Received: December 17, 2023. Revised: February 21, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Novel Approach of Hybrid Bounding Box Regression Mechanism to Improve Convergency Rate and Accuracy

Nugra Tasik Allo[1] **Indrabayu[1]*** **Zahir Zainuddin[1]**

[1]*Department of Informatics, Hasanuddin University, Indonesia*
* Corresponding author's Email: indrabayu@unhas.ac.id

Abstract: Bounding box regression is a commonly used technique aimed at enhancing the precision of object localization, which is crucial in the field of object recognition. The intersection over union (IoU) metric, which calculates the overlap between predicted and ground truth bounding boxes, is frequently used to evaluate the performance of object detection models. However, the MSE loss function used previously is not compatible with the IoU-based evaluation and has shown sensitivity to differences in object scales. The use of IoU as the basis for loss functions has become more common in recent years, and as a result, new techniques such as the Generalized IoU (GIoU) and complete IoU (CIoU) losses have grown to be developed. This paper introduces a hybrid mechanism called GCIoU loss, which combines GIoU and CIoU losses with the aim of further enhancing localization accuracy and convergence speed. According to our findings, the average precision (AP) is greatly improved by the GCIoU loss in comparison to the GIoU and CIoU losses by 7.72% (highest) to the basis of IoU loss, and 0.87% improvement to the CIoU loss. The GCIoU loss performs consistently well across different thresholds, especially at higher levels, and has improved AP75 by 3.07% compared to the CIoU loss as the default configuration. Additionally, GCIoU loss converges faster and even more robustly in the simulation experiments by taking 14% fewer epochs than the CIoU loss, leading to localization more precisely. By this, GCIoU loss is showing its usefulness in object identification and model optimization.

Keywords: Loss function, Bounding box regression, Intersection over union, Object detection.

1. Introduction

Object detection represents a fundamental and challenging task in the field of visual recognition. Its primary goal is to assign class labels to objects while simultaneously predicting their precise locations [1]. The successful application of deep neural networks in classification tasks has also contributed to the improvement of object detection, such as detection and learning components [2], neural network architecture [3], and object detection frameworks [4]. The standard performance metric used for object detection is known as Intersection over Union (IoU). When predicting an object within an image, IoU measures the similarity between the predicted region and its ground truth. It is calculated as the intersection's area divided by the union area of those two regions [5]. Intersection over Union is defined in Eq. (1).

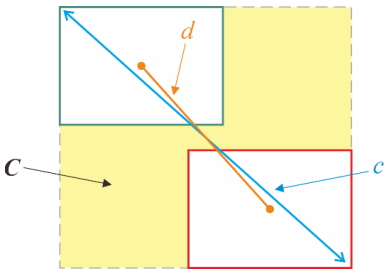


Figure. 1 Proposed GCIoU loss for bounding box regression, *C* is the area of the smallest enclosing box covering the predicted box and its ground truth, c is the diagonal length of C, and *d* is the distance (b, b^{gt}) of central points of the two boxes

$$IOU = \frac{|B^{pred} \cap B^{gt}|}{|B^{pred} \cup B^{gt}|} \quad (1)$$

where $B^{pred} = (x^{pred}, y^{pred}, w^{pred}, h^{pred})$ is the predicted box, and $B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$ is the ground truth. While early works on object detection use the l_n – norm (e.g., n = 1 or 2) loss to calculate the distance between the predicted box and its ground truth, also known as the MSE loss, defined in Eq. (2).

$$l_x(x) = \begin{cases} |x|, n = 1 \\ x^2, n = 2 \end{cases} \quad (2)$$

where x is the difference between the predicted box and the ground truth. However, using the natural logarithm of the l_n – norm loss as a loss function is not suitable. This is because when the l_n – norm loss calculated between predicted and ground truth is the same at different stages, their IoU may differ, making it inconsistent with the IoU as an evaluation metric[6].

Since then, the evolution of loss metrics has reached a point where the l_n – norm is no longer a viable option. This is because the l_n – norm loss treats all the coordinates of the bounding box ($x = x_{top}, x_{bottom}, x_{left}, x_{right}$) differently, in other hand the the l_n – norm loss primarily focuses on penalizing differences in coordinates. These causing failure when one or two bounds of a predicted box are very close to the ground truth box. Recent research has proposed IoU as a loss function calculation, not just as an evaluation metric. The IoU loss is stated in Eq. (3).

$$L_{IoU} = 1 - IoU \quad (3)$$

The successful of IoU loss has outperformed the l_n – norm loss in the context of improving performance. This is achieved not by regressing the bounding box as four different components, but as a single unit [7, 8]. However, IoU loss also comes with a major disadvantage when the predicted box do not overlap with its ground truth, making it not provide any moving gradient for non-overlapping boxes because the value will always be zero [9]. The non-overlapping boxes could make the model struggle to localize accurately and less robust.

To alleviate this shortcoming, H. Rezatofighi [9] proposed an IoU-based loss, namely Generalized Intersection over union (GIoU), as a new loss and evaluation metric. The GIoU loss effectively alleviates the IoU loss gradient vanishing problem caused by non-overlapping boxes by adding the third box as the smallest enclosing bounding box. This new box will be a border to cover and also provide a more informative measure about the dissimilarity between the ground truth box and the predicted box. GIoU loss suffers from two main limitations, first, it may degrade to the IoU loss when the predicted box is equal in size to the smallest enclosing box, resulting gradient of zero, and second, GIoU loss requires more iterations to converge, especially for vertical or horizontal cases. To address the weaknesses of GIoU loss, Z. Zheng [10] proposed Distance IoU (DIoU) loss and complete IoU (CIoU) loss as faster and better bounding box regression loss. These functions utilize the Euclidean distance between the center points of the two boxes to enhance the convergence speed. CIoU loss optimized the DIoU loss by adding calculation to the consistency of the aspect ratio for the prediction box to its ground truth. However, DIoU loss will degrade into IoU loss when the center points of the two boxes are at the same position, while CIoU loss will degenerate to DIoU loss when the aspect ratio of the predicted box is equal to the ground truth box. The sensitivity of CIoU loss to the same aspect ratio could lead to a poor convergence rate, decreasing the localization performance.

