

## Article

# Shrimp Larvae Counting Based on Improved YOLOv5 Model with Regional Segmentation

Hongchao Duan, Jun Wang, Yuan Zhang <sup>\*</sup>, Xiangyu Wu, Tao Peng, Xuhao Liu and Delong Deng

Centre for Optical and Electromagnetic Research, South China Academy of Advanced Optoelectronics, South China Normal University, Guangzhou 510006, China; 2021024069@m.scnu.edu.cn (H.D.); 2020023787@m.scnu.edu.cn (J.W.); 2022024121@m.scnu.edu.cn (X.W.); 2023024193@m.scnu.edu.cn (T.P.); 2023024174@m.scnu.edu.cn (D.D.)

<sup>\*</sup> Correspondence: yuan.zhang@coer-scnu.org

**Abstract:** Counting shrimp larvae is an essential part of shrimp farming. Due to their tiny size and high density, this task is exceedingly difficult. Thus, we introduce an algorithm for counting densely packed shrimp larvae utilizing an enhanced You Only Look Once version 5 (YOLOv5) model through a regional segmentation approach. First, the C2f and convolutional block attention modules are used to improve the capabilities of YOLOv5 in recognizing the small shrimp. Moreover, employing a regional segmentation technique can decrease the receptive field area, thereby enhancing the shrimp counter's detection performance. Finally, a strategy for stitching and deduplication is implemented to tackle the problem of double counting across various segments. The findings from the experiments indicate that the suggested algorithm surpasses several other shrimp counting techniques in terms of accuracy. Notably, for high-density shrimp larvae in large quantities, this algorithm attained an accuracy exceeding 98%.

**Keywords:** shrimp larvae counting; YOLOv5; regional segmentation; attention mechanism; repeat shrimp removal



**Citation:** Duan, H.; Wang, J.; Zhang, Y.; Wu, X.; Peng, T.; Liu, X.; Deng, D. Shrimp Larvae Counting Based on Improved YOLOv5 Model with Regional Segmentation. *Sensors* **2024**, *24*, 6328. <https://doi.org/10.3390/s24196328>

Academic Editor: Yongwha Chung

Received: 10 September 2024

Revised: 26 September 2024

Accepted: 27 September 2024

Published: 30 September 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Shrimp farming has become a vital economic sector within aquaculture, contributing significantly to the growth of the fish industry [1]. Counting shrimp larvae is an essential task in the shrimp farming process, as it assists farmers in determining the reproductive rate and accurately estimating the production potential [2]. In addition, it helps evaluate fertility, control the density of cultivation, and manage transport sales [3,4]. However, due to the tiny size, great flexibility, and dense numbers of shrimp larvae, counting them presents a significant challenge [5]. Currently, the process is predominantly carried out by hand, making it a time-consuming, labor-intensive, and somewhat imprecise task [6]. Thus, developing an innovative approach to overcome the current difficulties in counting shrimp larvae is of the utmost importance.

The progress in the counting of shrimp larvae consists primarily of manual counting [7], photoelectric detector counting [8–11], counting based on conventional image processing [12], and deep learning-based counting [13]. The first technique is highly dependent on the operator's expertise, resulting in substantial differences in the accuracy and efficiency of the counting process. In addition, the process of manual counting may harm shrimp larvae. Photoelectric devices for counting shrimp larvae work by shining light on the larvae and determining their number based on the light reflected back to the sensors [8]. Although this method significantly improves both the speed and accuracy, these counters can be affected by various environmental factors, such as lighting conditions and water quality, resulting in considerable counting errors. Furthermore, photoelectric devices are generally effective for shrimp larvae of certain sizes, which reduces their accuracy when dealing with larvae of different sizes.

Recently, computer vision technology has been applied in various fields [14], such as target detection, image super-resolution, and population counting [15]. Regarding the counting of shrimp larvae [16–19], traditional image processing techniques primarily utilize image segmentation and object detection methods to recognize and count target images. Kesvarakul et al. counted shrimp larvae in images by converting them into binary images with a threshold [20]. However, this technique is only effective when dealing with a small quantity of larvae and when their images are clear. Thai et al. used image segmentation and contour tracking methods for the counting of shrimp larvae [21]. This method is able to count individual shrimp larvae, although it struggles to accurately distinguish shrimp larvae which are stuck together. For populations with fewer shrimp larvae, counting was performed using algorithms, such as segmentation paired with the Canny edge detection method and blob processing approaches [22]. Although this technique offers an enhancement over previous algorithms, it still does not address the issue of shrimp and larvae adhering to each other. Traditional image processing-based counting techniques require prior knowledge to manually adjust image features, leading to poor accuracy in scenarios with complex backgrounds. Furthermore, the lack of generalization in these models hinders the accurate detection and counting of shrimp larvae in different scenarios.

Subsequently, deep learning-based [23–26] counting methods use trained object recognition models to identify targets within images. Armalivia et al. used the You Only Look Once version 3 (YOLOv3) algorithm to count shrimp larvae [27], achieving an average counting accuracy of 96% in low-density populations of around 100 larvae (in a circular area with a diameter of 40 cm), but a larger number of larvae could not be identified. Hu et al. used deep learning and a density map of shrimp larvae to estimate the populations of approximately 1000 larvae (in a 24 cm × 33 cm area) [13]. However, this approach does not accurately identify each individual larva. Hence, devising an automated and accurate method for counting shrimp larvae in densely populated areas remains a difficult challenge. Object detection algorithms based on deep learning mainly include Regions with Convolutional Neural Networks (R-CNN) and multiple versions of YOLO, among which YOLOv5 is a mature algorithm among target recognition algorithms which has the advantages of being lightweight and having a fast inference ability. However, in the detection and recognition of small targets, YOLOv5 is not effective, and thus we need to improve YOLOv5.

In this study, we investigate the enhancement of deep learning techniques to address the challenge of counting shrimp larvae in highly populated environments. In particular, we propose an algorithm which uses regional segmentation along with an improved YOLOv5 model to accurately count shrimp larvae, especially in highly dense environments. The remainder of this paper is structured as follows. In Section 2, we give the related work to highlight our motivation. In Section 3, we describe the details of the proposed shrimp larvae counting method. Section 4 presents the experimental results and further validates the superiority of the algorithm through controlled experiments. Section 5 gives our conclusions.

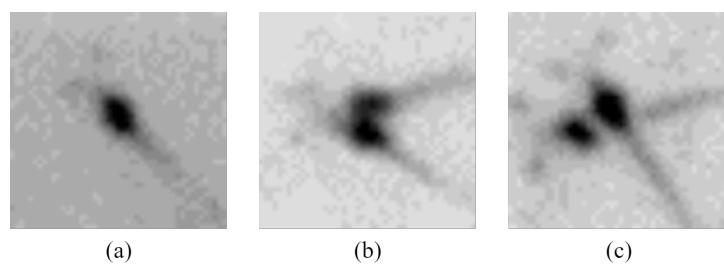
The main contributions are described as follows:

- We build an automatic shrimp collecting platform to obtain the corresponding dataset and provide a counting algorithm based on improved YOLOv5 with region segmentation.
- The C2f and attention mechanism are embedded in the YOLOv5 model, improving the detection ability for small shrimp.
- The segmentation of regions aims to boost the proportion of pixels representing shrimp larvae in a visual image.
- Experimental results prove that the proposed algorithm can identify shrimp larvae with high accuracy under the difficult circumstance of overlap between large numbers of shrimp and different light intensities.

## 2. Related Work

There are many challenges in the counting of shrimp larvae. First, shrimp larvae are small in size, making them relatively difficult to detect. In addition, shrimp larvae move rapidly, and their pixel values in images differ depending on their depth in the water. Conventional threshold segmentation techniques may easily overlook shrimp larvae which are at the bottom of the water.

As shown in the Figure 1, there are three states, including isolated, clumped, and overlapping shrimp larvae. Isolated shrimp larvae remain unaffected by the presence of other larvae and can be detected through conventional image processing approaches or deep learning methods. Clumped shrimp larvae consist of several larvae attached to each other, resulting in an underestimated count when using connected component analysis. This problem can be resolved through the application of watershed algorithms and distance transformation. However, these techniques necessitate precise threshold configurations and do not generalize well. Overlapping shrimp larvae appear when larvae at varying water depths align at the same spot in the image. The aforementioned watershed algorithm is ineffective in addressing this scenario. Utilizing concave point algorithms for separation might erroneously fragment the shrimp tails, causing overcounting of the shrimp. Consequently, conventional image processing techniques prove inadequate for identifying these instances, necessitating the use of deep learning approaches to accurately discern and identify the various states of shrimp larvae.



**Figure 1.** The states of shrimp larvae. (a) Isolated shrimp larvae. (b) Clumped shrimp larvae. (c) Overlapping shrimp larvae.

Minimizing the dimensions of the input image can enhance the accuracy of receptive field detection [28]. Two widely used techniques for decreasing the size of an image are image scaling and cropping. Image scaling often employs downsampling [29], which may lead to data loss. Another approach is image segmentation, which can be performed either randomly or in a consistent manner. Random segmentation employs random functions to determine the coordinates for the top-left and bottom-left sections of the local image of shrimp larvae [30]. Then, the coordinates for the top-right and bottom-right sections are determined according to the dimensions of the local image. In addition, the cropping function is applied to extract the local image. Beginning in the upper-left corner of the original image, fixed segmentation continues with sequential segmentation based on the dimensions of the local image. For regions along the edges which are smaller than the segmented dimensions, the true size of the image is maintained. Random segmentation might result in repeated recognition of the same regions, thereby increasing the training burden on the model. Consequently, this research uses fixed segmentation to handle the initial images.

