-AM model for pediatric wrist fracture detection, where the attention module is the part newly employed to the original YOLOv8 model architecture.

underdeveloped regions poses a significant challenge to the prompt delivery of patient care [7]–[9]. The studies [10], [11] indicate a concerning 26% error rate in medical imaging analysis during emergency cases.

Computer-assisted diagnosis (CAD) of medical images provides experts (i.e., radiologists, surgeons, etc.) with help in some decision tasks. With the continuous development of deep learning and the improvement of medical image processing techniques [12]–[15], more and more researchers are trying to employ neural networks for CAD, including fracture detection [16]–[20].

You Only Look Once (YOLO) model [21], as one of the most important models for object detection tasks [22], demonstrates satisfactory performance in fracture detection [23]–[25]. With the introduction of YOLOv8 [26], the latest version of the YOLO models, by Ultralytics in 2023, it has been widely used in various object detection tasks. As shown in FIGURE 1, GRAZPEDWRI-DX [27] is a public dataset of 20,327 pediatric wrist trauma X-ray images. This work follows the study [28] that evaluates the performance of different network models on this dataset in a fracture detection task.

Due to the capacity of attention mechanism to accurately focus on all pertinent information of the input, they are widely applied to various neural network architectures. Presently, two primary attention mechanisms exist: spatial attention and channel attention, designed to capture pixel-level pairwise relationships and channel dependencies, respectively [29]–[33]. Studies [34]–[37] have demonstrated that incorporating attention mechanism into convolutional blocks shows great potential for performance improvement.

Therefore, we propose the YOLOv8-AM model for fracture detection by employing four different attention modules, including Convolutional Block Attention Module (CBAM) [38], Global Attention Mechanism (GAM) [39], Efficient Channel Attention (ECA) [40] and Shuffle Attention (SA) [41], to the YOLOv8 architecture, as shown in FIGURE 2.

Experimental results [38] show that the model performance of combining ResBlock [42] and CBAM is better than that of CBAM, and therefore we incorporate ResBlock + CBAM (ResCBAM) to the YOLOv8 architecture for experiments. After comparing the effects of different attention modules on the YOLOv8-AM model performance, we find that the GAM module has poorer gain effect on the model performance, so we propose ResGAM, which combines ResBlock and GAM, and incorporates this attention module into the YOLOv8 architecture.

The main contributions of this paper are as follows:

- This work employs four different attention modules to the YOLOv8 architecture and proposes the YOLOv8-AM model for fracture detection, where the YOLOv8-AM model based on ResBlock + CBAM (ResCBAM) achieves the state-of-the-art (SOTA) performance.
- Since the performance of the YOLOv8-AM model

based on GAM is unsatisfactory, we propose ResBlock + GAM (ResGAM) and design the new YOLOv8-AM model based on it for fracture detection.
- This work demonstrates that compared to the YOLOv8 model, the performances of the YOLOv8-AM models with the incorporation of different attention modules are all greatly improved on the GRAZPEDWRI-DX dataset.

This paper is organized as follows: Section II introduces the research on fracture detection utilizing deep learning methods and outlines the evolution of attention mechanism. Section III presents the overall architecture of the YOLOv8-AM model and four different attention modules employed. Section IV conducts a comparative analysis of the performance of four YOLOv8-AM models against the baseline YOLOv8 model. Subsequently, Section V discusses the reasons why GAM provides less improvement in the performance of the YOLOv8-AM model for fracture detection. Finally, Section VI concludes this research work and explores the future works.

II. RELATED WORK

A. FRACTURE DETECTION

Fracture detection is a hot topic in medical image processing (MIP). Researchers usually employ various neural networks for prediction, including the YOLO series models [21], [26], [43]. Burkow et al. [44] utilized the YOLOv5 model [43] to recognize rib fracture on 704 pediatric Chest X-ray (CXR) images. Tsai et al. [45] performed data augmentation on CXR images and subsequently employed the YOLOv5 model to detect fractures. Warin et al. [46] firstly categorized the maxillofacial fractures into four categories (i.e., frontal, midfacial, mandibular, and no fracture), and predicted them using the YOLOv5 model on 3,407 CT images. In addition, Warin et al. [47] used the YOLOv5 model to detect fractures in the X-ray images of the mandible. Yuan et al. [48] incorporated external attention and 3D feature fusion methods into the YOLOv5 model for fracture detection in skull CT images. Furthermore, vertebral localization proves valuable for recognizing vertebral deformities and fractures. Mushtaq et al. [49] utilized YOLOv5 for lumbar vertebrae localization, achieving the mean Average Precision (mAP) value of 0.957. Although the YOLOv5 model is extensively employed in fracture detection, the utilization of the YOLOv8 model [26] is comparatively rare.

B. ATTENTION MODULE

SENet [50] initially proposed a mechanism to learn channel attention efficiently by applying Global Average Pooling (GAP) to each channel independently. Subsequently, channel weights were generated using the Fully Connected layer and the Sigmoid function, leading to the good model performance. Following the introduction of feature aggregation and feature recalibration in SENet, some studies [51], [52] attempted to improve the SE block by capturing more sophisticated channel-wise dependencies. Woo et al. [38] combined the channel attention module with the spatial attention module, introducing the CBAM to improve the representation capabilities of Convolutional Neural Networks (CNNs). To reduce information loss and enhance global dimension-interactive features, Liu et al. [39] introduced modifications to CBAM and presented GAM. This mechanism reconfigured submodules to magnify salient cross-dimension receptive regions. Although these methods [38], [39] have achieved better accuracy, they usually bring higher model complexity and suffer from heavier computational burden. Therefore, Wang et al. [40] proposed the ECA module, which captures local cross-channel interaction by considering every channel and its k neighbors, resulting in significant performance improvement at the cost of fewer parameters. Different from the ECA module, Zhang et al. [41] introduced the SA module. This module groups the channel dimensions into multiple sub-features and employs the Shuffle Unit to integrate the complementary sub-features and the spatial attention module for each sub-feature, achieving excellent performance with low model complexity. Each of these attention modules can be applied to different neural network architectures to improve model performance.

III. METHODOLOGY

A. BASELINE MODEL

YOLOv8 architecture comprises four key components: Backbone, Neck, Head, and Loss Function. The Backbone incorporates the Cross Stage Partial (CSP) [53] concept, offering the advantage of reducing computational loads while enhancing the learning capability of CNNs. Illustrated in FIGURE 3, YOLOv8 diverges from YOLOv5 employing the C3 module [43], adopting the C2f module, which integrates the C3 module and the Extended ELAN [54] (E-ELAN) concept from YOLOv7 [55]. Specifically, the C3 module involves three convolutional modules and multiple bottlenecks, whereas the C2f module consists of two convolutional modules concatenated with multiple bottlenecks. The convolutional module is structured as Convolution-Batch Normalization-SiLU (CBS).

In the Neck part, YOLOv5 employs the Feature Pyramid Network (FPN) [56] architecture for top-down sampling, ensuring that the lower feature map incorporates richer feature information. Simultaneously, the Path Aggregation Network (PAN) [57] structure is applied for bottom-up sampling, enhancing the top feature map with more precise location information. The combination of these two structures is executed to guarantee the accurate prediction of images across varying dimensions. YOLOv8 follows the FPN and PAN frameworks while deleting the convolution operation during the up-sampling stage, as illustrated in FIGURE 3.