This paper resolves the above-mentioned issues by processing bounding boxes using different ratios at each iteration. The proposed method uses a hybrid mechanism to leverage changes in size and ratios in the GIoU loss and then combine them with Euclidean distance in the CIoU loss, thereby making gradient calculations more efficient. This hybrid method can complement each other's shortcomings in both loss functions. The main contributions of this paper can be summarized as follows:

- We offer a novel approach using a hybrid method between GIoU loss and CIoU loss to cover both major limitations, where GIoU loss in this hybrid method will converge faster on vertical and horizontal cases.
- CIoU loss in our hybrid method will not degrade to DIoU loss because the aspect ratio of the two boxes will always be different due to the evolving predicted box of GIoU loss.
- By combining both advantages above, GCIoU loss will not only converge faster but also more robust on various positions and scales.

The remainder of this paper is organized as follows. Section 2 describes the literature on object detection in neural networks and the state-of-the-art losses as a comparison. Section 3 introduces our proposed loss for bounding box regression, called GCIoU loss, to improve the detection performance in object detection models. Section 4 shows the simulation and experimental results. Finally, section 5 concludes the results of this research.

2. Related works

2.1 Object detection

Object detection is a computer vision task that involves identifying and locating objects of interest within an image or a video frame. The goal is not only to classify the objects but also to draw a precise bounding box around the detected objects to pinpoint their precise location in the image.

In 2001, P. Viola and M. Jones [11] proposed a DJ detector on machine learning, which became the early real-time detection of fixed objects. Since then, advances in object detection have been made. The experimental results of this paper achieve high detection rates and minimize computation time, which is approximately 15 times faster than any previous approach. Since then, the improving of object detection has been put forward into improvements. In 2014, Girshick [12] proposed RCNN, which brought object detection into the deep neural network. RCNN applied high-capacity convolutional neural networks with bottom-up region proposals to localize and segment objects [13], resulting in a 30% relative improvement. One year later, Girshick [14] improved the RCNN into Fast-RCNN and Faster-RCNN. Right now, Faster-RCNN has become a popular two-stage algorithm for object detection [15, 16]. On the other hand, single-stage object detection, such as single shot multibox detector (SSD) [17] and you only look once (YOLO) series, has greatly improved the image processing speed. YOLO has been rapidly upgraded from YOLOv1 [18], YOLOv2 [19], YOLOv3 [20], and YOLOv4 [21] in just a few years.

2.2 Bounding box regression

Bounding box regression is a technique used in object detection and localization tasks to refine the coordinates of the bounding boxes around the detected objects. Many popular object detection algorithms use the l_n – norm as loss calculation, such as Faster-RCNN, SSD, or YOLO series. The IoU-based loss was first proposed in 2016 by J. Yu [7]. Subsequently, it was found that previous l_n – norm loss has found extensive use in traditional object detection networks, but sensitive to varying scales. The experiment replaced the l_n – norm with IoU loss and achieved better results on the face detection task. IoU loss performs accurately localizing objects on varied shapes and scales, and converges faster than the l_n – norm loss, but it suffers from non-overlapping boxes. Based on the IoU loss, in 2019, H. Rezatofighi [9] proposed GIoU loss to alleviate the gradient vanishing of the IoU loss when the predicted boxes do not overlap with the ground truth. GIoU loss was applied to YOLOv3, Faster-RCNN, and Mask-RCNN. The experimental results on PASCAL VOC 2007 and MS COCO 2014 showed that GIoU, as a bounding box regression loss, consistently improved performance and convergence speed. Although GIoU loss was the first method to implement IoU loss as its basis, it requires more iterations to fully converge. Within the same year, Z. Zheng in [10] and [22] proposed DIoU and CIoU losses which consider the normalization of both center points. CIoU loss calculates three geometric factors, i.e., overlap area, center coordinates, and aspect ratio. Proposed losses are then incorporated into YOLOv3, SSD, and faster RCNN algorithm. The evaluation results showed notable performance against IoU and GIoU losses in terms of localization performance and convergence speed. Furthermore, to enhance the performance improvement, DIoU loss can be easily incorporated into non-maximum supression (NMS) as the criterion. Which then leads to better localization. Popular YOLO series and other object detection algorithms then implement the CIoU loss as its default loss to converge the non-overlapping boxes while training the network, with GIoU loss and DIoU loss as other options. However, the CIoU loss is sensitive to the ratio of the same boxes, causing its convergence rate to decrease. Therefore, a method is needed to overcome this shortcoming.

3. Methodology

These improvements in object detection models have been a consequence of overcoming a major challenge, the slow convergence due to gradient vanishing in IoU loss, which is an essential part of precisely localizing and classifying objects within an image.

3.1 The solution to the gradient vanishing issue

In the field of bounding box regression, this topic faces a significant challenge, as accurate object localization requires precise fine-grained adjustments to bounding box coordinates [23]. When the gradient vanishes, which means the value is zero, the model struggles to make these fine adjustments, resulting in inaccurate and imprecise bounding box predictions [24]. To address this major problem, popular bounding box regression losses operate as follows:

3.1.1. Generalized intersection over union

H. Rezatofighi [9] proposed a penalty term to the non-overlapping bounding box in the GIoU loss. In

Eq. (4), C is the smallest enclosing box covering B and B^{gt}, while ∪ is the union area of the two boxes.

$$R_{GIoU} = \frac{|C - B \cup B^{gt}|}{|C|} \quad (4)$$

$$L_{GIoU} = 1 - IoU + R_{GIoU} \quad (5)$$

The GIoU loss aims to increase the size of the predicted box so that it overlaps with the ground truth, allowing the IoU term to work in maximizing the overlap. Although, GIoU loss is designed to prevent gradient disappearance, it still has some limitations, such as slow convergence speed especially for vertical or horizontal cases, and the potential to degrade to the IoU loss.

3.1.2. Distance intersection over union

Distance IoU elevates these shortcomings, Z. Zheng [10], [22] proposed DIoU loss by normalizing the center points distance of each bounding box.

$$R_{DIoU} = \frac{p^2(b,\ b^{gt})}{c^2} \quad (6)$$

$$L_{DIoU} = 1 - IoU + R_{DIoU} \quad (7)$$

where p^2 (b, b^{gt}) represents the Euclidean distance between the two center points, and c^2 represents the diagonal length of the smallest enclosing area covering the predicted box and ground truth box. When introducing the DIoU loss, its creators held the belief that an effective loss function should include three important and crucial geometric elements, i.e., the area of overlap, the distance between central points of both boxes, and the aspect ratio, which later becomes the CIoU loss.