The estimation of density maps is also a commonly used counting method [23] which estimates the number of objects by generating a density map. This is often applied in crowd counting [31]. However, this method cannot accurately provide the geometric and positional information of objects. When the detection of small objects does not require positional information, the method can provide good counting results. In shrimp larvae counting, when the larvae density is relatively low, positional and shape information may not be critical, and both density map estimation and object detection can achieve accurate

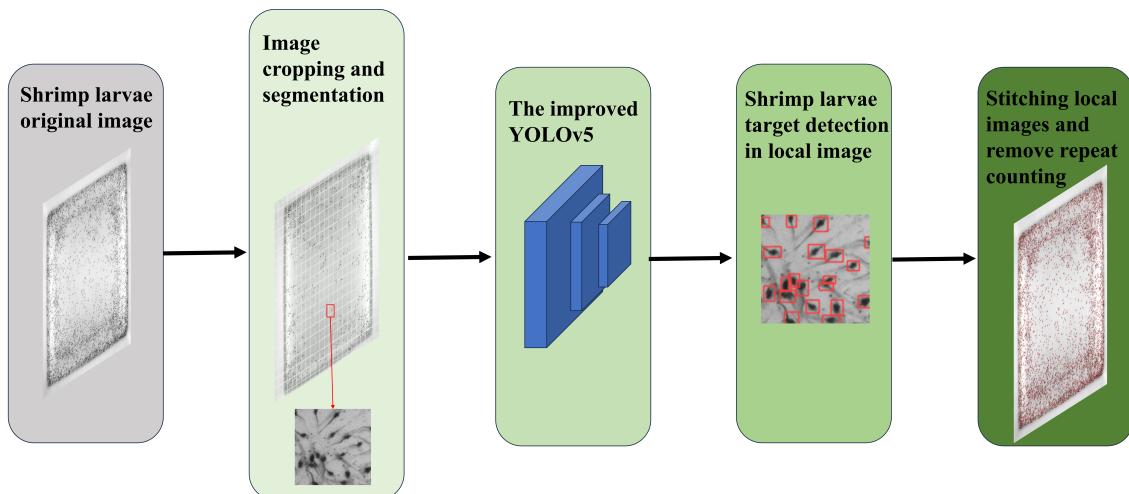
counts. However, as the larvae density increases, with overlapping larvae at different water depths, positional and shape information become important. In high-density areas, density map estimation is prone to misjudgments, leading to a decrease in count accuracy. By incorporating position and shape information, object detection methods can achieve better counting accuracy. In terms of runtime, density map estimation models are more complex and require a longer processing time, while object detection methods are faster, making them more suitable for quick shrimp larvae counting.

YOLO [32] is considered one of the best choices for object detection models due to its speed and superior accuracy compared with R-CNN [33]. YOLO pioneered an innovative approach by utilizing convolutional neural networks on a complete image to predict object classes and bounding boxes via direct feature regression. Over the years, YOLO has evolved into several versions. YOLOv1 uses a unified detection approach for object localization and classification tasks [32]. Subsequently, YOLOv2 improves on version 1 by having better accuracy, a faster speed, and the ability to recognize more objects [34]. YOLOv3 improves the object detection speed by implementing multiscale prediction [35], optimizing the core network, and refining the loss function. YOLOv4 introduces a fast and effective object detection model which substantially cuts down computational expenses, making it more compatible with general-purpose devices and those with hardware constraints. YOLOv5 brings significant advancements, including the CSPDarknet backbone and mosaic augmentation, balancing speed and accuracy [36–42]. YOLOv7 improves the structure of the extended efficient layer aggregation network, revises the model architecture, and introduces an efficient label assignment strategy, which increases detection performance while decreasing the number of model parameters [43]. Currently, researchers are constantly exploring the architectural design of YOLO, with the latest version being YOLOv10 [44]. It has achieved state-of-the-art performance and efficiency on various model scales. In the experimental section, we describe the performance of YOLOv10 and conduct shrimp larvae counting tests using this model. However, the YOLOv10 model is complex and has high hardware resource requirements, making its hardware deployment challenging. One of the primary objectives of this study is to deploy the algorithm on hardware to achieve an integrated system for shrimp larvae image acquisition and counting. YOLOv10 does not meet the requirements of our design. YOLOv5 stands out from its predecessors with its lighter architecture, faster inference times, and more developed ecosystem which offers enhanced compatibility [45]. Moreover, YOLOv5 has low hardware resource requirements, is relatively mature in terms of hardware deployment, and is easier to implement. Therefore, we selected YOLOv5 as the baseline for our study. However, in scenarios which involve the detection of small objects, such as shrimp larvae, the use of several convolutional layers may lead to missing small targets, resulting in reduced pixel occupancy for small objects. To address this problem, we upgraded the YOLOv5 backbone network to improve its ability to identify small targets.

### 3. Materials and Methods

For a more precise counting of shrimp larvae in situations with large numbers and a high density, we suggest an algorithm based primarily on regional segmentation and an improved YOLOv5 model. The flow chart of the algorithm can be viewed in Figure 2.

To begin with, we divide the whole original image into smaller segments using a region-based segmentation algorithm. Subsequently, we employ an enhanced YOLOv5 model to recognize shrimp larvae in each of these isolated images separately. Then, all segmented image blocks are reassembled into a complete image. A repeat count removal method is proposed to address the issue of repeat counting in the stitching position, ensuring that accurate counting of shrimp larvae can be achieved throughout the image.

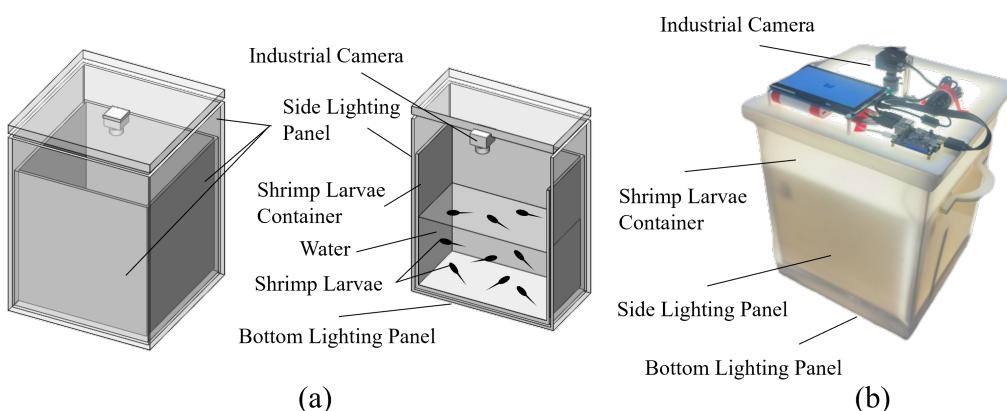


**Figure 2.** The flow chart of the proposed counting algorithm.

### 3.1. Image Collection

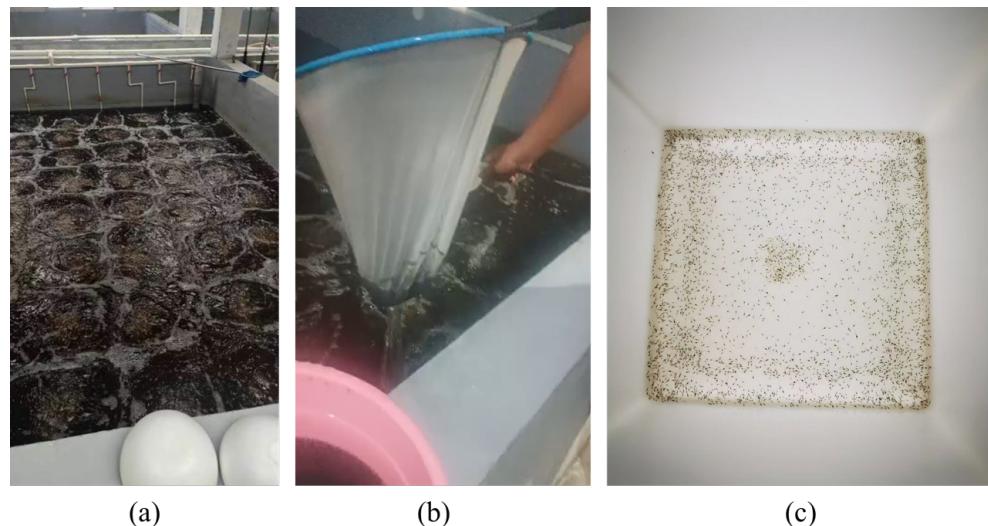
The quality of the original images, which is one of the key factors for computer vision, will affect the accuracy of the counting of shrimp larvae. For the best accuracy in counting the larvae, we designed and fabricated a shrimp larvae image acquisition device as shown in Figure 3.

Shrimp larvae are placed in a semi-transparent plastic square bucket, and white LEDs with light guide plates are added around the side wall and the bottom to provide uniform illumination. Shrimp larvae images are captured using an industrial camera (MV-CE060-10UM (Hikvision, Hangzhou, China)) with an 8 mm lens under supplementary white light. The resolution of the camera is  $3072 \times 2048$  pixels for a field of view of about  $40 \text{ cm} \times 30 \text{ cm}$ , and the effective area we used for the shrimp larvae was  $30 \text{ cm} \times 30 \text{ cm}$ .