In contrast to YOLOv5, which employs a coupled head, YOLOv8 adopts a decoupled head to separate the classification and detection heads. Specifically, YOLOv8 eliminates the objectness branch, only retaining the classification and regression branches. Additionally, it departs from anchor-based [58] method in favor of anchor-free [59] approach,

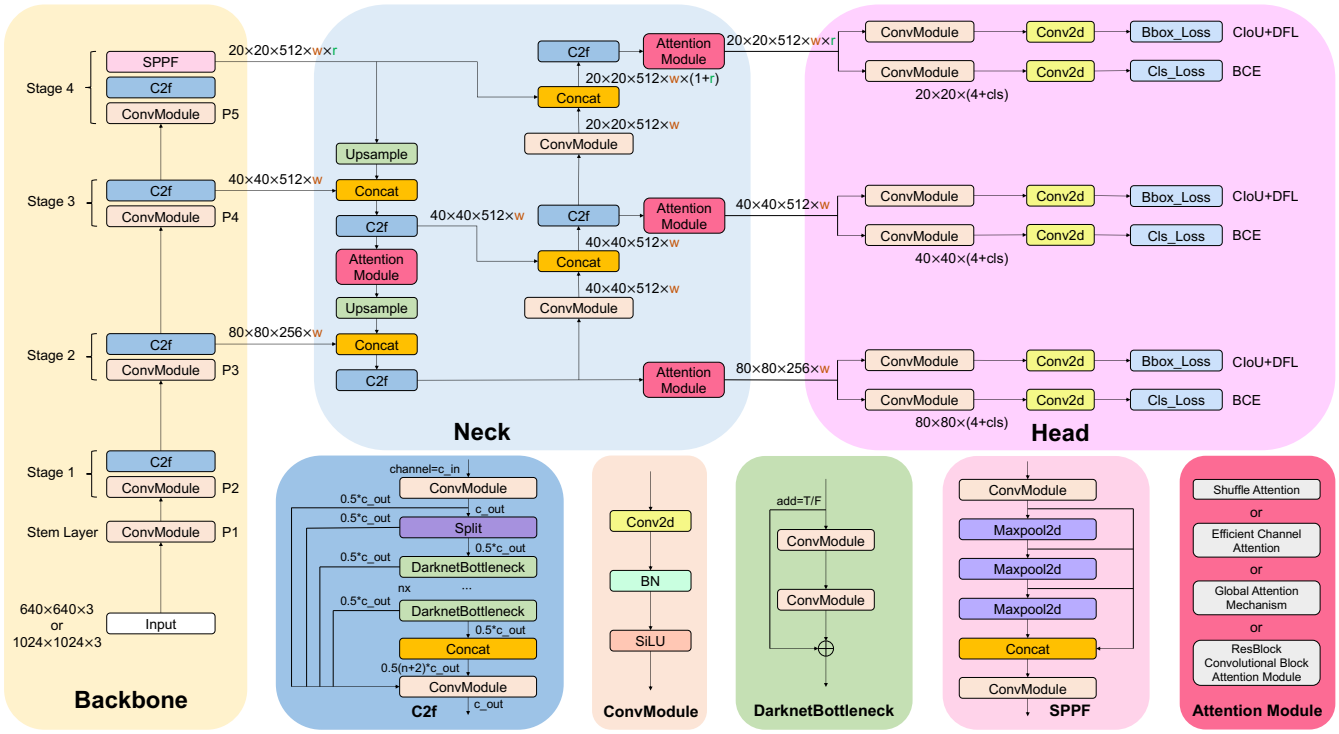

FIGURE 3: Detailed illustration of the YOLOv8-AM model architecture, where the attention modules are Shuffle Attention (SA), Efficient Channel Attention (ECA), Global Attention Mechanism (GAM), and ResBlock + Convolutional Block Attention Module (ResCBAM), respectively.

where the location of the target is determined by its center, and the prediction involves estimating the distance from the center to the boundary.

In YOLOv8, the loss function employed for the classification branch involves the utilization of the Binary Cross-Entropy (BCE) Loss, as expressed by the equation as follows:

$$Loss_{BCE} = -w[y_n \log x_n + (1 - y_n) \log (1 - x_n)], \quad (1)$$

where w denotes the weight; y_n represents the labeled value, and x_n signifies the predicted value generated by the model.

For the regression branch, YOLOv8 incorporated the use of Distribute Focal Loss (DFL) [60] and Complete Intersection over Union (CIoU) Loss [61]. The DFL function is designed to emphasize the expansion of probability values around object y. Its equation is presented as follows:

$$\begin{aligned} Loss_{DF} = -[(y_{n+1} - y) \log \frac{y_{n+1} - y_n}{y_{n+1} - y_n} \\ + (y - y_n) \log \frac{y - y_n}{y_{n+1} - y_n}]. \end{aligned} \quad (2)$$

The CIoU Loss introduces an influence factor to the Distance Intersection over Union (DIoU) Loss [62] by considering the aspect ratio of the predicted bounding box and the Ground-Truth bounding box. The corresponding equation is as follows:

$$Loss_{CIoU} = 1 - IoU + \frac{d^2}{c^2} + \frac{v^2}{(1 - IoU) + v}, \quad (3)$$

where IoU measures the overlap between the predicted and Ground-Truth bounding boxes; d is the Euclidean distance between the center points of the predicted and Ground-Truth bounding boxes, and c is the diagonal length of the smallest enclosing box that contains both predicted and Ground-Truth bounding boxes. Additionally, v represents the parameter quantifying the consistency of the aspect ratio, defined by the following equation:

$$v = \frac{4}{\pi^2}(\arctan \frac{w_{gt}}{h_{gt}} - \arctan \frac{w_p}{h_p})^2, \quad (4)$$

where w denotes the weight of the bounding box; h represents the height of the bounding box; gt means the Ground-Truth, and p means the prediction.

B. PROPOSED METHOD

In recent years, the attention mechanism has obtained excellent results in the field of object detection [63]–[65]. With the integration of the attention mechanism, the models can recognize the most important information of the input image for extraction and suppress the useless information.

This work incorporates the attention module into the Neck part of YOLOv8 to enhance the capture of key features and suppress the interfering information. As illustrated in FIGURE 3, attention modules, namely CBAM [38], GAM [39], ECA [40], and SA [41], are independently employed

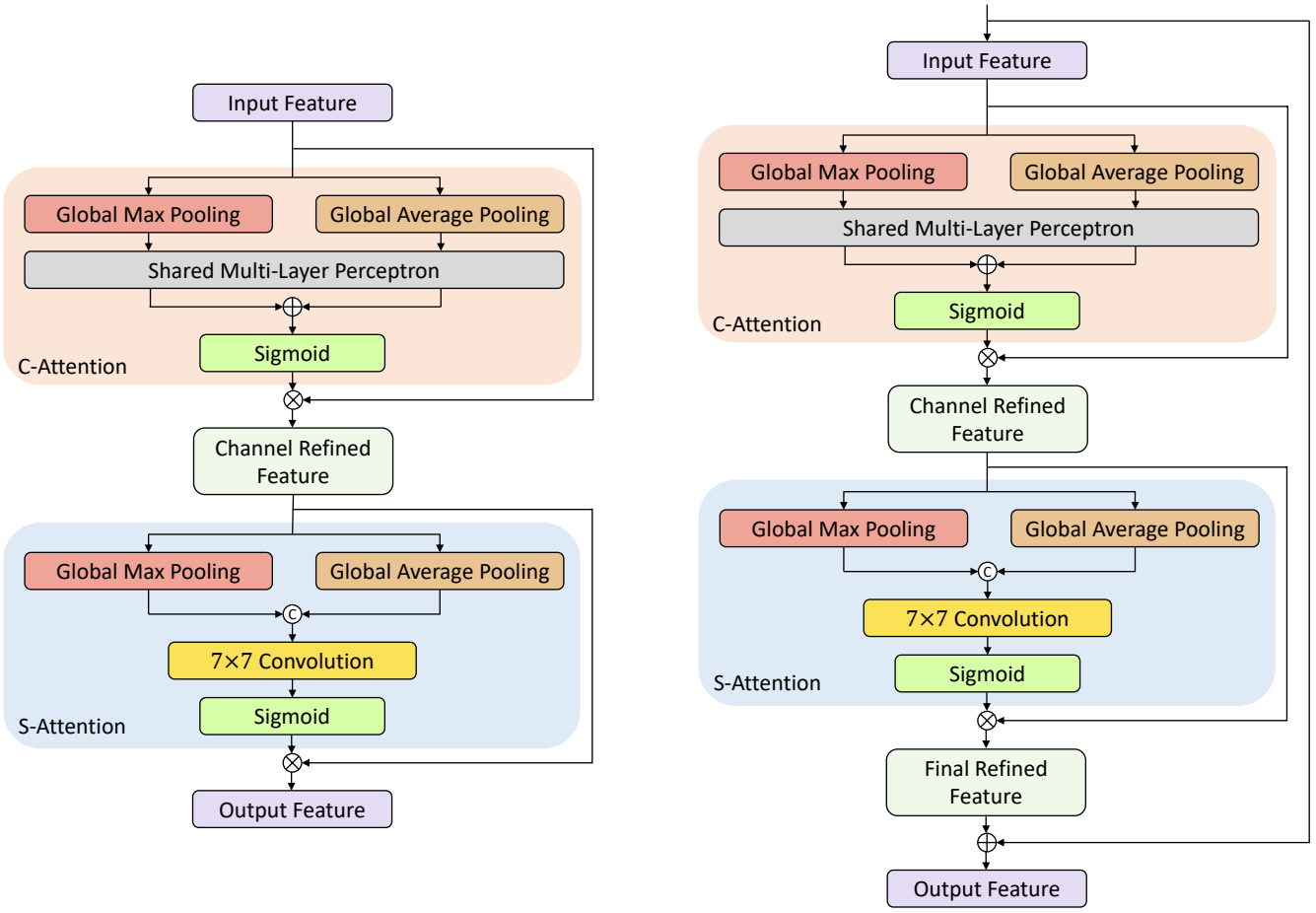

FIGURE 4: Detailed illustration of Convolutional Block Attention Module (CBAM), the left is the architecture of CBAM, and the right is the architecture of ResCBAM.

after each of the four C2f modules. A detailed introduction of these four attention modules is presented in Section III-C.