3.1.3. Complete intersection over union

On the basis of DIoU loss, Z. Zheng[22] added the calculation of aspect ratio, shown in Eq. (8).

$$R_{CIoU} = \frac{p^2(b,\ b^{gt})}{c^2} + \alpha v \quad (8)$$

where α is a positive trade-off parameter, and v measures the consistency of the aspect ratio for the predicted box and ground truth box.

$$v = \frac{4}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right)^2 \quad (9)$$

$$\alpha = \frac{v}{(1-IoU)+V} \quad (10)$$

The calculation of aspect ratio v cannot provide gradients of the same w and h of the anchor box. The optimization process for CIoU loss corresponds to DIoU loss, with the exception of the gradient of v with respect to w and h.

$$\frac{\partial v}{\partial w} = \frac{8}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right)\left(\frac{h}{w^2+h^2}\right) \quad (11)$$

$$\frac{\partial v}{\partial h} = -\frac{8}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h}\right)\left(\frac{w}{w^2+h^2}\right) \quad (12)$$

For cases h and w ranging in [0,1], the denominator $w^2 + h^2$ is typically a small value, which likely to result in gradient explosion. Therefore, for stable convergence in CIoU loss implementation, the denominator is removed, replacing the step size of $\frac{1}{w^2+h^2}$ to 1, and maintaining the gradient direction consistency stated above.

3.2 Proposed method

Let $B = \{x_1^p, x_1^p, x_2^p, x_2^p\}$ be the predicted box, and $B^{gt} = \{x_1^{gt}, x_1^{gt}, x_2^{gt}, x_2^{gt}\}$ be the ground truth box. Intersection over Union, a critical assessment metric in object detection algorithms, measures the proportion of overlap between predicted and ground truth bounding boxes by dividing their intersection area by their union area. Generally, IoU-based losses can be formalized as in Eq. (13).

$$L_{IoU} = 1 - IoU + R(B, B^{gt}) \quad (13)$$

where R (B, B^{gt}) is the penalty term for the bounding box B and its ground truth box B^{gt}. Based on the limitations of existing bounding box regression losses from the previous section.

Factors such as distance and aspect ratio will affect the loss value because training data usually vary due to factors like poor image quality or different sizes. These variations can decrease the model's generalization performance and lead to variability in the bounding boxes it predicts. These aspects should be included by a suitable loss function to improve the model's ability to generalize better. When the aspect ratio of the predicted box and the ground truth box vary, the CIoU loss performs better than the GIoU loss and DIoU loss.

In Eq. (9) when $w^{gt} / h^{gt} \neq w / h$, so the v and the penalty term of αv has a positive role, leading into better normalization. However, when $w^{gt} / h^{gt} = w / h$, meaning there is no gradient for aspect ratio calculation, so CIoU loss degrades into DIoU loss. In the other hand, GIoU loss penalize the predicted box

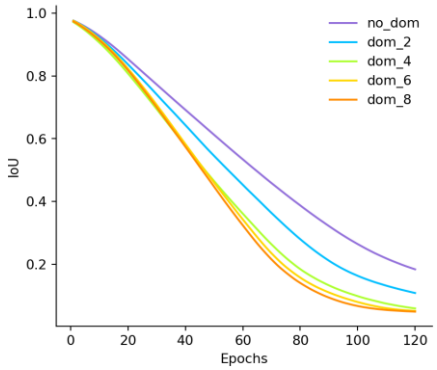


Figure. 2 The effect of different denominators scaling down the hybrid loss, with denominator 8 shows the fastest convergence speed. This result is part of the simulation experiments conducted on all cases scenario

Algorithm 1: GCIoU as bounding box loss

Input : $B = \{x_1^p, y_1^p, x_2^p, y_2^p\}$ and b

$B^{gt} = \{x_1^{gt}, y_1^{gt}, x_2^{gt}, y_2^{gt}\}$ and b^{gt}

output : GCIoU loss

1. Calculate area of $B = (x_2^p - x_1^p)(y_2^p - y_1^p)$

   For *B*, ensuring $x_2^p > x_1^p$ and $y_2^p > y_1^p$

   $\dot{x}_1^p = \min(x_1^p, x_2^p)$, $\dot{x}_2^p = \max(x_1^p, x_2^p)$

   $\dot{y}_1^p = \min(y_1^p, y_2^p)$, $\dot{y}_2^p = \max(y_1^p, y_2^p)$
2. Calculate area of $B^{gt} = (x_2^{gt} - x_1^{gt})(y_2^{gt} - y_1^{gt})$
3. Calculate the intersection I

   $x_1^I = \max(\dot{x}_1^p, x_1^{gt})$, $x_2^I = \min(\dot{x}_2^p, x_2^{gt})$

   $y_1^I = \max(\dot{y}_1^p, y_1^{gt})$, $y_2^I = \min(\dot{y}_2^p, x_2^{gt})$

$$I = \begin{cases} (x_2^I, x_1^I)(y_2^I, y_1^I) & if\ x_2^I > x_1^I, y_2^I > y_1^I \\ 0 & \text{otherwise} \end{cases}$$

4. Calculate area of C

   $x_1^C = \min(\dot{x}_1^p, x_1^{gt})$, $x_2^C = \max(\dot{x}_2^p, x_2^{gt})$

   $y_1^C = \min(\dot{y}_1^p, y_1^{gt})$, $y_2^C = \max(\dot{y}_2^p, x_2^{gt})$

   $C = (x_2^C - x_1^C)(y_2^C - y_1^C)$
5. Calculate union $U = B + B^{gt} - I$
6. Calculate diagonal c^2
7. Calculate the distance *u*

   $center_x^p = (x_1^p + x_2^p) / 2$, $center_x^{gt} = (x_1^{gt} + x_2^{gt}) / 2$

   $center_y^p = (y_1^p + y_2^p) / 2$, $center_x^{gt} = (x_1^{gt} + y_2^{gt}) / 2$

   $ux = (center_x^p - center_x^{gt})$

   $uy = (center_y^p - center_y^{gt})$

   $u = (dx, dy)^2$
8. Calculate distance $d = u / c^2$
9. Calculate the αv
10. IoU = I / U
11. $GCIoU = IoU - \frac{(\frac{C - B \cup B^{gt}}{C}) + \frac{p^2(b,\ b^{gt})}{c^2}}{8} + \alpha v$
12. GCIoU loss = 1 - GCIoU

not only to extent the overlap, but will also change both position and aspect ratio across iterations.