**Figure 3.** Shrimp larvae acquisition device. (a) Schematic diagram. (b) Photographs of the actual device.

The shrimp larvae dataset used in this experiment was collected from on-site photography at the Hongkai Shrimp Larvae Farm in Zhuhai City, Guangdong Province, China. Figure 4a shows the real shrimp larvae farming environment. The staff used a fishing net to extract shrimp larvae from the pool (as shown in Figure 4b) and then placed the collected larvae in our custom-built shrimp larvae image acquisition device as shown in Figure 4c.



**Figure 4.** Acquisition of shrimp larvae datasets in a real aquaculture scenario. (a) Shrimp culture tank. (b) Shrimp larvae collection. (c) Acquisition device.

### 3.2. Region Segmentation

The receptive field refers to the size of the region in the input image which corresponds to each pixel in the output feature map [46]; that is, a single point on the feature map corresponds to a specific region on the input image. The size of the area of the receptive field directly affects the detection accuracy [47,48]. The formula for calculating the receptive field is as follows:

$$R_i = (R_{i+1} - 1) \times St_i + Ks_i \quad (1)$$

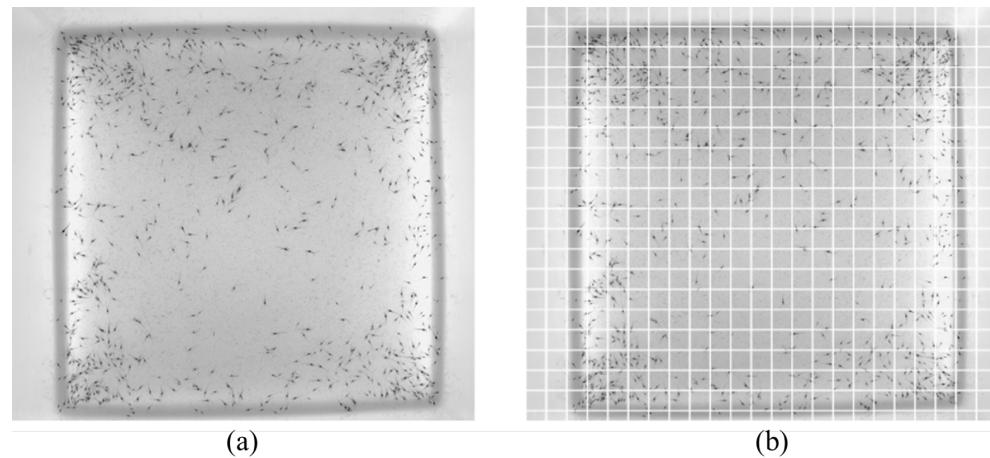
where  $R_i$  represents the receptive field on the  $i$ th convolutional layer,  $R_{i+1}$  is the receptive field on the  $i + 1$ th layer,  $St_i$  is the stride of the convolution, and  $K_S$  is the size of the convolutional kernel for the current layer.

When the original image of shrimp larvae ( $3072 \times 2048$  pixels) is used as the input image, the corresponding receptive field size is  $38 \times 38$  pixels. This indicates that each point on the feature map corresponds to a area of  $38 \times 38$  pixels on the original image, which is significantly larger than the size of the shrimp larvae (approximately  $10 \times 10$  pixels). As a result, information on shrimp larvae would be overlooked or covered, greatly reducing the detection accuracy.

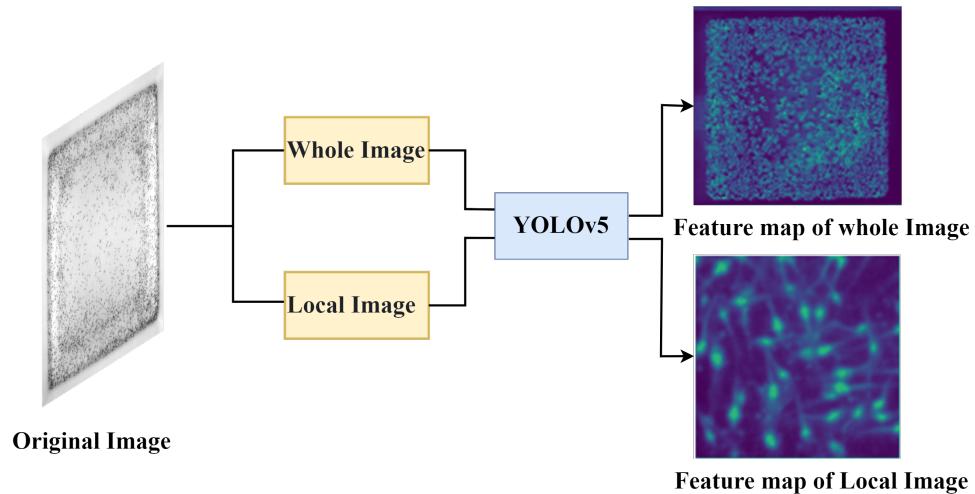
To address this issue, it is necessary to reduce the size of the receptive field to improve the shrimp larvae detection performance. Therefore, we adopted the region segmentation method to divide the original image, reducing the size of the input image and thereby decreasing the receptive field area. For example, the original image of shrimp larvae (as shown in Figure 5a) was divided into multiple image blocks ( $100 \times 100$  pixels for each, as shown in Figure 5b) for subsequent labeling and training.

In the whole image, the proportion of shrimp larvae in terms of pixels in the whole image is extremely small. Therefore, we used segmentation technology to obtain multiple local regions of shrimp larvae, making feature extraction easier.

Figure 6 shows the feature maps selected from the 17th layer of the YOLOv5 model for both the full image and the local image. To facilitate comparison, we extracted the same region from the full image as that in the local image. From Figure 6, it is evident that the full image did not accurately identify local regions, while the local image allowed a more precise extraction of features from the shrimp larvae.



**Figure 5.** Image segmentation. (a) Original shrimp larvae image. (b) Segmented image.

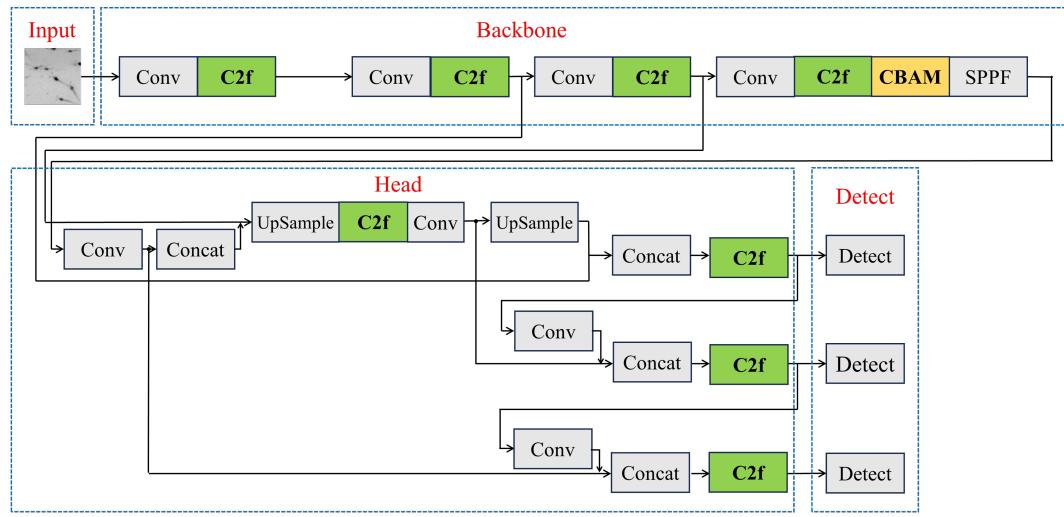


**Figure 6.** The feature map referring to the shrimp larvae image.

### 3.3. The Improved YOLOv5

The core idea of YOLOv5 is to convert the target detection problem into a regression problem, predicting the boundary boundaries and categories of the target in the image. It has high accuracy and a fast inference ability, making it one of the best-performing target detection models available today. However, the tiny size of the shrimp larvae, along with various body positions such as their overlap, leads to a decrease in the identification accuracy of the YOLOv5 model. Therefore, the recognition accuracy can be improved by increasing the classification performance of features.

As shown in Figure 7, the C2f module (Figure 8) was used to replace the C3 module. In the backbone network of the YOLOv5 algorithm, the main function of the C3 module is to extract image features and enhance the learning capacity of the convolutional network. The C3 module cannot meet the requirements for the detection of small targets such as shrimp larvae and needs further improvement to enhance the feature extraction capabilities of the model. The C2f module processes the input data using two convolutional layers, which assist in extracting features at varying levels and degrees of abstraction. This improves the feature extraction efficiency and simultaneously reduces the network weight, facilitating more abundant gradient flow information [49]. In contrast to the C3 module, the C2f module is more lightweight, has reduced computational demands, and demonstrates robust feature extraction capabilities.