C. ATTENTION MODULES

1) Convolutional Block Attention Module (CBAM)

CBAM comprises both channel attention (C-Attention) and spatial attention (S-Attention), as shown on the left of FIGURE 4. Given an intermediate feature map denoted as $F_{input} \in \mathbb{R}_{C\times H\times W}$, CBAM sequentially infers a 1D channel attention map $M_C \in \mathbb{R}^{C\times 1\times 1}$ and a 2D spatial attention map $M_S \in \mathbb{R}^{1\times H\times W}$ through the following equation:

$$F_{CR} = M_C(F_{input}) \otimes F_{input}, \tag{5}$$

$$F_{FR} = M_S(F_{CR}) \otimes F_{CR}, \tag{6}$$

where $\otimes$ is the element-wise multiplication; F_{CR} is the Channel Refined Feature, and F_{FR} is the Final Refined Feature. For CBAM, F_{output} is F_{FR} as shown in the following equation:

$$F_{output} = F_{FR}. \tag{7}$$

It can be seen from the right of FIGURE 4, for ResBlock + CBAM (ResCBAM), F_{output} is the element-wise summation of F_{input} and F_{FR} as shown in the following equation:

$$F_{output} = F_{input} + F_{FR}. \tag{8}$$

Based on the previous studies [66], [67], CBAM employs both Global Average Pooling (GAP) and Global Max Pooling (GMP) to aggregate the spatial information of a feature map, which generates two different spatial contextual descriptors. Subsequently, these two descriptors share the same Multi-Layer Perceptron (MLP) with one hidden layer. Finally, the output feature vectors from the element-wise summation are input to the sigmoid function (σ). The specific channel attention equation is as follows:

$$M_C(F) = \sigma[MLP(GAP(F)) + MLP(GMP(F))]. \tag{9}$$

For the spatial attention, CBAM performs GAP and GMP along the channel axis respectively, and then concatenates them together to effectively highlight the information regions

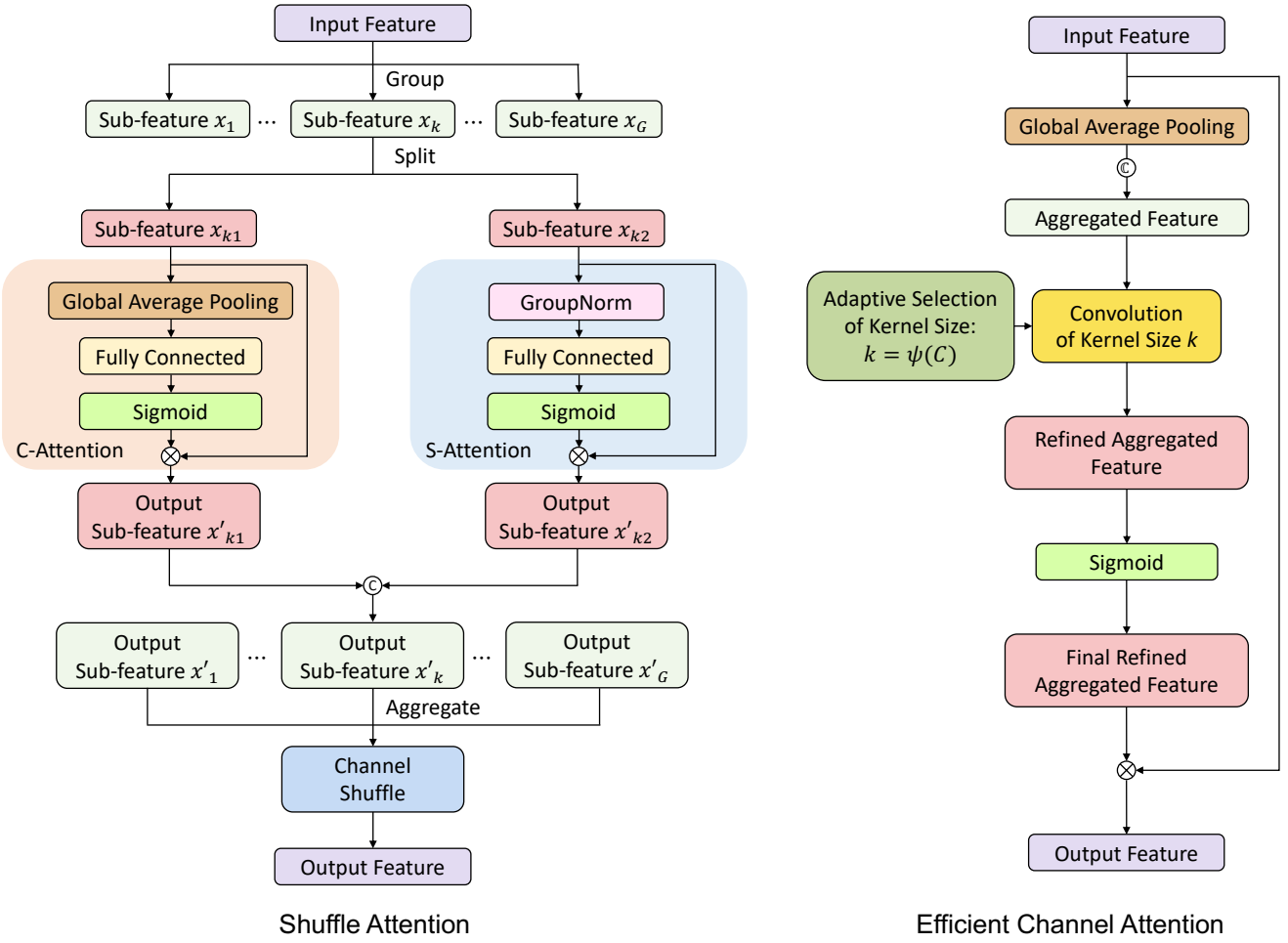

FIGURE 5: Detailed illustration of Shuffle Attention (SA) and Efficient Channel Attention (ECA), the left is the architecture of SA, and the right is the architecture of ECA.

[68], with the symbol © denoting the concatenation. Subsequently, a 7×7 convolutional layer is used to perform the convolution operation on these features. The output of this convolution is used as the input of the sigmoid function (σ). The spatial attention is computed using the following equation:

$$M_S(F) = \sigma[f^{7\times7}(GAP(F)©GMP(F))]. \quad (10)$$

2) Efficient Channel Attention (ECA)

ECA primarily encompasses cross-channel interaction and 1D convolution with an adaptive convolution kernel, as shown on the right of FIGURE 5. Cross-channel interaction represents a novel approach to combining features, enhancing the expression of features for specific semantics. The input feature map $F_{input} \in \mathbb{R}^{C\times H\times W}$ obtains the aggregated feature F_a after performing GAP and cross-channel interaction, where $\mathbb{C}$ represents cross-channel interaction. The equation is shown as follows:

$$F_a = \mathbb{C}(GAP(F_{input})). \quad (11)$$

For the aggregated features, ECA captures the local cross-channel interaction by considering the interaction between the features of each channel and their neighboring k channels, avoiding the use of 1D convolution for dimensionality reduction, and effectively realizing the multi-channel interaction, where the weights of features F_{ai} are as follows:

$$\omega_i = \sigma(\sum_{j=1}^{k} \mathbb{W}^j F_{ai}^j), F_{ai}^j \in \Omega_i^k, \quad (12)$$

where σ is the sigmoid function, and Ω_i^k denotes the set of k neighboring channels of F_{ai}. From Eq. 12 we can see that all the channels have the same inclination parameter, so the model will be more efficient.