$$\frac{\partial AR}{\partial w\prime} = \frac{1}{h\prime} - \frac{w\prime}{(h\prime)^2}\frac{\partial h\prime}{\partial w\prime} \tag{14}$$

$$\frac{\partial AR}{\partial h\prime} = -\frac{w'}{h^2} + \frac{w\prime}{h^3}\frac{\partial h\prime}{\partial h\prime} \tag{15}$$

where $(\partial AR / \partial w')$ is the aspect ratio changes when the predicted *w'* is adjusted, the $(1 / h')$ is the natural change in aspect ratio if we want to change the *w'* while keeping the *h'* constant. The $(-\frac{w\prime}{(h\prime)^2}\frac{\partial h\prime}{\partial w\prime})$ considers how changes in *h'* due to adjustments in width affect the aspect ratio. For the second term, $(\partial AR / \partial w')$ tells how the aspect ratio changes concerning adjustments to the *h'*. The $(- w' / h^2)$ represents the natural change in aspect ratio if we change the *h'* while keeping the *w'* constant, and the $\frac{w\prime}{h^3}\frac{\partial h\prime}{\partial h\prime}$ accounts how changes in the *w'* due to adjustments in *h'* affect the aspect ratio. This is the effect of modifying the *w'* while changing the *h*'. For simplify, both Eq. (14) and Eq. (15) show how the predicted box ratio of GIoU loss evolves when the *w'* and *h'* adjusted. Based on the limitation of CIoU loss that sensitive to same aspect ratio and the evolving predicted aspect ratio of GIoU loss, we propose a hybrid hybrid mechanism for both losses in Eq. (16).

Our proposed loss, namely the GCIoU loss, will calculate different aspect ratio for each iteration, leading to better gradient calculation and faster optimalization.

$$R_{GCIOU} = \frac{(\frac{C - B \cup B^{gt}}{C}) + \frac{p^2(b,\ b^{gt})}{c^2}}{8} + \alpha v \tag{16}$$

where *C* is the area of smallest enclosing box covering predicted box and ground truth box, *b* is the center point of predicted box, b^{gt} is the center point of ground truth box, *c* is the diagonal length of *C*. The αv is used to calculate the consistency of the aspect ratio in Eq. (9) and Eq. (10) as explained in CIoU loss above. It divides by *C* and *c2* to normalize the calculation of both losses. As a result, the GCIoU loss is defined as in Eq. (17).

$$L_{GCIOU} = 1 - IOU + R_{GCIOU} \tag{17}$$

The crucial step in this proposed loss is located on the denominator, in Fig. 2 shows intriguing result on how numerous different denominators affect the convergence speed. When combining loss functions, the gradient during backpropagation process become excessively large. So, in this study we try scale down

the gradients by trying to use various constant. In Table 1 below, we experimented different numbers as denominators to achieve better IoU performance. The loss function plays an important role in object detection algorithm, by measuring the difference between the predicted box and the ground truth box to provide a quantitative measure of how well the model is performing. The goal of the loss function is to minimize the difference loss of both boxes while training. The lower its loss, the more accurate the predicted bounding box is compared to the ground truth box, meaning the higher its IoU value.

Table 1. The IoU value across different denominators along with the GIoU loss and CIOU loss. This experiment conducted on all cases scenario.

Denominators	IoU
no_dom	0.838
GIoU	0.864
dom_2	0.910
dom_4	0.941
CIoU	0.947
dom_6	0.951
dom_8	**0.952**

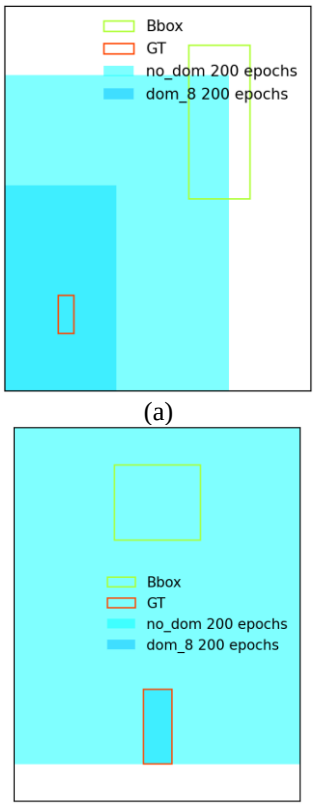


Figure. 3 The difference of no_dom with dom_8 affect the size of the predicted box on simulation experiments for: (a) diagonal and (b) horizontal cases. This experiment also conducted on all cases scenario

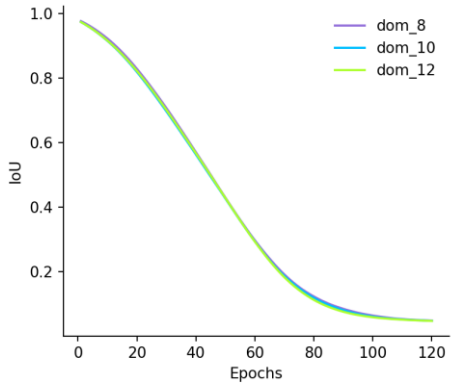


Figure. 4 Convergence rate compared to higher number than 8 as denominator

In Fig. 3 for (a) diagonal and (b) horizontal position cases, the observation by scaling down the gradients by 8 leads to a smaller predicted bounding box size, while on the other hand, using a lower value resulting a larger box. The denominators are not directly dividing specific physical quantities but rather scaling down the gradients for updating the parameters. If the loss function L is scaled down by a denominator α, with respect to a parameter θ is also scaled by α. In Table 1, each divisor also comes with a different IoU value compared to GIoU loss and CIoU loss, divisor of 8 comes with the best result. While in Fig. 4, using higher number than 8 as denominator will not affect much on the convergence rate.

4. Experimental results

4.1 Dataset and experimental environment

We conduct the experiment on the popular benchmark PASCAL VOC 2007. PASCAL VOC is one of the most popular established benchmarks to evaluate the performance of algorithms in object detection. It contains 20 different classes and the annotation format is XML-based [25]. The experimental environment in this paper was configured as follows: AMD Ryzen™ 5 7535HS (3.30 GHz 6 cores), 16GB memory, Windows 11 Pro 64-bit, NVIDIA RTX 4050 Mobile GPU with 6GB of video memory, CUDA 12.2 and cuDNN 8.9.4.25.