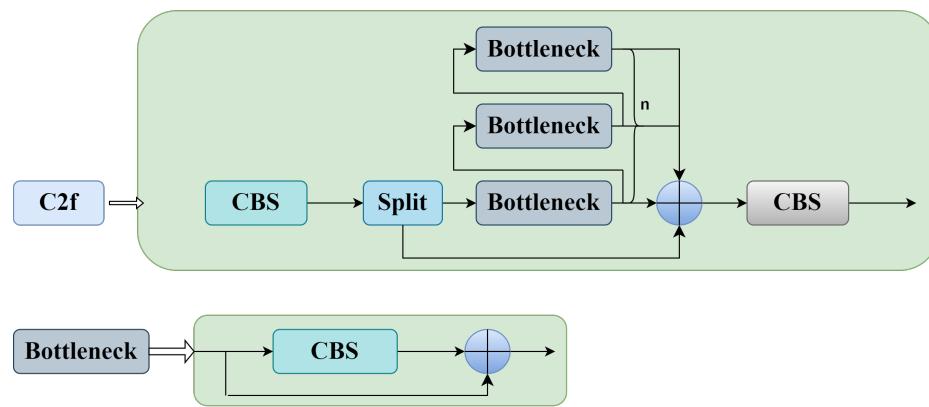


**C2f:** Faster implementation of CSP bottleneck with 2 convolutions

**C3:** CSP bottleneck with 3 convolutions

**CSP:** Cross Stage Partial

**Figure 7.** The structure of the improved YOLOv5 model.

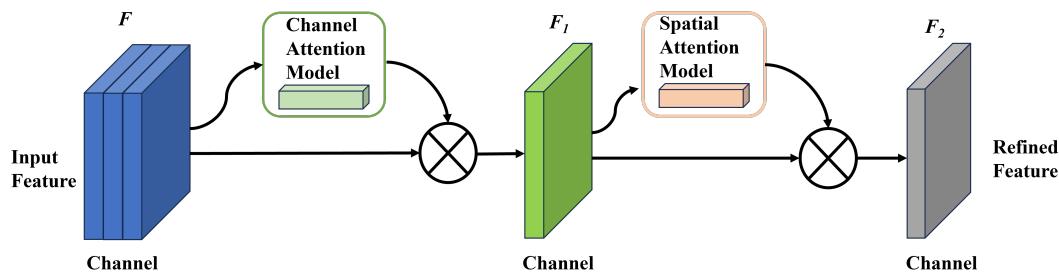


**CBS:** consist of convolution, normalization and silu activation function

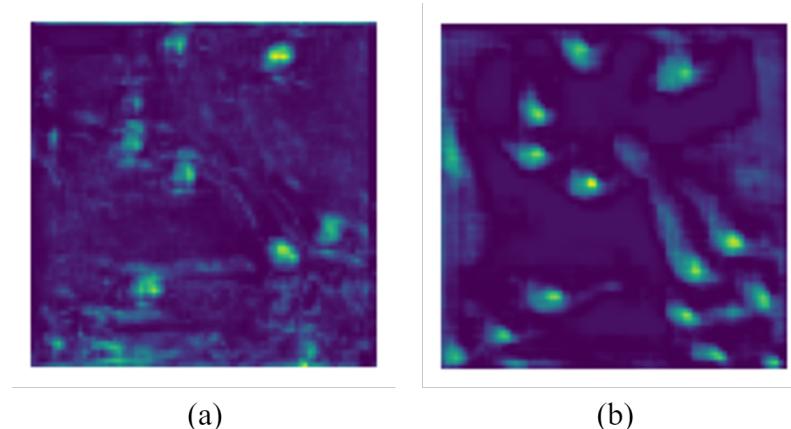
**Figure 8.** Structure of the C2f network.

To further enhance the recognition capability of the model, the convolutional block attention module (CBAM) [50] is introduced into the YOLOv5 framework. The CBAM consists of the channel attention and spatial attention modules, as shown in Figure 9. For an input  $F$ , the global average and maximum pooling operations are first applied to obtain global information for each channel. Subsequently, two fully connected layers are utilized to produce the channel attention vector. These layers apply weights to the input feature  $F$  per channel, which yields the refined feature  $F_1$ . In the spatial attention module, the feature map is combined to capture spatial information. Furthermore, the feature  $F_1$  undergoes a convolution layer  $3 \times 3$ , resulting in the creation of a spatial attention map via the sigmoid function. The spatial attention map is subsequently used to enhance the feature  $F_1$ , leading to production of the fused feature  $F_2$ . The attention module improves the depiction of shrimp larvae features across different conditions.

Figure 10 displays the feature map of both the enhanced YOLOv5 network and the original YOLOv5 network. Clearly, the proposed approach provides more precise counting outcomes even in instances of densely packed shrimp larvae.



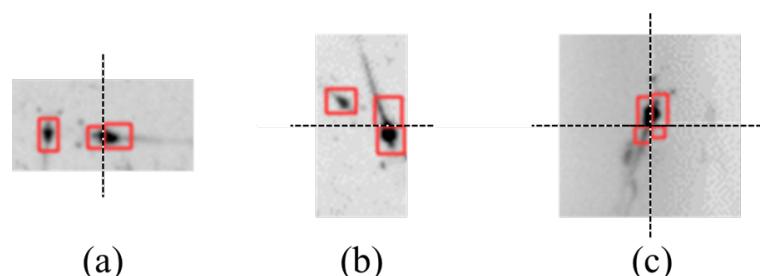
**Figure 9.** Structure illustration of CBAM.



**Figure 10.** Feature map comparison. (a) The original YOLOv5 network. (b) The improved YOLOv5 network.

### 3.4. Repeat Count Removal via Stitching

The method described in Section 3.2 allows for segmentation of the entire image. The method described in Section 3.3 can identify shrimp larvae in each segmented image block. However, identical shrimp larvae can be segmented into neighboring image blocks and detected at the same time (in Figure 11), resulting in reduced counting accuracy. To address the issue of duplicate counts, an additional detection model was developed.

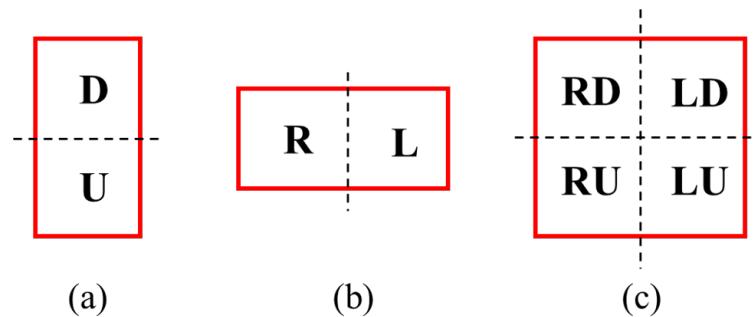


**Figure 11.** Segmentation of same shrimp larvae in adjacent image blocks (dashed lines indicate the stitching positions). (a) Vertical segmentation. (b) Horizontal segmentation. (c) Both horizontal and vertical segmentation.

Figure 11a shows a shrimp larva divided vertically into two sections and identified in two separate subimages. Figure 11b presents two shrimp larvae segmented horizontally into two subimages and identified. Figure 11c shows shrimp larvae cut horizontally and vertically, resulting in four distinct subimages. Figure 12 shows a schematic illustration of the segmentation of shrimp larvae, helping in the evaluation of the segmentation configurations.

In Figure 12, the red boxes represent the detection boundary boxes of the detection output, and the black lines represent the cropping boundaries of the image. Taking into

account the spatial relationship between the detection boundary boxes and the image cropping boundaries in the output, nine distinct scenarios can be identified: undivided shrimp larvae ( $N$ ), top edge touching ( $U$ ), bottom edge touching ( $D$ ), right edge touching ( $R$ ), left edge touching ( $L$ ), right and bottom edges touching ( $RD$ ), right and top edges touching ( $RU$ ), left and bottom edges touching ( $LD$ ), and left and top edges touching ( $LU$ ). A set  $S$ , where  $S = \{N, D, U, R, L, RD, RU, LD, LU\}$ , was defined to store all detection output. By traversing the detection outputs and checking the boundaries, the detection results were placed in the corresponding sets.

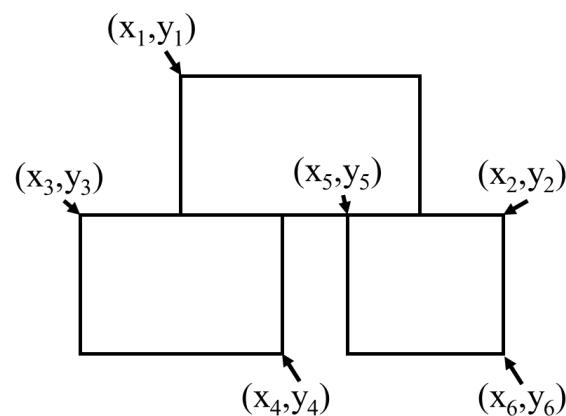


**Figure 12.** Schematic diagram of shrimp larvae segmentation. (a) Vertical segmentation. (b) Horizontal segmentation. (c) Both horizontal and vertical segmentation.