For 1D convolution with adaptive convolution kernel, ECA introduces an adaptive method to determine the size of the value. Specifically, the size k of the convolution kernel is directly related to the channel dimension C, indicating a nonlinear mapping relationship between them, as illustrated by the following equation:

$$C = \phi(k) = 2^{\gamma*k-b}. \quad (13)$$

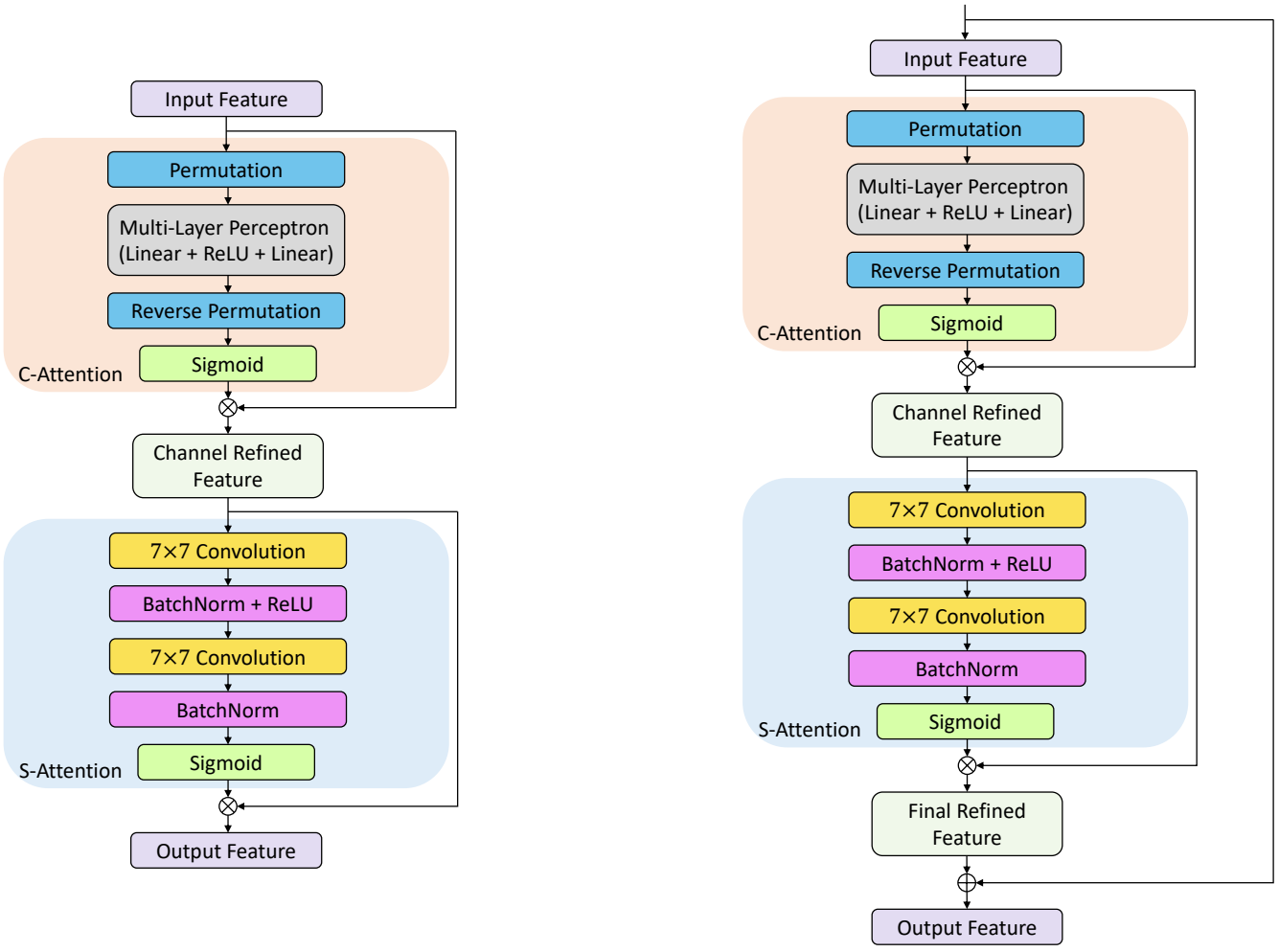

FIGURE 6: Detailed illustration of Global Attention Mechanism (GAM), the left is the architecture of GAM, and the right is the architecture of ResGAM.

Meanwhile, the convolution kernel size k can be adaptively determined as shown in the following equation:

$$k = \psi(C) = \left| \frac{\log_2 C}{\gamma} + \frac{b}{\gamma} \right|_{odd}, \tag{14}$$

where t is closest to $|t|_{odd}$, and based on the experimental results of ECA [40], the values of γ and b are set to 2 and 1, respectively.

3) Shuffle Attention (SA)

SA divides the input feature maps into different groups, employing the Shuffle Unit to integrate both channel attention and spatial attention into one block for each group, as shown on the left of FIGURE 5. Subsequently, the sub-features are aggregated, and the "Channel Shuffle" operator, as employed in ShuffleNetV2 [69], is applied to facilitate information communication among various sub-features.

For channel attention, SA employs GAP to capture and embed global information for the sub-feature x_{k1}. In addition, a simple gating mechanism with sigmoid functions is used to create a compact function that facilitates precise and adaptive selection. The final output of the channel attention can be obtained by the following equation:

$$x_{k1}{}' = \sigma[FC(GAP(x_{k1}))] \otimes x_{k1}. \tag{15}$$

For spatial attention, to complement the channel attention, SA first uses Group Normalization (GN) [70] for the sub-feature x_{k2} to obtain spatial-wise statistics. Subsequently, the representation of the output sub-feature $x_{k2}{}'$ is enhanced through the function of $FC()$ as shown in the following equation:

$$x_{k2}{}' = \sigma[FC(GN(x_{k2}))] \otimes x_{k2}. \tag{16}$$

It can be seen on the left of FIGURE 5, the output sub-feature $x_k{}'$ is obtained by concatenating $x_{k1}{}'$ and $x_{k2}{}'$, and

the equation is shown as follows:

$$x_k{}' = x_{k1}{}' \copyright x_{k2}{}'. \tag{17}$$

4) Global Attention Mechanism (GAM)

GAM adopts the main architecture proposed by CBAM consisting of channel at-tention and spatial attention, and redesigns the submodules as shown in FIGURE 6. Additionally, we add Shortcut Connection [42] between the layers within GAM, which allows the inputs to propagate forward faster, as shown in the following equation:

$$\begin{aligned} F_{output} = F_{input} + [M_S(M_C(F_{input}) \otimes F_{input}) \\ \otimes (M_C(F_{input}) \otimes F_{input})]. \end{aligned} \tag{18}$$

For channel attention, GAM employs a 3D permutation initially to retain three-dimensional information. Subsequently, it employs a two-layer MLP to amplify the channel-spatial dependencies across dimensions. In summary, the equation is presented as follows:

$$\begin{aligned} M_C(F) = \sigma[ReversePermutation(\\ MLP(Permutation(F)))]. \end{aligned} \tag{19}$$

For spatial attention, GAM uses two 7×7 convolution layers to integrate spatial information. It also adopts a reduction rate r consistent with the approach in BAM [67]. The corresponding equation is presented as follows:

$$M_S(F) = \sigma[BN(f^{7\times7}(BN + ReLU(f^{7\times7}(F))))]. \tag{20}$$

In contrast to CBAM, the authors of GAM considered that max pooling would reduce the amount of information and have a negative effect, so pooling was eliminated to further preserve the feature map.

IV. EXPERIMENT

A. DATASET

GRAZPEDWRI-DX [27] is a public dataset of 20,327 pediatric wrist trauma X-ray images released by the University of Medicine of Graz. These X-ray images were collected by multiple pediatric radiologists at the Department for Pediatric Surgery of the University Hospital Graz between 2008 and 2018, involving 6,091 patients and a total of 10,643 studies. This dataset is annotated with 74,459 image labels, featuring a total of 67,771 labeled objects.

B. PREPROCESSING AND DATA AUGMENTATION

In the absence of predefined training, validation, and test sets by the dataset publisher, we perform a random division, allocating 70% to the training set, 20% to the validation set, and 10% to the test set. Specifically, the training set comprises 14,204 images (69.88%), the validation set includes 4,094 images (20.14%), and the test set comprises 2,029 images (9.98%).

Due to the limited diversity in brightness among the X-ray images within the GRAZPEDWRI-DX dataset, the model trained only on these images may not perform well in predicting other X-ray images. Therefore, to enhance the robustness of the model, we employ data augmentation techniques to expand the training set. More specifically, we fine-tune the contrast and brightness of the X-ray images using the addWeighted function available in OpenCV, which is an open-source computer vision library.

C. EVALUATION METRIC

1) Parameters (Params)

The quantity of parameters in a model depends on its architectural complexity, the number of layers, neurons per layer, and various other factors. More parameters in a model usually means a larger model size. Generally, the larger the model obtains the better model performance, but it also means that more data and computational resources are needed for training. In real-world applications, it is necessary to balance the relationship between model complexity and computational cost.