4.2 Evaluation protocol

The proposed loss function was evaluated by incorporating it into YOLO v4. YOLO v4 is a popular single-stage detection algorithm at present, because YOLO series offer high speed and efficiency in detecting objects. To evaluate the experimental

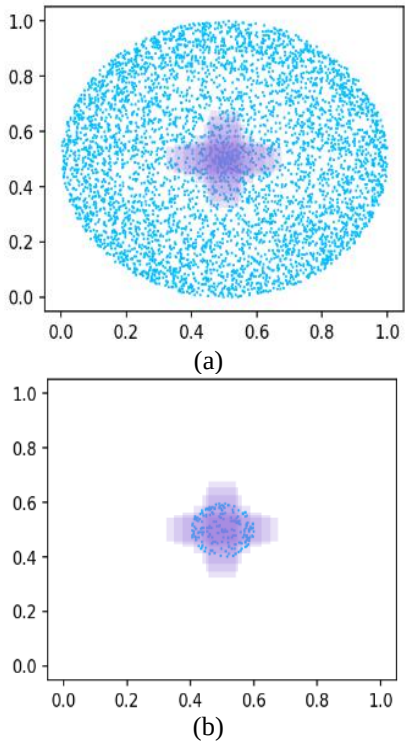


Figure. 5 Anchor points (blue) and target boxes (purple) on simulation experiments: (a) all cases and (b) major cases

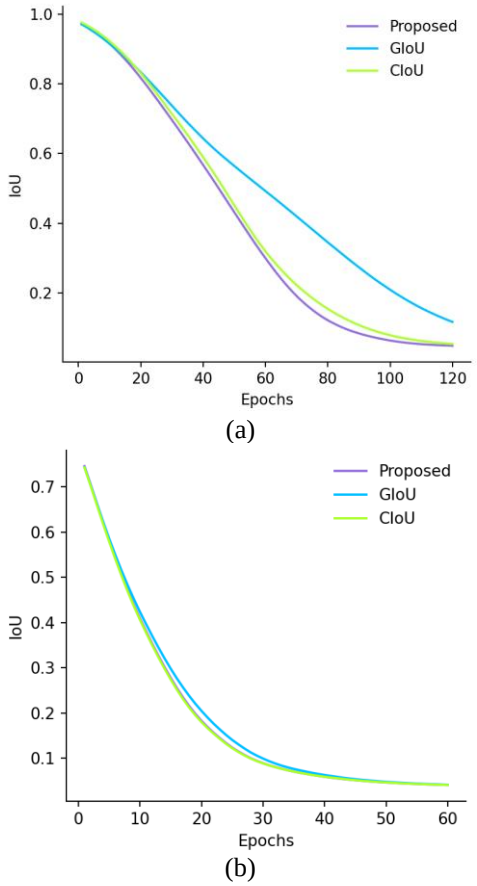


Figure. 6 Optimization curves on different losses for: (a) all cases, and (b) major cases

results, we use AP = (AP50 + AP55 + … AP95)/10 which means the mean of AP values on different ten thresholds.

4.3 Simulation experiment

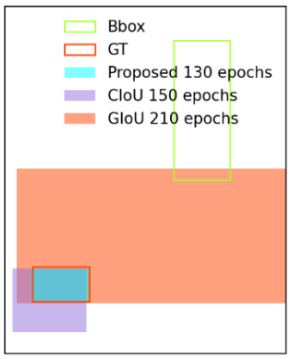
For evaluation, we employ the simulation experiment suggested by Z. Zheng [10] on Z.Tong [26] experiments to compare each loss function for bounding box regression in a preliminary manner. We generate target boxes of seven aspect ratios (i.e., 1:4, 1:3, 1:2, 1:1, 2:1, 3:1, 4:1) at (0.5, 0.5), all of which have an area of 1/32. In a circular region centered at (0.5, 0.5) with a radius of r, $20000r^2$ anchor points are consistently produced. While 49 anchor boxes with seven scales (i.e., 1/321/24, 3/64, 1/16, 1/12, 3/32, 1/8) and seven aspect ratios (i.e., 1:4, 1:3, 1:2, 1:1, 2:1, 3:1, 4:1) are positioned for each anchor points. The anchor boxes must be matched to the target boxes, and the regression cases are $6860000r^2$. We set up the following experimental settings in order to compare the convergence rate over various time periods. The experiments are divides into two cases, one for the case where there are overlapping and non-overlapping boxes, and the other one is just for overlapping boxes.

In Fig. 5, the (a) all cases simulation represents r = 0.5, where the anchor boxes are positioned both inside and outside the target box's coverage area. For (b) major cases simulation, the r = 0.1, making anchor boxes are created in the target box's coverage area. Both cases represent each case in the bounding box regression.

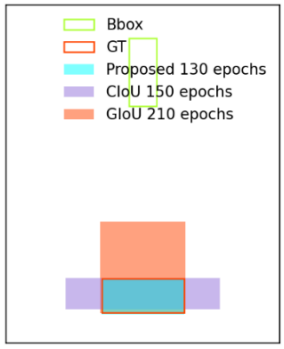
4.4 Experimental results

IoU loss works only when the bounding boxes have overlap, it would not give any moving gradient for non-overlapping scenarios. So, GIoU loss aims to expand the size of predicted box at first, making it have overlap with target box, and then the IoU term will work. Since GIoU depends highly on the IoU term, it takes more iterations empirically to converge. So, DIoU loss and CIoU loss directly normalized the distance between central points, making it converge faster. The simulation results between GIoU loss and CIoU loss compared to our GCIoU loss are shown below.

According to Fig. 6 its shown that our proposed GCIoU loss is faster than the CIoU loss in (a) all cases scenarios. Furthermore, for the (b) major cases, all losses have extremely similar converge rates. From both cases conducted, it appears that non-overlapping bounding boxes represent the majority of

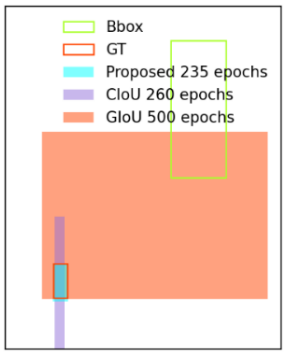


(a)

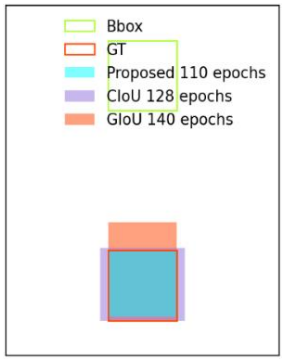


(b)

Figure. 7 Regression results using GIoU loss, CIoU loss, and our proposed GCIoU loss on different bounding box aspect ratio for: (a) diagonal and (b) horizontal cases



(a)



(b)

Figure. 8 Regression results using GIoU loss, CIoU loss, and our proposed GCIoU loss on the same bounding box aspect ratio for: (a) diagonal and (b) horizontal cases

the variation in converge rates. We conduct another simulation for the non-overlapping cases shown below.