To avoid duplication, we examined the shrimp larva detection boxes in neighboring regions to determine whether the detected shrimp larvae were identical. If several detection boxes corresponded to each other, as illustrated in Figure 13, then  $(x_i, y_i)$  are the coordinates of the vertices of the detection frame, and the detection box at the bottom which intersects the cropping boundary coincides with the two detection boxes at the top, which also intersect with the cropping boundary. Therefore, an overlap ratio must be implemented to facilitate the selection process. We define a variable  $R_o$  to represent the percentage of overlap between detection frames, whose formula is as follows:

$$R_o = \frac{x_4 - x_1}{x_2 - x_3} \quad (2)$$

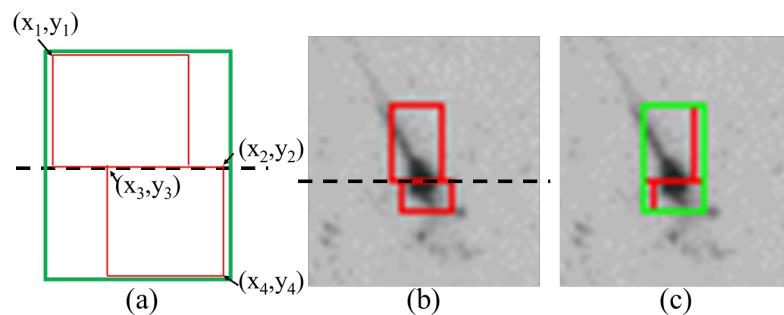
where  $x_1, x_2, x_3, x_4$  are the horizontal coordinates of the detection frame.



**Figure 13.** Multiple detection box matching.

By calculating the overlap ratio, the detection box with the highest overlap ratio is selected. After determining the matching object, a minimum bounding rectangle is used to completely enclose the two detection boxes, thus identifying the deduplication area (indicated in Figure 14a, for example). The two red detection boxes in Figure 14b

correspond to a typical repeat count case (according to Figure 11b). We determined whether they were connected by examining the coordinates of the two detection boxes. Once the connected boxes were identified, we used a minimum enclosing box (indicated by the green rectangle in Figure 14a,c) to completely cover the two red detection boxes and define the deduplication area.



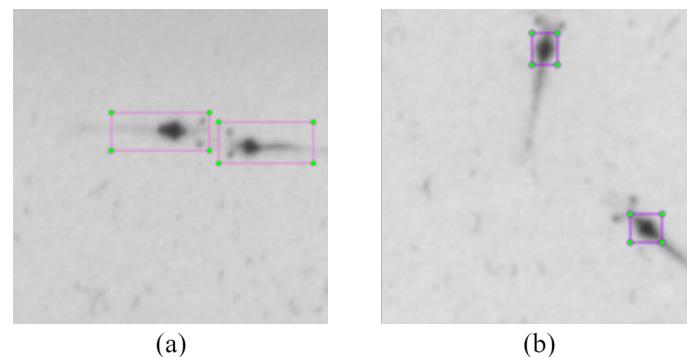
**Figure 14.** The deduplication area. (a) Schematic diagram of how to find the deduplication area (the dashed line indicates region segmentation positions), where the two red rectangles are the detection boxes and the corresponding minimum enclosing rectangle is drawn in green. (b) Original detection result with two detection boxes detecting the same larva. (c) Minimum enclosing rectangle for the case in (b).

Once the deduplication area is identified, it is fed back into the enhanced YOLOv5 network mentioned in Section 3.3 to accurately count the shrimp larvae in this region. This process ensures that duplicate counts are removed from the entire composite image of shrimp larvae.

#### 4. Experimental Results and Analysis

##### 4.1. Implementation Details

In the experiment, we collected 20,000 valid local images of shrimp larvae, with 15,000 allocated for training and 5000 for testing. For annotating the dataset, the conventional labeling approach (Figure 15a) involves describing all the shrimp larvae, yet this can introduce redundant details and impede the model's ability to learn the features of shrimp larvae. We marked the heads of the shrimp larvae as shown in Figure 15b. This annotation method concentrates on the feature of the shrimp larvae, making it easier to distinguish overlapping larvae. The comparison algorithms included the YOLOv5 algorithm (which means using YOLOv5 without regional segmentation), the Mask R-CNN algorithm [17], and popular commercial software (Smart Shrimp Farm V2.1.3 [51]). In addition, the proposed algorithm was performed on an NVIDIA RTX 3080Ti (NVIDIA, Santa Clara, CA, USA). A total of 500 epochs were trained, with the learning rate set to 0.001. The other parameters were the default YOLOv5 parameters. The optimizer was the stochastic gradient descent.



**Figure 15.** Dataset annotation. (a) The traditional labeling method. (b) The annotation method we used.

We used images of shrimp larvae of different densities to evaluate the performance of the proposed method. Under the fixed field of view provided (30 cm × 30 cm), we examined four different densities of shrimp larvae: approximately 1000 larvae for density level 1, approximately 2300 larvae for density level 2, approximately 4000 larvae for density level 3, and roughly 5000 larvae for density level 4. We chose 100 × 100 pixels for the segmentation image size in the counting experiments hereafter. The dataset and codes can be requested by email (Hongchao Duan: 2021024069@m.scnu.edu.cn).

#### 4.2. Evaluation Criterion

To quantitatively evaluate the performance of the proposed algorithm, we define  $A$  as the measurement for the counting accuracy of the formula as follows:

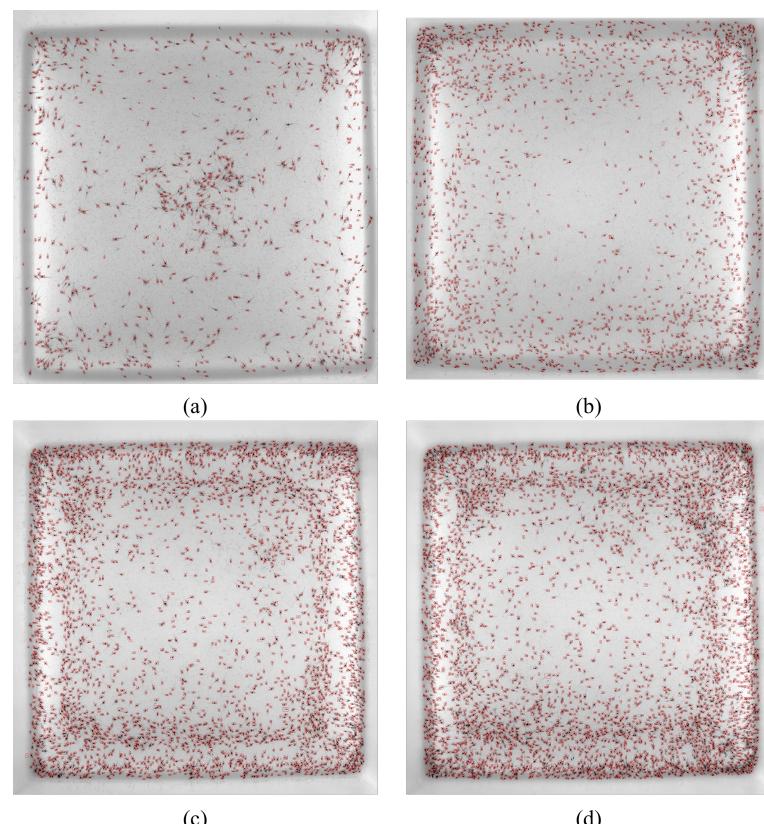
$$A = \frac{A_c - |I_d - A_c|}{A_c} \times 100\% \quad (3)$$

where  $A_c$  represents the actual number of shrimp larvae and  $I_d$  denotes the number of shrimp larvae.

For accurate results, 15 images of shrimp larvae were analyzed at each density level, with the mean count accuracy serving as the statistical result for each level.

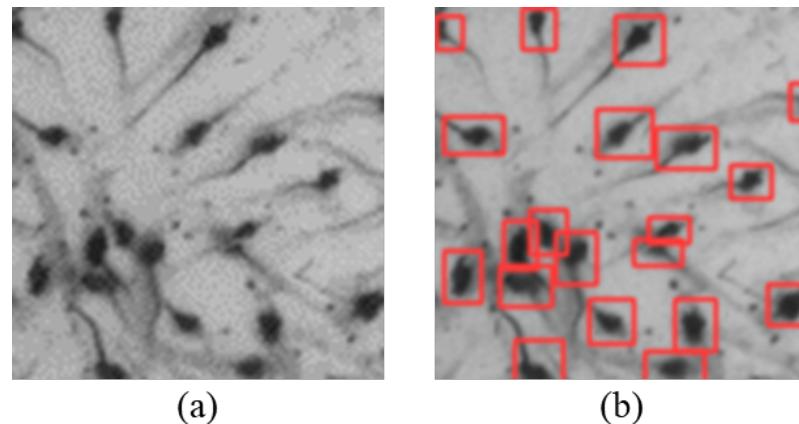
#### 4.3. Experiment Comparison

To verify the effectiveness of the above proposed method, we collected a large number of shrimp images of different densities (four densities mentioned in Section 4.1) for the validation experiment. Figure 16a–d shows the recognition effect of the proposed method under different densities. It can be seen in Figure 16 that the proposed algorithm had a good recognition effect for different densities, especially in places where shrimp larvae were densely populated.