2) Floating Point Operations (FLOPs)

Floating Point Operations serves as a metric for assessing the performance of computers or computing systems and is commonly employed to evaluate the computational complexity of neural network models. FLOPs represent the number of floating-point operations executed per second, providing a crucial indicator of the computational performance and speed of the model. In resource-limited environments, models with lower FLOPs may be more suitable, while those with higher FLOPs may necessitate more powerful hardware support.

3) Mean Average Precision (mAP)

Mean Average Precision is a common metric used to evaluate the performance of object detection models. In the object detection task, the prediction aim of the model is to recognize objects within the image and determine their locations. Precision measures the proportion of objects detected by the model that correspond to the real objects, while recall measures the proportion of the real objects detected by the model. These two metrics need to be weighed, and mAP is a combination of precision and recall.

For each category, the model computes the area under the Precision-Recall curve, referred to as Average Precision. This metric reflects the performances of models for each category. We subsequently average the computed Average Precision values across all categories to derive the overall mAP.

4) F1-Score

The F1-Score is based on the summed average of the precision and recall values of the model. Its value ranges from 0 to 1, where value closer to 1 indicating that the model has a better balance between precision and recall. When one of precision and recall values is biased towards 0, the F1-Score will also be close to 0, which indicates poor model performance. Considering both precision and recall values together, the F1-Score helps to assess the accuracy of the model in positive category prediction and its sensitivity to positive categories.

TABLE 1: Experimental results of fracture detection on the GRAZPEDWRI-DX dataset using the YOLOv8-AM models with three different attention modules (i.e., ResCBAM, SA, and ECA).

Model Size	Input Size	mAP_{50}^{val}	mAP_{50-95}^{val}	Params (M)	FLOPs (B)	Inference (ms)
Small	640	61.6%	38.9%	16.06	38.27	1.9
Medium	640	62.8%	39.8%	33.84	98.19	2.9
Large	640	62.9%	40.1%	53.87	196.20	4.1
Small	1024	63.2%	39.0%	16.06	38.27	3.0
Medium	1024	64.3%	41.5%	33.84	98.19	5.7
Large	1024	65.8%	42.2%	53.87	196.20	8.7

(a) ResBlock + Convolutional Block Attention Module

Model Size	Input Size	mAP_{50}^{val}	mAP_{50-95}^{val}	Params (M)	FLOPs (B)	Inference (ms)
Small	640	62.7%	39.0%	11.14	28.67	1.7
Medium	640	63.3%	40.1%	25.86	79.10	2.5
Large	640	64.0%	41.5%	43.64	165.44	3.9
Small	1024	63.5%	39.8%	11.14	28.67	2.9
Medium	1024	64.1%	40.3%	25.86	79.10	5.2
Large	1024	64.3%	41.6%	43.64	165.44	8.0

(b) Shuffle Attention

Model Size	Input Size	mAP_{50}^{val}	mAP_{50-95}^{val}	Params (M)	FLOPs (B)	Inference (ms)
Small	640	61.4%	37.4%	11.14	28.67	1.9
Medium	640	62.1%	38.7%	25.86	79.10	2.5
Large	640	62.6%	40.2%	43.64	165.45	3.6
Small	1024	62.1%	38.7%	11.14	28.67	2.7
Medium	1024	62.4%	40.1%	25.86	79.10	5.2
Large	1024	64.2%	41.9%	43.64	165.45	7.7

(c) Efficient Channel Attention

In the problem involving imbalanced category classification, relying solely on accuracy for evaluation may lead to bias, as the model may perform better in predicting the category with a larger number of samples. Therefore, using the F1-Score provides a more comprehensive assessment of the model performance, especially when dealing with imbalanced datasets.

5) Inference Time

Inference Time refers to the duration required by network models to process X-ray images from input to the final prediction, including preprocessing, inference, and post-processing stages. In this work, the inference time per image is measured using single NVIDIA GeForce RTX 3090 GPU.

D. EXPERIMENT SETUP

We train the YOLOv8 model and different YOLOv8-AM models on the dataset [27]. In contrast to the 300 epochs recommended by Ultralytics [26] for YOLOv8 training, the experimental results [28] indicate that the best performance is achieved within 60 to 70 epochs. Consequently, we set 100 epochs for all models training.

For the hyperparameters of model training, we select the SGD [71] optimizer instead of the Adam [72] optimizer based on the result of the ablation experiment in study [28]. Following the recommendation of Ultralytics [26], this work establishes the weight decay of the optimizer at 5e-4, coupled with a momentum of 0.937, and the initial learning rate to 1e-2. To compare the effects of different input image sizes on the performance of the models, this work sets the input image size to 640 and 1024 for the experiments respectively. This work employs Python 3.9 for training all models on the framework of PyTorch 1.13.1. We advise readers to utilize versions higher than Python 3.7 and PyTorch 1.7 for model training, and the specific required environment can be accessed on our GitHub repository. All experiments are executed using one single NVIDIA GeForce RTX 3090 GPU, with the batch size of 16 set to accommodate GPU memory constraints.

E. EXPERIMENTAL RESULTS

In the fracture detection task, to compare the effect of different input image sizes on the performance of the YOLOv8-AM model, we train our model using training sets with input image sizes of 640 and 1024, respectively. Subsequently, we evaluate the performance of the YOLOv8-AM model based on different attention modules on the test set with the corresponding image sizes.

As shown in TABLEs 1 and 2, the performance of the models trained using the training set with input image size of 1024 surpasses that of models trained using the training set with input image size of 640. Nevertheless, it is noteworthy

TABLE 2: Experimental results of fracture detection on the GRAZPEDWRI-DX dataset using the YOLOv8-AM models with two attention modules (i.e., GAM and ResGAM).

Model Size	Input Size	mAP_{50}^{val}	mAP_{50-95}^{val}	Params (M)	FLOPs (B)	Inference (ms)
Small	640	0.625	0.397	13.86	34.24	2.2
Medium	640	0.628	0.398	30.27	90.26	3.6
Large	640	0.633	0.407	49.29	183.53	8.7
Small	1024	0.635	0.400	13.86	34.24	4.3
Medium	1024	0.637	0.405	30.27	90.26	8.9
Large	1024	0.642	0.410	49.29	183.53	12.7

(a) Global Attention Mechanism

Model Size	Input Size	mAP_{50}^{val}	mAP_{50-95}^{val}	Params (M)	FLOPs (B)	Inference (ms)
Small	640	61.4%	38.6%	13.86	34.24	2.7
Medium	640	62.8%	40.5%	30.27	90.26	3.9
Large	640	64.0%	41.2%	49.29	183.53	9.4
Small	1024	64.8%	41.2%	13.86	34.24	4.4
Medium	1024	64.9%	41.3%	30.27	90.26	12.4
Large	1024	65.0%	41.8%	49.29	183.53	18.1

(b) ResBlock + Global Attention Mechanism

TABLE 3: Quantitative comparison (F1-Score/mAP/Inference) of fracture detection on the GRAZPEDWRI-DX dataset using YOLOv8 and YOLOv8-AM models. Best and 2nd best performance are in red and blue colors, respectively.

Module	N/A	ResCBAM	SA	ECA	GAM	ResGAM
Params	43.61M	53.87M	43.64M	43.64M	49.29M	49.29M
FLOPs	164.9B	196.2B	165.4B	165.5B	183.5B	183.5B
F1-Score	0.62	0.64	0.63	0.65	0.65	0.64
mAP_{50}^{val}	63.6%	65.8%	64.3%	64.2%	64.2%	65.0%
mAP_{50-95}^{val}	40.4%	42.2%	41.6%	41.9%	41.0%	41.8%
Inference	7.7ms	8.7ms	8.0ms	7.7ms	12.7ms	18.1ms

that this improvement in performance is accompanied by an increase in inference time. For example, considering the ResCBAM-based YOLOv8-AM model with the large model size, the mean Average Precision at IoU 50 (mAP 50) attains 42.2% for the input image size of 1024, which is 5.24% higher than that of 40.1% obtained for the input image size of 640. However, the inference time of the model increases from 4.1ms to 8.7ms because the model size becomes larger.

TABLE 1 shows the model performance of the YOLOv8-AM, employing three different attention modules including ResCBAM, SA and ECA. These experimental results are obtained with different model sizes and different input image sizes. In TABLE 2, the model performance of the YOLOv8-AM is presented when utilizing GAM directly. Additionally, we introduce a novel approach by incorporating ResBlock and GAM, named ResGAM, to enhance the overall model performance of the YOLOv8-AM. Specifically, when the input image size is 1024 and the model size is medium, the newly introduced ResGAM demonstrates a notable enhancement in mAP for the YOLOv8-AM model based on GAM. The mAP 50 increases from 63.7% to 64.9%, providing that our proposed ResGAM positively contributes to the performance enhancement.