During training, model may provide many bounding box predictions for an object within an image. To achieve precise object localization with fewer epoch, a good loss function should handle gradients efficiently. Accurate localization has a major effect on the overall model performance of the model. Our proposed GCIoU loss consistently handles all gradients quickly and efficiently in various scenarios of different positions and scales, this indicates that our proposed GCIoU loss is suitable to multiple situations.

The regression findings in Fig. 7 demonstrate that our proposed GCIoU loss achieves complete overlap with the ground truth in fewer iterations, specifically, only after 130 epochs in the case of diagonal bounding box position with different aspect ratios. The CIoU loss is still being adjusted at 150 epochs, whereas the GIoU loss is being adjusted at 210 epochs. In the case of the horizontal positioning, our proposed GCIoU loss achieved its convergence at 130 epochs, whereas the CIoU and GIoU losses are still being optimized at 150 and 210 epochs, respectively. In the case of bounding boxes with the the same aspect ratios shown in Fig. 8, our proposed GCIoU loss achieves full overlap after 235 epochs, while the CIoU loss is still attempting to achieve it after 260 epochs. On the other hand, the GIoU loss fails to achieve full overlap even after 500 epochs. Finally, in the horizontal scenario with equal aspect ratios, our proposed GCIoU requires just 110 epochs, although the optimization of CIoU and GIoU losses is still attempting to converge. From the simulation cases above, it's clear that our proposed GCIoU loss is optimal for four different positions because it takes fewer iterations to converge and its more robust to various scales. For further analysis, we documented average precision (AP) in detail for each of the 20 different categories in the PASCAL VOC 2007 dataset. The goal of this analysis is to provide information regarding the richness and variety of the losses. We provided a thorough comparison of the performance indicators related to the different loss functions that were used in order to compare each loss. This evaluation provides a deep understanding of the dataset by allowing us to evaluate these loss functions' effectiveness in various tasks and providing a wider view on their applicability and efficiency in challenging of object detection tasks.

We train the model by following the suggested instructions in PASCAL VOC 2007 official website

(a) (b) (c)

Figure. 9 Example results from PASCAL VOC 2007 dataset trained using: (a) GIoU loss, (b) CIoU loss, and (c) Proposed GCIoU loss

Table 2. The comparison of performance with different loss functions in the YOLOv4.

Loss	AP50	AP75	AP95
IoU	70.05	24.72	0.15
GIoU	73.33	29.82	0.12
Rel. improv.%	3.28	5.1	-0.03
CIoU	76.54	35.73	0.22
Rel. improv.%	**6.49**	11.01	0.07
GCIoU	74.19	38.80	0.50
Rel. improv.%	4.14	**14.08**	**0.35**

[27], the train set containing 2,501 images along with the validation set with 2,510 images. We evaluate the trained model on the 2007 test set containing 4,952 images. For setup settings, we use batch size of 64 and 64 subdivisions to train various loss functions. 64 batch refers to the number of images processed together for one iteration, and 64 subdivisions mean the division of this batch into 64 smaller parts. From Table 2 shows that our proposed GCIoU loss achieves a comparable performance of 74.19% at AP50, whereas the CIoU loss achieves 76.54%, and the GIoU loss achieves 73.33%. For higher level of thresholds, our proposed GCIoU loss starts to perform well. Compared to IoU loss, the AP75 on GCIoU loss increases by 14.08%, making it the highest performance at AP75, CIoU loss with 11.01% improvement and then GIoU loss with 5.10% improvement. Lastly, at AP95 which uses an IoU threshold of 0.95, meaning that for an object detection prediction to be considered correct, it must have at least a 95% overlap with the ground truth, our proposed GCIoU loss offers the highest performance compared to IoU loss with 0.35% improvement followed by CIoU loss with 0.07% improvement and GIoU loss at -0.03% change.

To evaluate the precise performance of each loss evaluated on the dataset, we show more detailed comparisons, such as different threshold levels and performance on each class on the AP75. From the Table 3, our proposed GCIoU loss offers comparable performance on the lower thresholds, while performing consistently well on the higher thresholds AP 65:95. Achieving high AP at higher threshold indicates that the model is reliable and precise in its

Table 3. PASCAL VOC 2007 Performance comparison between three losses on different level of thresholds

Loss %	AP50	AP55	AP60	AP65	AP70	AP75	AP80	AP85	AP90	AP95	AP
IoU	70.05	65.71	59.27	49.26	37.27	24.72	13.18	5.09	1.64	0.15	32.63
GIoU	73.33	71.79	66.52	57.54	45.05	29.82	15.66	5.73	1.30	0.12	36.69
Rel. Improv.%	3.28	6.08	7.25	8.28	7.78	5.10	2.48	0.64	-0.34	-0.03	4.05
CIoU	76.54	73.70	68.10	60.49	50.02	35.73	20.58	8.72	1.48	0.22	39.50
Rel. Improv.%	**6.49**	**7.99**	**8.83**	11.23	12.75	11.01	7.40	3.63	-0.16	0.07	6.86
GCIoU	74.19	71.94	67.52	61.19	51.54	38.80	24.05	10.95	3.00	0.50	40.37
Rel. Improv.%	4.14	6.23	8.25	**11.93**	**14.27**	**14.08**	**10.87**	**5.86**	**1.36**	**0.35**	**7.73**

Table 4. Performance comparison for each class at IoU 0.75

Loss % (AP75)	aeroplane	bicycle	bird	boat	bottle	bus	car
IoU	24.51	12.25	18.73	6.03	21.39	32.32	29.60
GIoU	18.90	**42.80**	**37.22**	**24.22**	18.14	47.98	38.54
CIoU	17.88	27.23	32.48	23.80	22.20	66.01	53.25
Proposed GCIoU	**28.80**	39.26	37.02	18.27	**28.20**	**68.83**	**55.11**
Loss % (AP75)	cat	chair	cow	diningtable	dog	horse	motorbike
IoU	53.52	24.30	26.44	9.08	34.46	31.01	24.09
GIoU	26.12	25.99	40.80	16.80	**49.60**	19.18	26.74
CIoU	42.27	21.77	**50.07**	24.44	44.61	56.10	35.82
Proposed GCIoU	**54.26**	**27.32**	32.16	**27.56**	45.71	**56.48**	**46.98**
Loss % (AP75)	person	pottedplant	sheep	sofa	train	tvmonitor	
IoU	28.25	6.19	22.78	13.63	38.03	27.66	
GIoU	40.02	14.81	15.26	24.00	23.65	45.62	
CIoU	**41.89**	15.12	**32.70**	24.88	49.04	28.09	
Proposed GCIoU	36.11	**15.31**	26.08	**27.40**	**52.16**	**53.00**	