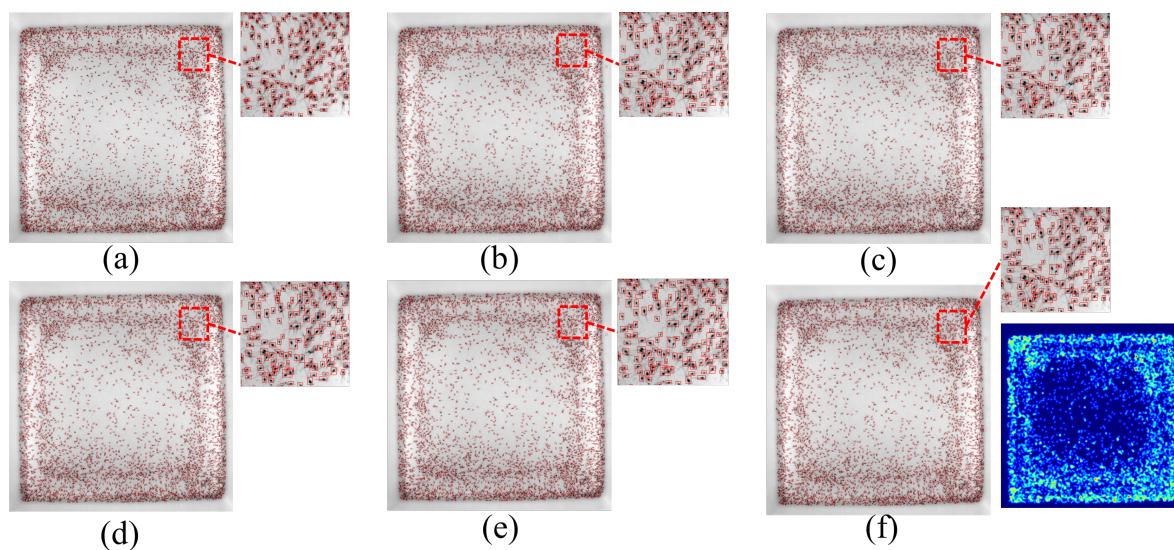
**Figure 16.** Counting results at different densities. (a) Density 1. (b) Density 2. (c) Density 3. (d) Density 4.

Figure 17a shows a typical local image of the original high-density shrimp larvae (approximately 5000 shrimp larvae in a  $30\text{ cm} \times 30\text{ cm}$  area), and Figure 17b shows the corresponding identification results. It shows that the proposed method exhibited high recognition accuracy even when the density of the shrimp larvae was high and there were overlaps between larvae. This indicates that the regional segmentation approach adopted in this study makes the receptive field more accurate, which makes the improved YOLOv5 model more accurate in the recognition of shrimp larvae targets.



**Figure 17.** A typical local area of Figure 16. (a) Original image. (b) Detection results.

Among the various comparison algorithms, Smart Shrimp Farm only provided the final count results without the detection images. We tested high-density shrimp larvae images (approximately 5000 shrimp larvae in a  $30\text{ cm} \times 30\text{ cm}$  area) using the remaining algorithms, and the comparison of the test results is shown in Figure 18. For Density Map Regression, we provide the corresponding density map in Figure 18f.



**Figure 18.** Test result images for different algorithms. (a) The proposed method. (b) YOLOv5. (c) Mask R-CNN. (d) YOLOv10. (e) Shrimpseed\_Net. (f) Density map regression.

Table 1 presents the statistical results of the average counting accuracy  $A$  for the comparison algorithm.

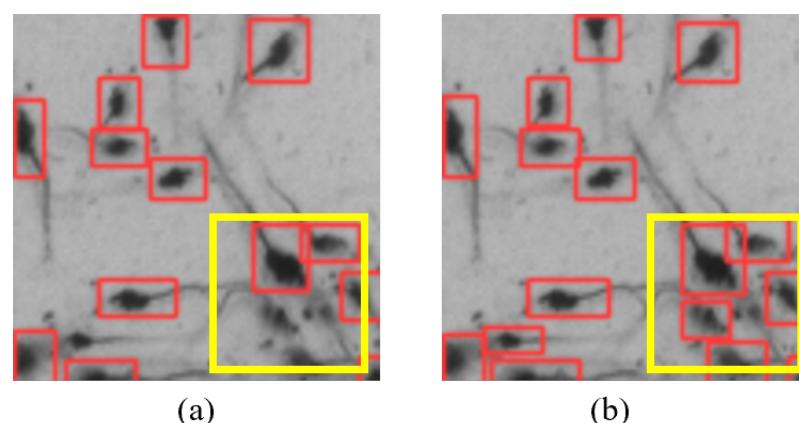
**Table 1.** Average counting accuracy  $A$  for different shrimp larvae densities.

Method	Density 1	Density 2	Density 3	Density 4
<b>The Proposed Method</b>	<b>99.32%</b>	<b>99.17%</b>	<b>98.53%</b>	<b>98.23%</b>
Smart Shrimp Farm [51]	91.43%	90.35%	88.25%	82.99%
YOLOv5 [52]	87.79%	85.46%	83.66%	81.03%
Mask R-CNN [17]	86.63%	84.52%	81.25%	80.32%
YOLOv10 [44]	91.41%	90.68%	89.29%	88.75%
Shrimpseed_Net [53]	91.65%	90.23%	89.47%	87.62%
<b>Density Map Regression [23]</b>	<b>92.33%</b>	<b>91.58%</b>	<b>90.75%</b>	<b>88.61%</b>

The results clearly indicate that, compared with commercial software such as YOLOv5, YOLOv10, Shrimpseed\_Net, Density Map Regression, and Mask R-CNN, the proposed algorithm excelled in counting shrimp larvae at various densities, particularly in high-density scenarios. Although the accuracy decreased somewhat as the density increased, the overall recognition accuracy of the proposed method remained above 98%, significantly outperforming the other three cases (just above 80%). Moreover, the accuracy, which exceeded 98%, demonstrates that the proposed algorithm could potentially substitute the manual shrimp counting process on farms.

#### 4.4. Ablation Study

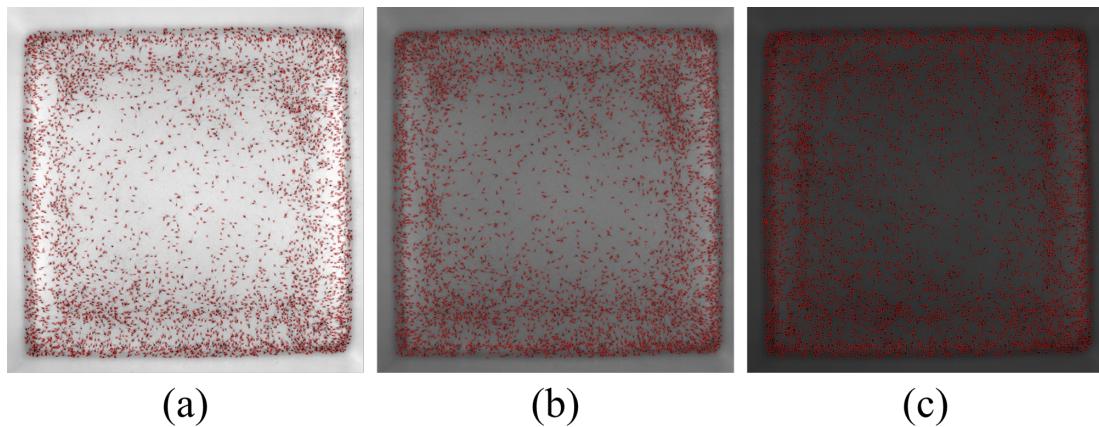
To validate the superiority of the improved YOLOv5 model proposed in this article, we compared the identification results in the segmentation images of  $100 \times 100$  pixels using both the original YOLOv5 model and the improved YOLOv5 model in Section 3.3 (typical results are shown in Figure 19).



**Figure 19.** Recognition performance of different algorithm models. (a) Original YOLOv5 model. (b) Improved YOLOv5 model.

As shown in Figure 19, the improved YOLOv5 model demonstrated better recognition performance when the shrimp larvae were densely packed. To quantitatively analyze the differences between the two models, we used the counting algorithm on all images, and the results of the average counting accuracy for the four density levels are summarized in Table 2. For density 4, we collected images of shrimp larvae under three different lighting conditions (bright, normal brightness, and dim) for testing. The test results are shown in Figure 20, and the data are summarized in Table 2.

The statistical results in Table 2 show that the improved YOLOv5 model achieved an approximately 2% higher counting accuracy compared with the original YOLOv5 model, validating the effectiveness of the proposed method. Under different lighting conditions, the counting accuracy of our proposed algorithm exceeded 98%, outperforming the original YOLOv5 algorithm.