To compare the effect of different attention modules on the model performance, we organize the experimental data with input image size of 1024 and model size of large. The corresponding results are presented in TABLE 3. In summary, the F1-Score, mAP 50-95, and mAP 50 values for all YOLOv8-AM models surpass that of the YOLOv8 model. Specifically, the mAP 50 for the YOLOv8-AM models based on SA and ECA stands at 64.3% and 64.2%, respectively. These values are marginally superior to 63.6% mAP 50 obtained by the YOLOv8 model. Notably, this enhanced performance requires almost the same inference time as that of the YOLOv8 model. For the YOLOv8-AM model based on ResCBAM, it obtained mAP 50 of 65.8%, achieving the SOTA model performance. However, the gain on the performance of the YOLOv8 model by GAM is not satisfactory, so we propose the incorporation of ResGAM into the YOLOv8-AM model.

In this paper, to evaluate the gain of the attention module on the accuracy of the YOLOv8 model for predicting fractures in a real-world diagnostic scenario, four X-ray images are randomly selected, and FIGURE 7 demonstrates the prediction results of different YOLOv8-AM models. The YOLOv8-AM model, serving as a CAD tool, plays a crucial role in supporting radiologists and surgeons during diagnosis by effectively identifying fractures and detecting metal punctures in singular fracture scenarios. However, it is important to note that the accuracy of model prediction may decrease in instances involving dense fractures.

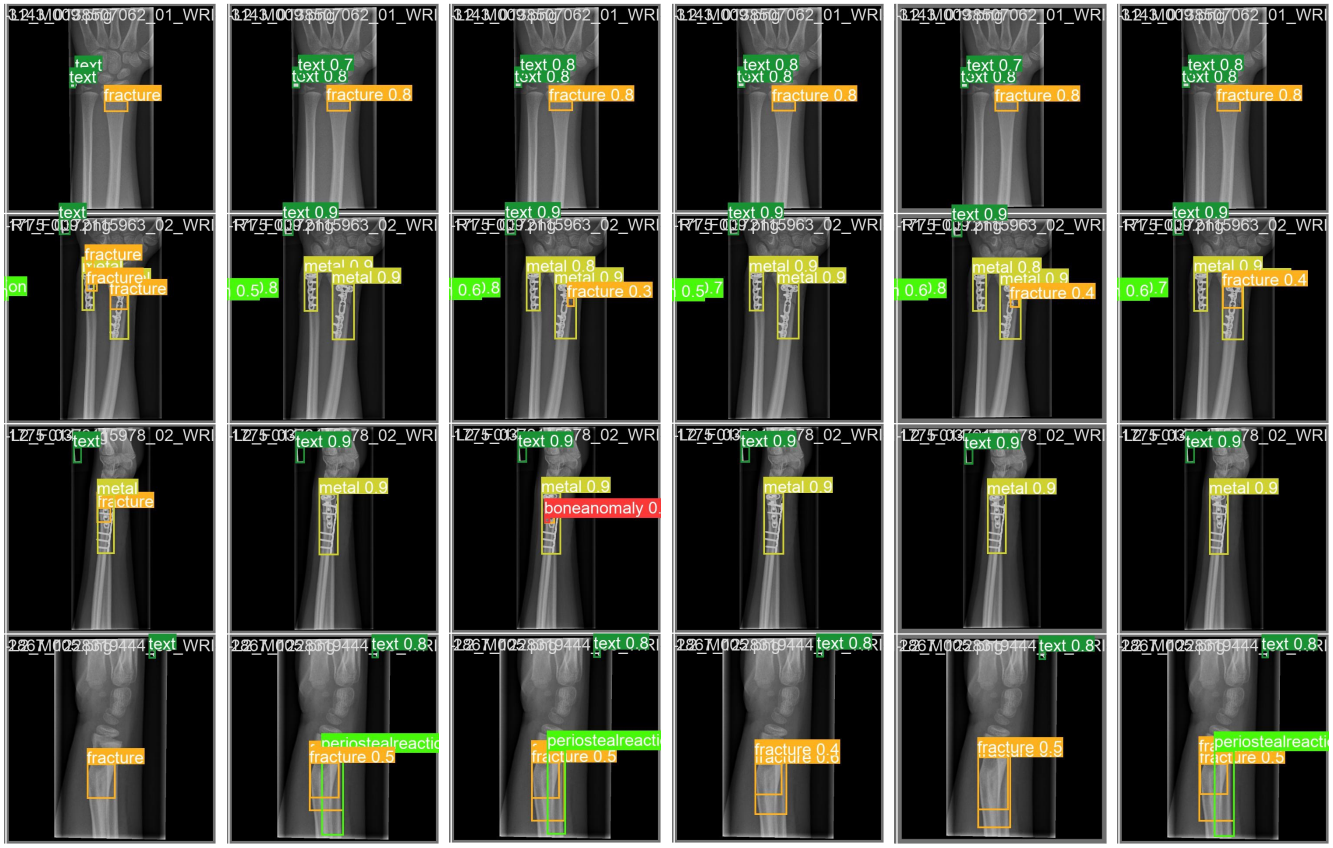

FIGURE 7: Examples of results of different YOLOv8-AM models applied to pediatric wrist fracture detection and Ground-Truth, where (a) manually labeled data; (b) ResCBAM; (c) ECA; (d) SA; (e) GAM; and (f) ResGAM.

FIGURE 8 shows the Precision-Recall Curve (PRC) for each category predicted by different YOLOv8-AM models. From the figure, we can see that different YOLOv8-AM models have greater ability to correctly detect fracture, metal, and text, with the average accuracy exceeding 90%. However, for two categories, bone anomaly and soft tissue, the ability to correctly detect them is poorer, with accuracy approximately at 45% and 30%, respectively. These low accuracy rates seriously effect the mAP 50 values of the models. We consider this is due to the small number of objects within these two categories in the used dataset. As described in GRAZPEDWRI-DX [27], the number of bone anomaly and soft tissue accounts for 0.41% and 0.68% of the total number of objects, respectively. Consequently, any improvement in model performance via architectural enhancements is constrained by this data limitation. To enhance the performance of the model, a recourse to incorporating extra data becomes imperative.

V. DISCUSSION

It is evident from TABLE 3 that the gain of GAM on the model performance of the YOLOv8-AM is poor on the GRAZPEDWRI-DX dataset. To further enhance the performance of the YOLOv8-AM model based on GAM, we have designed ResGAM, but it is still not as good as the performance gain provided by ResCBAM. According to the related theory [73], we think this is due to the decision of GAM to abandon pooling. In the attention mechanism, pooling serves to extract crucial features in each channel, thereby facilitating a concentration on the important aspects.

[39] demonstrated performance enhancements surpassing those achieved with CBAM on the CIFAR100 [74] and ImageNet-1K [75] datasets by enabling the neural network to acquire features across all dimensions, leveraging the remarkable adaptability inherent in the neural network. Nevertheless, it is noteworthy that the CIFAR100 dataset comprises 50,000 images representing diverse scenes in its training set, while the training set of the ImageNet-1K dataset includes a total of 1,281,167 images. In contrast, our model is trained using a small training set of 14,204 X-ray images. Consequently, the neural network is only required to learn the important features, such as bone fractures and lesions, within the X-ray images. This situation is different from the theory proposed by [39], given the limited scope of our dataset and the specific focus on relevant features on X-ray images.