Figure. 10 Example results from PASCAL VOC 2007 dataset for CIoU loss (top) and GCIoU loss (bottom)

predictions. In Table 4 we show the AP for each 20 classes, and our proposed loss showing its dominance even if we set the threshold to be 75%. By maintaining high precision, especially at high level of thresholds is important because a high precision indicates that a large proportion of positive instances are detected correctly. Poor localization is often caused by noisy dataset, which fail to provide enough features to the network to process, resulting in low accuracy. Therefore, we trained the PASCAL VOC 2007 + 2012[28] dataset applied with 0.1 noise, to test each losses performance on the low image quality.

The effectiveness of each loss on a noisy dataset is provided in the Table 5 at different level of thresholds. CIoU loss takes the highest AP50 with 0.18% improvement to the IoU loss, followed by GIoU loss at 0.15% and GCIoU loss at -0.06% change. GCIoU loss then achieves its effectiveness at higher thresholds, at AP75 it outperforms other losses by achieving 0.66% improvement, then GIoU loss at 0.45%, and CIoU loss at 0.12% improvement. We

Table 5. Noisy PASCAL VOC 2007 + 2012 Performance comparison between three losses on different level of thresholds

Loss %	AP50	AP55	AP60	AP65	AP70	AP75	AP80	AP85	AP90	AP95	AP
IoU	70.38	67.56	62.71	55.13	44.09	30.94	17.96	8.51	3.82	0.95	36.20
GIoU	70.53	67.75	62.81	55.36	44.69	31.39	17.67	8.56	3.73	0.39	36.27
Rel. Improv.%	0.15	**0.09**	**0.10**	0.23	0.60	0.45	-0.29	0.05	-0.09	-0.56	0.07
CIoU	70.56	67.61	62.65	55.43	44.45	31.06	18.20	9.00	3.11	0.64	36.27
Rel. Improv.%	**0.18**	0.05	-0.06	**0.30**	0.36	0.12	0.24	**0.49**	-0.71	-0.31	0.07
GCIoU	70.32	67.61	62.63	55.35	44.72	31.60	18.53	8.69	4.16	0.97	36.45
Rel. Improv.%	-0.06	0.05	-0.08	0.22	**0.63**	**0.66**	**0.57**	0.18	**0.34**	**0.02**	**0.25**

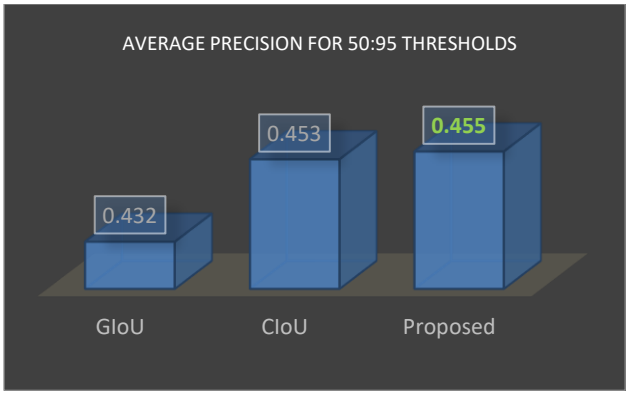


Figure 11. Average precision across different thresholds ranging from 50 to 95 with 0.5 steps

can see that at AP95, the GCIoU loss is still maintaining its robustness by outperform previous losses at 0.02% improvement, while others are fail to improve. It is proved that our proposed loss is more robust on low quality datasets, which offer highest AP at 0.25% than IoU loss as basis, followed by GIoU and CIoU losses at 0.07%.

In Fig. 11 the GCIoU loss yields the highest precision compared to other losses. This indicates that it provides better accuracy in localizing objects within an image, making the GCIoU loss can localize objects better, leading to detect more positive objects. From Fig. 10 compared to the default CIoU loss, our proposed GCIoU loss can produce better classification and localization. From all the experiments, its proved that our proposed loss is faster in convergence rate, robust on noisy dataset, and its more accurate in localizing positive objects.

5. Conclusions

In this paper, we propose a novel method of hybrid mechanism loss function called generalized complete intersection over union that combines the GIoU loss and the CIoU loss for faster and better bounding box regression. Previous losses suffer from two main limitations, slow convergence rate and inaccurate localization which lead to poor performance. To alleviate these shortcomings, our proposed GCIoU loss combines both losses in order to cover both limitations. GCIoU loss then suffer from gradient explosion while combining the two losses, to overcome this issue we use a divisor of 8 to scale down the gradient while performing backpropagation. The divisor has been proven to converge faster and more robustly in all case scenarios, outperforming the GIoU loss and CIoU loss by converging 20 fewer epochs, or an estimated 14% faster. By incorporating it into the YOLOv4 algorithm, we evaluate each losses performance on the PASCAL VOC 2007 dataset and show that GCIoU loss improved the detection performance. GCIoU loss outperforms the state-of-the-art losses by improving the AP by 7.72%, followed by CIoU loss at 6.85% and GIoU loss at 4.04% to the basis of IoU loss. GCIoU loss also generalize better because it performs consistently well at higher level APs both for clean or noisy dataset, and it localizes positive objects more accurately according to its performance. Noteworthy, that our proposed loss function only tested on the YOLOv4 and needs to be verified on other deep learning algorithms, such as Faster-RCNN and SSD.

Nomenclature

Terms	Representation
$x_1^p, y_1^p, x_2^p, y_2^p$	Properties of predicted box
$x_1^{gt}, y_1^{gt}, x_2^{gt}, y_2^{gt}$	Properties of ground truth box
X_{min} and X_{max}	Variables' lowest and maximum values
B	Area of the predicted box
b	Center coordinate of the predicted box
B^{gt}	Area of the ground truth box

b^{gt}	Center coordinate of the ground truth box
C	Area of the smallest enclosing box covering B and B^{gt}
p^2	Euclidean distance calculation
c^2	Diagonal of the box C
v	Aspect ratio calculation for both boxes
α	Trade-off parameter for v
$\frac{\partial v}{\partial w}$	Gradient calculation for width in aspect ratio consistency
$\frac{\partial v}{\partial h}$	Gradient calculation for height in aspect ratio consistency
$\frac{\partial AR}{\partial w'}$	Width changes in predicted box aspect ratio
$\frac{\partial AR}{\partial h'}$	Height changes in predicted box aspect ratio

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, Nugra and Zahir; methodology, Nugra and Zahir; software, Nugra; validation, Nugra; formal analysis, Nugra and Indrabayu; writing-original draft preparation, Nugra; dataset collection, Nugra; result analysis and comparison, Nugra; visualization Nugra; supervision, Indrabayu and Zahir; project administration, Indrabayu.