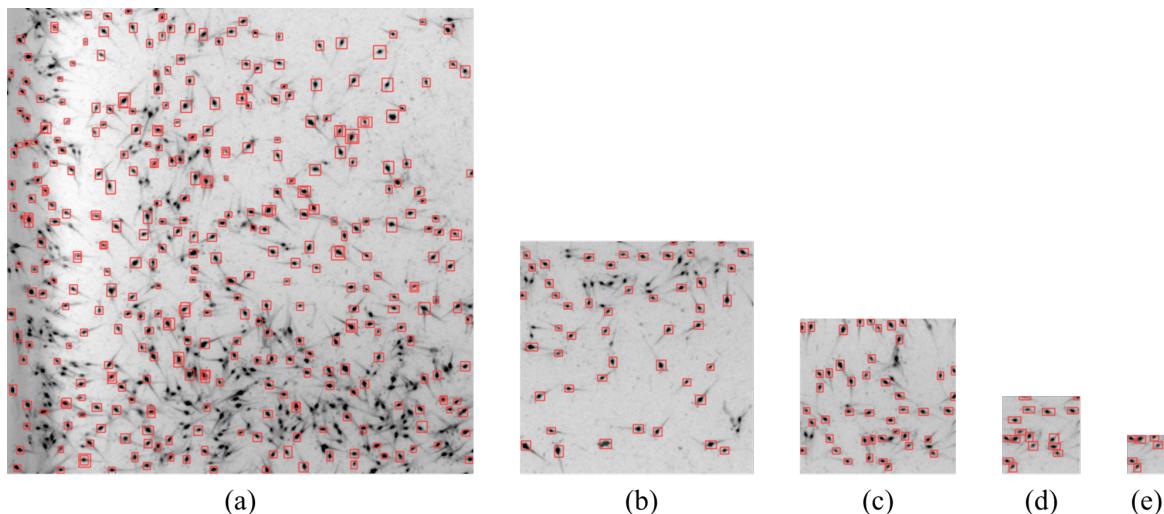


**Figure 20.** Test results under three different lighting conditions. (a) Bright. (b) Normal brightness. (c) Dim.

**Table 2.** Average counting accuracy  $A$  for the two YOLOv5 models.

Density	The Original YOLOv5	The Improved YOLOv5
Density 1	98.13%	99.32%
Density 2	97.85%	99.17%
Density 3	96.81%	98.53%
Density 4 (Bright)	96.26%	98.23%
Density 4 (Normal Brightness)	94.66%	98.21%
Density 4 (Dim)	93.52%	98.17%

The original image of the larvae was divided into many local areas to decrease the size of the receptive field in Section 3.2. To evaluate the impact of segmentation image sizes on the accuracy of shrimp larvae detection, we cropped the original shrimp larvae images into five different sizes:  $600 \times 600$  pixels,  $300 \times 300$  pixels,  $200 \times 200$  pixels,  $100 \times 100$  pixels, and  $50 \times 50$  pixels (as shown in Figure 21). Subsequently, we used the improved YOLOv5 model to train and test these cropped images of different sizes separately.



**Figure 21.** Recognition results for different segmentation image sizes: (a)  $600 \times 600$  pixels, (b)  $300 \times 300$  pixels, (c)  $200 \times 200$  pixels, (d)  $100 \times 100$  pixels, and (e)  $50 \times 50$  pixels.

As shown in Figure 21, when the segmentation image size was  $600 \times 600$  pixels, many shrimp larvae were not identified. As the segmentation size decreased, the number of undetected shrimp larvae also decreased. When the segmentation image size was reduced

to  $100 \times 100$  pixels and  $50 \times 50$  pixels, all shrimp larvae in the local images were detected. To quantitatively analyze the differences between the various segmentation image sizes, we tested the counting accuracy using images with four different densities of shrimp larvae mentioned in Section 4.1, and the statistical results of the average counting accuracy  $A$  for each level of density of shrimp larvae are presented in Table 3.

**Table 3.** Average counting accuracy  $A$  for different segmentation sizes.

Density	600 × 600 Pixels	300 × 300 Pixels	200 × 200 Pixels	100 × 100 Pixels	50 × 50 Pixels
Density 1	91.69%	94.27%	97.79%	<b>99.32%</b>	99.34%
Density 2	90.83%	93.85%	97.24%	<b>99.17%</b>	99.20%
Density 3	88.72%	93.31%	96.38%	<b>98.53%</b>	98.87%
Density 4	88.15%	92.56%	95.93%	<b>98.23%</b>	98.21%

The results indicate that the size of the image segmentation had a significant impact on the accuracy of the counting of shrimp larvae. As the segmentation image size decreased, the accuracy of the shrimp larvae count clearly increased. When the segmentation image size was reduced to  $50 \times 50$  pixels, the counting accuracy was the best, but we also realize that the cost of computing increased exponentially. Considering the calculation cost and the counting accuracy, we chose  $100 \times 100$  pixels for the segmentation image size in the counting experiments hereafter.

#### 4.5. Discussion

Comparative experiments showed that the proposed region segmentation algorithm improved the accuracy of counting shrimp larvae. In contrast to the standard YOLOv5 model, the enhanced YOLOv5 variant provides more accurate detection of small objects such as shrimp larvae. Validation experiments indicated that the proposed algorithm performed better, with a counting accuracy of 98% even under conditions with high densities and large volumes of shrimp larvae. However, there are still aspects which need to be improved. To begin with, the dataset for this experiment was gathered using an optical platform which we designed. To improve the generalizability of the algorithm, it is essential to include images of shrimp larvae obtained from various environments and a range of imaging devices in the training dataset. In addition, the stitching and deduplication process occasionally overlooks a few shrimp larvae due to stitching errors, suggesting that this algorithm needs further improvement. In conclusion, the proposed algorithm can be integrated into shrimp larvae imaging devices to develop a comprehensive shrimp larvae counting system.

The image segmentation approach and the improved YOLOv5 model proposed in this paper not only solve the problem of shrimp larvae counting but also have significant research potential in various fields, such as dense crowd counting and medical image analysis. Using segmentation algorithms, the proportion of the pixel of the target objects in the input images increased, thereby improving the detection accuracy. However, the algorithm needs to be adjusted according to the specific problem at hand.

#### 5. Conclusions

We introduced a method for counting shrimp larvae which uses region segmentation along with an improved YOLOv5 model. By segmenting the regions, we can reduce the receptive field area, thereby enhancing the detection accuracy of shrimp larvae using the improved YOLOv5 model. Furthermore, the deduplication model can tackle the problem of repeated counts. Based on this, the experimental results show that the number of larvae was about 5000, and the counting accuracy of our algorithm remained above 98%, which would be suitable for replacing the work of manually counting shrimp larvae.

**Author Contributions:** Conceptualization, Y.Z.; methodology, Y.Z., J.W. and H.D.; software, J.W. and H.D.; validation, X.W., T.P., X.L. and D.D.; formal analysis, J.W. and H.D.; investigation, J.W., X.W. and H.D.; resources, T.P.; data curation, D.D.; writing—original draft preparation, H.D.; writing—review and editing, Y.Z.; visualization, Y.Z.; supervision, Y.Z.; project administration, Y.Z.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are available from the authors on request.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Racotta, I.S.; Palacios, E.; Ibarra, A.M. Shrimp larval quality in relation to broodstock condition. *Aquaculture* **2003**, *227*, 107–130. [\[CrossRef\]](#)
2. Li, D.; Miao, Z.; Peng, F.; Wang, L.; Hao, Y.; Wang, Z.; Chen, T.; Li, H.; Zheng, Y. Automatic counting methods in aquaculture: A review. *J. World Aquac. Soc.* **2021**, *52*, 269–283. [\[CrossRef\]](#)
3. Tierney, T.W.; Fleckenstein, L.J.; Ray, A.J. The effects of density and artificial substrate on intensive shrimp *Litopenaeus vannamei* nursery production. *Aquac. Eng.* **2020**, *89*, 102063. [\[CrossRef\]](#)
4. Naegel, L.C.; Gómez-Humáran, I.M. Effect of sample volume and population density on precision of larval population estimates. *Aquac. Eng.* **1998**, *17*, 11–19. [\[CrossRef\]](#)
5. Hsieh, Y.K.; Hsieh, J.W.; Hu, W.C.; Tseng, Y.C. AIoT-Based Shrimp Larvae Counting System Using Scaled Multilayer Feature Fusion Network. *IEEE Internet Things J.* **2024**, *early Access*. [\[CrossRef\]](#)
6. Zhang, L.; Li, W.; Liu, C.; Zhou, X.; Duan, Q. Automatic fish counting method using image density grading and local regression. *Comput. Electron. Agric.* **2020**, *179*, 105844. [\[CrossRef\]](#)
7. Yeh, C.T.; Ling, M.S. Portable Device for Ornamental Shrimp Counting Using Unsupervised Machine Learning. *Sens. Mater.* **2021**, *33*, 3027–3036. [\[CrossRef\]](#)
8. Martinez-Palacios, C.; Novoa, M.O.; Chavez-Martinez, C. A simple apparatus for self-separation of post-larval prawns, *Macrobrachium* spp., in mixed larval rearing tanks. *Aquac. Res.* **1985**, *16*, 341–348. [\[CrossRef\]](#)
9. Work, H.P.S. Photoelectric Sensors for Counting and Classifying Vehicles. In *Transportation Research Record 1311*; Transportation Research Board: Washington, DC, USA, 1991; p. 79.
10. Carmichael, H. Photoelectric Detection II. In *An Open Systems Approach to Quantum Optics: Lectures Presented at the Université Libre de Bruxelles October 28 to November 4, 1991*; Springer: Berlin/Heidelberg, Germany, 1993; pp. 93–112.
11. Spratt, M.D. Preliminary results of a computer imaging system applied to estimating the quantity of larvae and fingerling fish for aquaculture. In *Fish Quality Control by Computer Vision*; Routledge: New York, NY, USA, 2017; pp. 263–282.
12. Awalludin, E.; Muhammad, W.W.; Arsal, T.; Yusoff, W.H.W. Fish larvae counting system using image processing techniques. In *Proceedings of the Journal of Physics Conference Series*; IOP Publishing: Bandung, Indonesia, 2020; Volume 1529, p. 052040.
13. Hu, W.C.; Chen, L.B.; Hsieh, M.H.; Ting, Y.K. A deep-learning-based fast counting methodology using density estimation for counting shrimp larvae. *IEEE Sens. J.* **2022**, *23*, 527–535. [\[CrossRef\]](#)
14. Sun, Y.; Lin, Y.; Zhao, G.; Svanberg, S. Identification of flying insects in the spatial, spectral, and time domains with focus on mosquito imaging. *Sensors* **2021**, *21*, 3329. [\[CrossRef\]](#)
15. Patwal, A.; Diwakar, M.; Tripathi, V.; Singh, P. Crowd counting analysis using deep learning: A critical review. *Procedia Comput. Sci.* **2023**, *218*, 2448–2458. [\[CrossRef\]](#)
16. Kaewchote, J.; Janyong, S.; Limprasert, W. Image recognition method using Local Binary Pattern and the Random forest classifier to count post larvae shrimp. *Agric. Nat. Resour.* **2018**, *52*, 371–376. [\[CrossRef\]](#)
17. Nguyen, K.T.; Nguyen, C.N.; Wang, C.Y.; Wang, J.C. Two-phase instance segmentation for whiteleg shrimp larvae counting. In *Proceedings of the 2020 IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, 4–6 January 2020; pp. 1–3.
18. Qu, Y.; Jiang, S.; Li, D.; Zhong, P.; Shen, Z. SLCOBNet: Shrimp larvae counting network with overlapping splitting and Bayesian-DM-count loss. *Biosyst. Eng.* **2024**, *244*, 200–210. [\[CrossRef\]](#)
19. Zhang, J.; Yang, G.; Sun, L.; Zhou, C.; Zhou, X.; Li, Q.; Bi, M.; Guo, J. Shrimp egg counting with fully convolutional regression network and generative adversarial network. *Aquac. Eng.* **2021**, *94*, 102175. [\[CrossRef\]](#)
20. Kesvarakul, R.; Chianrabutra, C.; Chianrabutra, S. Baby shrimp counting via automated image processing. In *Proceedings of the 9th International Conference on Machine Learning and Computing*, Singapore, 24–26 February 2017; pp. 352–356.
21. Thai, T.T.N.; Nguyen, T.S.; Pham, V.C. Computer vision based estimation of shrimp population density and size. In *Proceedings of the 2021 International Symposium on Electrical and Electronics Engineering (ISEE)*, Ho Chi Minh, Vietnam, 15–16 April 2021; pp. 145–148.