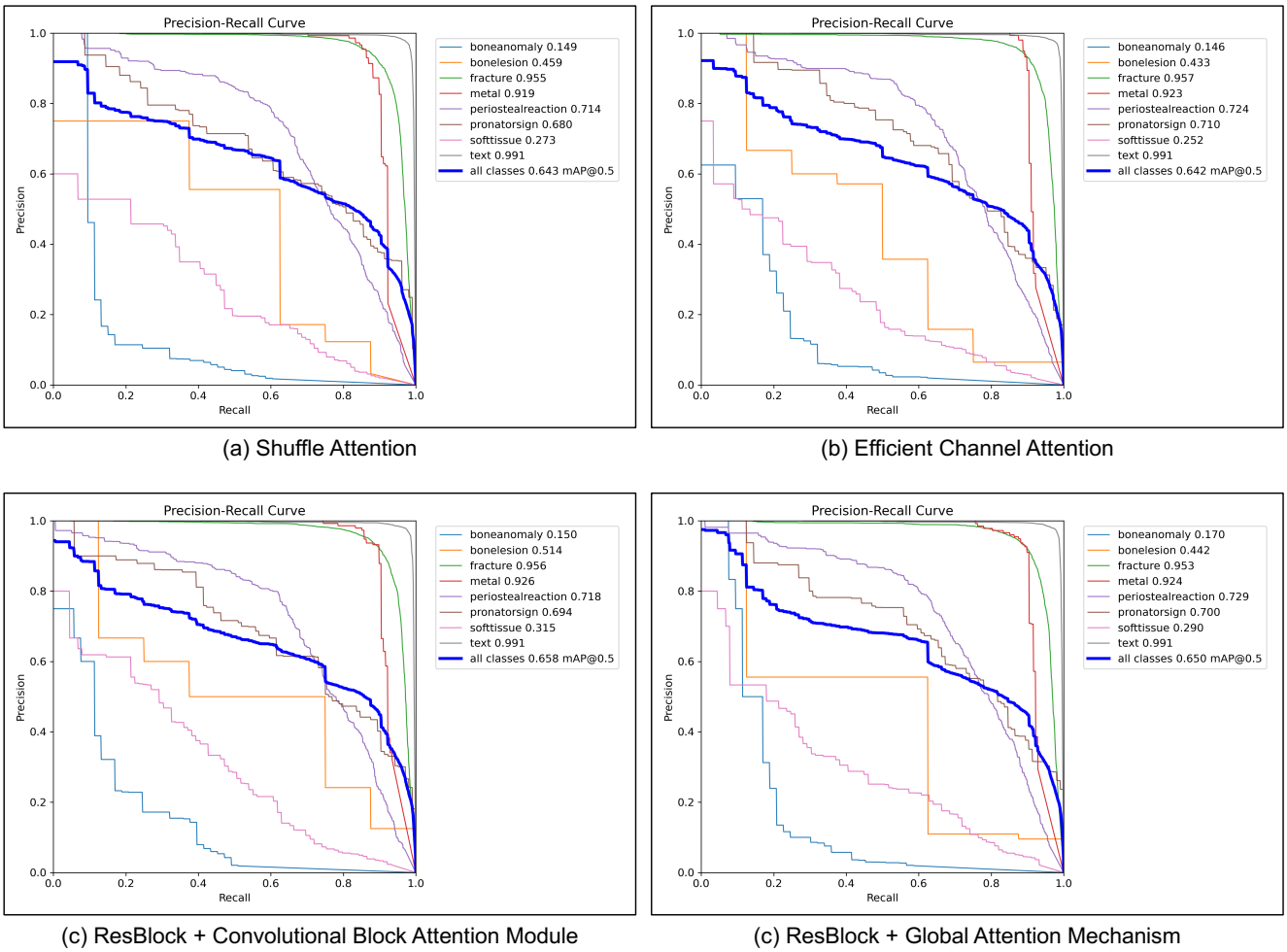

FIGURE 8: Detailed illustrations of the Precision-Recall Curves (%) for four different YOLOv8-AM models on each category in the GRAZPEDWRI-DX dataset, where (a) SA; (b) ECA; (c) ResCBAM; and (d) ResGAM.

VI. CONCLUSION AND FUTURE WORKS

Following the introduction of the YOLOv8 model by Ultralytics in 2023, researchers began to employ it for the detection of fractures across various parts of the body. While the YOLOv8 model, the latest version of the YOLO models, demonstrated commendable performance on the GRAZPEDWRI-DX dataset, it fell short of achieving the SOTA. To address this limitation, we incorporate four attention modules (CBAM, ECA, SA, and GAM) into the YOLOv8 architecture respectively to enhance the model performances. Additionally, we combine ResBlock with CBAM and GAM to form ResCBAM and ResGAM, respectively. Notably, the mAP 50 for the YOLOv8-AM model based on ResGAM improves from 64.2% (GAM) to 65.0% without increasing the model Parameters and FLOPs. Meanwhile, the mAP 50 for the YOLOv8-AM model with ResCBAM obtains a superior performance of 65.8%, surpassing the SOTA benchmark.

To support this work's application as CAD tools in medical imaging diagnosis, we plan to deploy the proposed YOLOv8-AM models as web and mobile applications (e.g., Android and iOS). This will ensure ease of use for both surgeons and medical professionals.

REFERENCES

[1] E. M. Hedström, O. Svensson, U. Bergström, and P. Michno, "Epidemiology of fractures in children and adolescents: Increased incidence over the past decade: a population-based study from northern sweden," Acta orthopaedica, vol. 81, no. 1, pp. 148–153, 2010.

[2] P.-H. Randsborg, P. Gulbrandsen, J. Š. Benth, E. A. Sivertsen, O.-L. Hammer, H. F. Fuglesang, and A. Årøen, "Fractures in children: epidemiology and activity-specific fracture rates," JBJS, vol. 95, no. 7, p. e42, 2013.

[3] R. Bamford and D.-M. Walker, "A qualitative investigation into the rehabilitation experience of patients following wrist fracture," Hand Therapy, vol. 15, no. 3, pp. 54–61, 2010.

[4] R. Kraus and L. Wessel, "The treatment of upper limb fractures in children and adolescents," Deutsches Ärzteblatt International, vol. 107, no. 51-52, p. 903, 2010.

[5] A. B. Wolbarst, Looking within: how X-ray, CT, MRI, ultrasound, and other medical images are created, and how they help physicians save lives. Univ of California Press, 1999.

[6] S. S. Boochever, "His/ris/pacs integration: getting to the gold standard." Radiology management, vol. 26, no. 3, pp. 16–24, 2004.

[7] T. K. Burki, "Shortfall of consultant clinical radiologists in the uk," The Lancet Oncology, vol. 19, no. 10, p. e518, 2018.