References

[1] A. N. Gajjar and J. Jethva, "Intersection over Union based analysis of Image detection/ segmentation using CNN model", In: *Proc. of Second International Conference on Power, Control and Computing Technologies (ICPC2T)*, pp. 1–6, 2022, doi: 10.1109/ICPC2T53885.2022.9776896.

[2] X. Wu, D. Sahoo, and S. C. H. Hoi, "Recent advances in deep learning for object detection", *Neurocomputing*, Vol. 396, pp. 39–64, 2020, doi: 10.1016/j.neucom.2020.01.085.

[3] W. Farsal, S. Anter, and M. Ramdani, "Deep Learning: An Overview", In: *Proc. of the 12th International Conference on Intelligent Systems: Theories and Applications, in SITA'18. New York, USA: Association for Computing Machinery*, pp. 1–6, 2018, doi: 10.1145/3289402.3289538.

[4] Z. Q. Zhao, P. Zheng, S. Xu, and X. Wu, "Object Detection with Deep Learning: A Review", *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1-21, 2019, 10.1109/TNNLS.2018.2876865.

[5] M. A. Rahman and Y. Wang, "Optimizing Intersection-Over-Union in Deep Neural Networks for Image Segmentation", In: *Proc. of Advances in Visual Computing, Springer International Publishing*, pp. 234–244, 2016, doi: 10.1007/978-3-319-50835-1_22.

[6] C. Zhang, K. Luo, F. Meng, and Q. Wu, "The Elliptic Energy Loss for Rotated Object Detection in Aerial Images", In: *Proc. of International Conference on Image Processing (ICIP)*, pp. 3384–3388, 2023, doi: 10.1109/ICIP49359.2023.10222610.

[7] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: An Advanced Object Detection Network", In: *Proc. of the 24th ACM International Conference on Multimedia*, pp. 516–520, 2016, doi: 10.1145/2964284.2967274.

[8] D. Zhou et al., "IoU Loss for 2D/3D Object Detection", In: *Proc. of International Conference on 3D Vision (3DV)*, pp. 85–94, 2019, doi: 10.1109/3DV.2019.00019.

[9] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression", *CVPR, Long Beach*, CA, USA: IEEE, pp. 658–666, 2019, doi: 10.1109/CVPR.2019.00075.

[10] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression", *AAAI*, Vol. 34, No. 07, pp. 12993–13000, 2020, doi: 10.1609/aaai.v34i07.6999.

[11] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", In: *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR*, USA: IEEE Comput. Soc, p. I-511-I–518, 2001, doi: 10.1109/CVPR.2001.990517.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", In: *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, doi: 10.1109/CVPR.2014.81.

[13] A. L. O. M. Armadi, Indrabayu, and I. Nurtanio, "Snacks Detection Under Overlapped Conditions Using Computer Vision", In: *Proc. of International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, pp. 360–364, 2023, doi: 10.1109/IAICT59002.2023.10205599.

[14] R. Girshick, "Fast R-CNN", In: *Proc. of International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015, doi:

10.1109/ICCV.2015.169.

[15] A. Darnilasari, Indrabayu, and I. S. Areni, "Implementation of Faster R-CNN with Colour and Blur Augmentation For Differentiate Cloves From Debris", In: *Proc. of 2nd International Conference on Computer System, Information Technology, and Electrical Engineering (COSITE)*, pp. 72–77, 2023, doi: 10.1109/COSITE60233.2023.10249515.

[16] A. Z. Syaharuddin, Z. Zainuddin, and Andani, "Multi-Pole Road Sign Detection Based on Faster Region-based Convolutional Neural Network (Faster R-CNN)", In: *Proc. of International Conference on Artificial Intelligence and Mechatronics Systems (AIMS)*, pp. 1–5, 2021, doi: 10.1109/AIMS52415.2021.9466014.

[17] W. Liu et al., "SSD: Single Shot MultiBox Detector", in *Computer Vision – ECCV 2016, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing*, pp. 21–37, 2016, doi: 10.1007/978-3-319-46448-0_2.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, USA: IEEE, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.

[19] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger", In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, pp. 6517–6525, 2017, doi: 10.1109/CVPR.2017.690.

[20] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement", *Cornell University: Ithaca*, NY, USA, Available Online: http://arxiv.org/abs/1804.02767, doi: 10.48550/arXiv.1804.02767, 2018.

[21] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection", *Cornell University: Ithaca*, NY, USA, Available Online: https://arxiv.org/abs/2004.10934, doi: 10.48550/arXiv.2004.10934, 2020.

[22] Z. Zheng et al., "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, 2022, doi: 10.1109/TCYB.2021.3095305.

[23] J. Wang et al., "Side-Aware Boundary Localization for More Precise Object Detection", In: *Proc. of ECCV 2020*, pp. 403–419, 2020, doi: 10.1007/978-3-030-58548-8_24.

[24] H. Li, Q. Zhou, Y. Mao, B. Zhang, and C. Liu, "Alpha-SGANet: A multi-attention-scale feature pyramid network combined with lightweight network based on Alpha-IoU loss", *PLoS One*, Vol. 17, No. 10, p. e0276581, 2022, doi: 10.1371/journal.pone.0276581.

[25] M. Everingham, L. V. Gool, C. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) challenge", *International Journal of Computer Vision*, Vol. 88, pp. 303–338, 2010, doi: 10.1007/s11263-009-0275-4.

[26] Z. Tong, Y. Chen, Z. Xu, and R. Yu, "Wise-IoU: Bounding Box Regression Loss with Dynamic Focusing Mechanism", Available Online: https://arxiv.org/abs/2301.10051.

[27] PASCAL VOC 2007 dataset Available online: http://host.robots.ox.ac.uk/pascal/VOC/voc2007.

[28] PASCAL VOC 2012 dataset Available online: http://host.robots.ox.ac.uk/pascal/VOC/voc2012.