22. Awalludin, E.A.; Yaziz, M.M.; Rahman, N.A.; Yussof, W.N.J.H.W.; Hitam, M.S.; Arsal, T.T. Combination of canny edge detection and blob processing techniques for shrimp larvae counting. In Proceedings of the 2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuala Lumpur, Malaysia, 17–19 September 2019; pp. 308–313.
23. Zhou, C.; Yang, G.; Sun, L.; Wang, S.; Song, W.; Guo, J. Counting, locating, and sizing of shrimp larvae based on density map regression. *Aquac. Int.* **2024**, *32*, 3147–3168. [\[CrossRef\]](#)
24. Zhang, L.; Zhou, X.; Li, B.; Zhang, H.; Duan, Q. Automatic shrimp counting method using local images and lightweight YOLOv4. *Biosyst. Eng.* **2022**, *220*, 39–54. [\[CrossRef\]](#)
25. Bereciartua-Pérez, A.; Gómez, L.; Picón, A.; Navarra-Mestre, R.; Klukas, C.; Eggers, T. Insect counting through deep learning-based density maps estimation. *Comput. Electron. Agric.* **2022**, *197*, 106933. [\[CrossRef\]](#)
26. Yu, X.; Wang, J.; Wen, S.; Yang, J.; Zhang, F. A deep learning based feature extraction method on hyperspectral images for nondestructive prediction of TVB-N content in Pacific white shrimp (*Litopenaeus vannamei*). *Biosyst. Eng.* **2019**, *178*, 244–255. [\[CrossRef\]](#)
27. Armalivia, S.; Zainuddin, Z.; Achmad, A.; Wicaksono, M.A. Automatic counting shrimp larvae based you only look once (YOLO). In Proceedings of the 2021 International Conference on Artificial Intelligence and Mechatronics Systems (AIMS), Bandung, Indonesia, 28–30 April 2021; pp. 1–4.
28. Liu, S.; Huang, D. Receptive field block net for accurate and fast object detection. In Proceedings of the European conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 385–400.
29. Zhang, Y.; Zhao, D.; Zhang, J.; Xiong, R.; Gao, W. Interpolation-dependent image downsampling. *IEEE Trans. Image Process.* **2011**, *20*, 3291–3296. [\[CrossRef\]](#)
30. Lan, Y.h.; Zhang, Y.; Li, C.h.; Zhao, X.f. A novel image segmentation method based on random walk. In Proceedings of the 2009 Asia-Pacific Conference on Computational Intelligence and Industrial Applications (PACIIA), Wuhan, China, 28–29 November 2009; Volume 1, pp. 207–210.
31. Lin, W.; Chan, A.B. Optimal transport minimization: Crowd localization on density maps for semi-supervised counting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023; pp. 21663–21673.
32. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
33. Han, X.; Chang, J.; Wang, K. You only look once: Unified, real-time object detection. *Procedia Comput. Sci.* **2021**, *183*, 61–72. [\[CrossRef\]](#)
34. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
35. Redmon, J. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
36. Hussain, M. YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision. *arXiv* **2024**, arXiv:2407.02988.
37. Fu, H.; Song, G.; Wang, Y. Improved YOLOv4 marine target detection combined with CBAM. *Symmetry* **2021**, *13*, 623. [\[CrossRef\]](#)
38. Wang, Z.; Jin, L.; Wang, S.; Xu, H. Apple stem/calyx real-time recognition using YOLO-v5 algorithm for fruit automatic loading system. *Postharvest Biol. Technol.* **2022**, *185*, 111808. [\[CrossRef\]](#)
39. Kim, J.H.; Kim, N.; Park, Y.W.; Won, C.S. Object detection and classification based on YOLO-V5 with improved maritime dataset. *J. Mar. Sci. Eng.* **2022**, *10*, 377. [\[CrossRef\]](#)
40. Yang, R.; Li, W.; Shang, X.; Zhu, D.; Man, X. KPE-YOLOv5: An improved small target detection algorithm based on YOLOv5. *Electronics* **2023**, *12*, 817. [\[CrossRef\]](#)
41. Wang, L.; Liu, X.; Ma, J.; Su, W.; Li, H. Real-time steel surface defect detection with improved multi-scale YOLO-v5. *Processes* **2023**, *11*, 1357. [\[CrossRef\]](#)
42. Sun, Q.; Li, P.; He, C.; Song, Q.; Chen, J.; Kong, X.; Luo, Z. A Lightweight and High-Precision Passion Fruit YOLO Detection Model for Deployment in Embedded Devices. *Sensors* **2024**, *24*, 4942. [\[CrossRef\]](#)
43. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. 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, Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
44. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. Yolov10: Real-time end-to-end object detection. *arXiv* **2024**, arXiv:2405.14458.
45. Dong, X.; Yan, S.; Duan, C. A lightweight vehicles detection network model based on YOLOv5. *Eng. Appl. Artif. Intell.* **2022**, *113*, 104914. [\[CrossRef\]](#)
46. Chai, E.; Ta, L.; Ma, Z.; Zhi, M. ERF-YOLO: A YOLO algorithm compatible with fewer parameters and higher accuracy. *Image Vis. Comput.* **2021**, *116*, 104317. [\[CrossRef\]](#)
47. Kim, B.J.; Choi, H.; Jang, H.; Lee, D.G.; Jeong, W.; Kim, S.W. Dead pixel test using effective receptive field. *Pattern Recognit. Lett.* **2023**, *167*, 149–156. [\[CrossRef\]](#)
48. Wang, Q.; Qian, Y.; Hu, Y.; Wang, C.; Ye, X.; Wang, H. M2YOLOF: Based on effective receptive fields and multiple-in-single-out encoder for object detection. *Expert Syst. Appl.* **2023**, *213*, 118928. [\[CrossRef\]](#)

49. Chen, Y.; Zhan, S.; Cao, G.; Li, J.; Wu, Z.; Chen, X. C2f-Enhanced YOLOv5 for Lightweight Concrete Surface Crack Detection. In Proceedings of the 2023 International Conference on Advances in Artificial Intelligence and Applications, Wuhan, China, 18–20 November 2023; pp. 60–64.
50. Wang, W.; Tan, X.; Zhang, P.; Wang, X. A CBAM based multiscale transformer fusion approach for remote sensing image change detection. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 6817–6825. [[CrossRef](#)]
51. Smart Shrimp Farm. 2024. Available online: <http://m.ux6.com/app/222810.html> (accessed on 1 March 2024).
52. Zhang, Y.; Guo, Z.; Wu, J.; Tian, Y.; Tang, H.; Guo, X. Real-time vehicle detection based on improved yolo v5. *Sustainability* **2022**, *14*, 12274. [[CrossRef](#)]
53. Liu, D.; Xu, B.; Cheng, Y.; Chen, H.; Dou, Y.; Bi, H.; Zhao, Y. Shrimpseed\_Net: Counting of shrimp seed using deep learning on smartphones for aquaculture. *IEEE Access* **2023**, *11*, 85441–85450. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.