[8] A. Rimmer, "Radiologist shortage leaves patient care at risk, warns royal college," BMJ: British Medical Journal (Online), vol. 359, 2017.
[9] D. A. Rosman, J. J. Nshizirungu, E. Rudakemwa, C. Moshi, J. de Dieu Tuyisenge, E. Uwimana, and L. Kalisa, "Imaging in the land of 1000 hills: Rwanda radiology country report," Journal of Global Radiology, vol. 1, no. 1, 2015.
[10] E. Erhan, P. Kara, O. Oyar, and E. Unluer, "Overlooked extremity fractures in the emergency department," Ulus Travma Acil Cerrahi Derg, vol. 19, no. 1, pp. 25–8, 2013.
[11] J. Mounts, J. Clingenpeel, E. McGuire, E. Byers, and Y. Kireeva, "Most frequently missed fractures in the emergency department," Clinical pediatrics, vol. 50, no. 3, pp. 183–186, 2011.
[12] S. J. Adams, R. D. Henderson, X. Yi, and P. Babyn, "Artificial intelligence solutions for analysis of x-ray images," Canadian Association of Radiologists Journal, vol. 72, no. 1, pp. 60–72, 2021.
[13] J. W. Choi, Y. J. Cho, S. Lee, J. Lee, S. Lee, Y. H. Choi, J.-E. Cheon, and J. Y. Ha, "Using a dual-input convolutional neural network for automated detection of pediatric supracondylar fracture on conventional radiography," Investigative radiology, vol. 55, no. 2, pp. 101–110, 2020.
[14] S. W. Chung, S. S. Han, J. W. Lee, K.-S. Oh, N. R. Kim, J. P. Yoon, J. Y. Kim, S. H. Moon, J. Kwon, H.-J. Lee et al., "Automated detection and classification of the proximal humerus fracture by using deep learning algorithm," Acta orthopaedica, vol. 89, no. 4, pp. 468–473, 2018.
[15] L. Tanzi, E. Vezzetti, R. Moreno, A. Aprato, A. Audisio, and A. Massè, "Hierarchical fracture classification of proximal femur x-ray images using a multistage deep learning approach," European journal of radiology, vol. 133, p. 109373, 2020.
[16] C. Blüthgen, A. S. Becker, I. V. de Martini, A. Meier, K. Martini, and T. Frauenfelder, "Detection and localization of distal radius fractures: Deep learning system versus radiologists," European journal of radiology, vol. 126, p. 108925, 2020.
[17] K. Gan, D. Xu, Y. Lin, Y. Shen, T. Zhang, K. Hu, K. Zhou, M. Bi, L. Pan, W. Wu et al., "Artificial intelligence detection of distal radius fractures: a comparison between the convolutional neural network and professional assessments," Acta orthopaedica, vol. 90, no. 4, pp. 394–400, 2019.
[18] D. Kim and T. MacKinnon, "Artificial intelligence in fracture detection: transfer learning from deep convolutional neural networks," Clinical radiology, vol. 73, no. 5, pp. 439–445, 2018.
[19] R. Lindsey, A. Daluiski, S. Chopra, A. Lachapelle, M. Mozer, S. Sicular, D. Hanel, M. Gardner, A. Gupta, R. Hotchkiss et al., "Deep neural network improves fracture detection by clinicians," Proceedings of the National Academy of Sciences, vol. 115, no. 45, pp. 11 591–11 596, 2018.
[20] E. Yahalomi, M. Chernofsky, and M. Werman, "Detection of distal radius fractures trained by a small set of x-ray images and faster r-cnn," in Intelligent Computing: Proceedings of the 2019 Computing Conference, Volume 1. Springer, 2019, pp. 971–981.
[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.
[22] R.-Y. Ju, C.-C. Chen, J.-S. Chiang, Y.-S. Lin, and W.-H. Chen, "Resolution enhancement processing on low quality images using swin transformer based on interval dense connection strategy," Multimedia Tools and Applications, vol. 83, no. 5, pp. 14 839–14 855, 2024.
[23] F. Hržić, S. Tschauner, E. Sorantin, and I. Štajduhar, "Fracture recognition in paediatric wrist radiographs: An object detection approach," Mathematics, vol. 10, no. 16, p. 2939, 2022.
[24] P. Samothai, P. Sanguansat, A. Kheaksong, K. Srisomboon, and W. Lee, "The evaluation of bone fracture detection of yolo series," in 2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC). IEEE, 2022, pp. 1054–1057.
[25] Z. Su, A. Adam, M. F. Nasrudin, M. Ayob, and G. Punganan, "Skeletal fracture detection with deep learning: A comprehensive review," Diagnostics, vol. 13, no. 20, p. 3245, 2023.
[26] G. Jocher, A. Chaurasia, and J. Qiu, "Yolo by ultralytics," Code repository, 2023.
[27] E. Nagy, M. Janisch, F. Hržić, E. Sorantin, and S. Tschauner, "A pediatric wrist trauma x-ray dataset (grazpedwri-dx) for machine learning," Scientific data, vol. 9, no. 1, p. 222, 2022.
[28] R.-Y. Ju and W. Cai, "Fracture detection in pediatric wrist trauma x-ray images using yolov8 algorithm," Scientific Reports, vol. 13, no. 1, p. 20077, 2023.
[29] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 603–612.
[30] H. Lee, H.-E. Kim, and H. Nam, "Srm: A style-based recalibration module for convolutional neural networks," in Proceedings of the IEEE/CVF International conference on computer vision, 2019, pp. 1854–1862.
[31] X. Li, Z. Zhong, J. Wu, Y. Yang, Z. Lin, and H. Liu, "Expectation-maximization attention networks for semantic segmentation," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 9167–9176.
[32] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. C. Loy, D. Lin, and J. Jia, "Psanet: Point-wise spatial attention network for scene parsing," in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 267–283.
[33] Z. Zhu, M. Xu, S. Bai, T. Huang, and X. Bai, "Asymmetric non-local neural networks for semantic segmentation," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 593–602.
[34] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "Gcnet: Non-local networks meet squeeze-excitation networks and beyond," in Proceedings of the IEEE/CVF international conference on computer vision workshops, 2019, pp. 0–0.
[35] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 3146–3154.
[36] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 510–519.
[37] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, and J. Wang, "Ocnet: Object context network for scene parsing," arXiv preprint arXiv:1809.00916, 2018.
[38] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 3–19.
[39] Y. Liu, Z. Shao, and N. Hoffmann, "Global attention mechanism: Retain information to enhance channel-spatial interactions," arXiv preprint arXiv:2112.05561, 2021.
[40] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 11 534–11 542.
[41] Q.-L. Zhang and Y.-B. Yang, "Sa-net: Shuffle attention for deep convolutional neural networks," in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021, pp. 2235–2239.
[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[43] G. Jocher, K. Nishimura, T. Mineeva, and R. Vilariño, "yolov5," Code repository, p. 9, 2020.
[44] J. Burkow, G. Holste, J. Otjen, F. Perez, J. Junewick, and A. Alessio, "Avalanche decision schemes to improve pediatric rib fracture detection," in Medical Imaging 2022: Computer-Aided Diagnosis, vol. 12033. SPIE, 2022, pp. 611–618.
[45] H.-C. Tsai, Y.-Y. Qu, C.-H. Lin, N.-H. Lu, K.-Y. Liu, and J.-F. Wang, "Automatic rib fracture detection and localization from frontal and oblique chest x-rays," in 2022 10th International Conference on Orange Technology (ICOT). IEEE, 2022, pp. 1–4.
[46] K. Warin, W. Limprasert, S. Suebnukarn, T. Paipongna, P. Jantana, and S. Vicharueang, "Maxillofacial fracture detection and classification in computed tomography images using convolutional neural network-based models," Scientific Reports, vol. 13, no. 1, p. 3434, 2023.
[47] K. Warin, W. Limprasert, S. Suebnukarn, S. Inglam, P. Jantana, and S. Vicharueang, "Assessment of deep convolutional neural network models for mandibular fracture detection in panoramic radiographs," International Journal of Oral and Maxillofacial Surgery, vol. 51, no. 11, pp. 1488–1494, 2022.
[48] G. Yuan, G. Liu, X. Wu, and R. Jiang, "An improved yolov5 for skull fracture detection," in International Symposium on Intelligence Computation and Applications. Springer, 2021, pp. 175–188.
[49] M. Mushtaq, M. U. Akram, N. S. Alghamdi, J. Fatima, and R. F. Masood, "Localization and edge-based segmentation of lumbar spine vertebrae to identify the deformities using deep learning models," Sensors, vol. 22, no. 4, p. 1547, 2022.
[50] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132–7141.

[51] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng, "Aˆ 2-nets: Double attention networks," Advances in neural information processing systems, vol. 31, 2018.
[52] Z. Gao, J. Xie, Q. Wang, and P. Li, "Global second-order pooling convolutional networks," in Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, 2019, pp. 3024–3033.
[53] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "Cspnet: A new backbone that can enhance learning capability of cnn," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2020, pp. 390–391.
[54] C.-Y. Wang, H.-Y. M. Liao, and I.-H. Yeh, "Designing network design strategies through gradient path analysis," arXiv preprint arXiv:2211.04800, 2022.
[55] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 7464–7475.
[56] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.
[57] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8759–8768.
[58] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," Advances in neural information processing systems, vol. 28, 2015.
[59] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 6569–6578.
[60] X. Li, W. Wang, L. Wu, S. Chen, X. Hu, J. Li, J. Tang, and J. Yang, "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," Advances in Neural Information Processing Systems, vol. 33, pp. 21 002–21 012, 2020.
[61] Z. Zheng, P. Wang, D. Ren, W. Liu, R. Ye, Q. Hu, and W. Zuo, "Enhancing geometric factors in model learning and inference for object detection and instance segmentation," IEEE transactions on cybernetics, vol. 52, no. 8, pp. 8574–8586, 2021.
[62] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," in Proceedings of the AAAI conference on artificial intelligence, vol. 34, 2020, pp. 12 993–13 000.
[63] T. Jiang, C. Li, M. Yang, and Z. Wang, "An improved yolov5s algorithm for object detection with an attention mechanism," Electronics, vol. 11, no. 16, p. 2494, 2022.
[64] W. Li, K. Liu, L. Zhang, and F. Cheng, "Object detection based on an adaptive attention mechanism," Scientific Reports, vol. 10, no. 1, p. 11307, 2020.
[65] Y. Zhang, Y. Chen, C. Huang, and M. Gao, "Object detection network based on feature fusion and attention mechanism," Future Internet, vol. 11, no. 1, p. 9, 2019.
[66] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13. Springer, 2014, pp. 818–833.
[67] J. Park, S. Woo, J.-Y. Lee, and I. S. Kweon, "Bam: Bottleneck attention module," arXiv preprint arXiv:1807.06514, 2018.
[68] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2921–2929.
[69] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 116–131.
[70] Y. Wu and K. He, "Group normalization," in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 3–19.
[71] S. Ruder, "An overview of gradient descent optimization algorithms," arXiv preprint arXiv:1609.04747, 2016.
[72] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.
[73] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.
[74] A. Krizhevsky, G. Hinton et al., "Learning multiple layers of features from tiny images," Toronto, ON, Canada, 2009.
[75] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009, pp. 248–255.