SuperYOLO: Super Resolution Assisted Object Detection in Multimodal Remote Sensing Imagery

Jiaqing Zhang, Jie Lei, *Member, IEEE*, Weiying Xie, *Member, IEEE*, Zhenman Fang, *Member, IEEE*, Yunsong Li, *Member, IEEE*, and Qian Du, *Fellow, IEEE*

***Abstract*— Accurately and timely detecting multiscale small objects that contain tens of pixels from remote sensing images (RSI) remains challenging. Most of the existing solutions primarily design complex deep neural networks to learn strong feature representations for objects separated from the background, which often results in a heavy computation burden. In this article, we propose an accurate yet fast object detection method for RSI, named SuperYOLO, which fuses multimodal data and performs high-resolution (HR) object detection on multiscale objects by utilizing the assisted super resolution (SR) learning and considering both the detection accuracy and computation cost. First, we utilize a symmetric compact multimodal fusion (MF) to extract supplementary information from various data for improving small object detection in RSI. Furthermore, we design a simple and flexible SR branch to learn HR feature representations that can discriminate small objects from vast backgrounds with low-resolution (LR) input, thus further improving the detection accuracy. Moreover, to avoid introducing additional computation, the SR branch is discarded in the inference stage, and the computation of the network model is reduced due to the LR input. Experimental results show that, on the widely used VEDAI RS dataset, SuperYOLO achieves an accuracy of 75.09% (in terms of mAP_{50}), which is more than 10% higher than the SOTA large models, such as YOLOv5l, YOLOv5x, and RS designed YOLOrs. Meanwhile, the parameter size and GFLOPs of SuperYOLO are about 18× and 3.8× less than YOLOv5x. Our proposed model shows a favorable accuracy–speed tradeoff compared to the state-of-the-art models. The code will be open-sourced at https://github.com/icey-zhang/SuperYOLO.**

***Index Terms*— Feature fusion, multimodal remote sensing image, object detection, super resolution (SR).**

I. Introduction

OBJECT detection plays an important role in various fields involving computer-aided diagnosis or autonomous piloting. Over the past decades, numerous excellent deep neural network (DNN)-based object detection frameworks [1], [2], [3], [4], [5] have been proposed, updated, and optimized in computer vision. The remarkable accuracy enhancement of DNN-based object detection frameworks owes to the application of large-scale natural datasets with accurate annotations [6], [7], [8].

Compared with natural scenarios, there are several vital challenges for accurate object detection in remote sensing images (RSIs). First, the number of labeled samples is relatively small, which limits the training of DNNs to achieve high detection accuracy. Second, the size of objects in RSI is much smaller, accounting for merely tens of pixels in relation to the complicated and broad backgrounds [9], [10]. Moreover, the scale of those objects is diverse with multiple categories [11]. As shown in Fig. 1(a), the object car is considerably small within a vast area. As shown in Fig. 1(b), the objects have large-scale variations, to which the scale of a car is smaller than that of a camping vehicle.

Currently, most object detection techniques are solely designed and applied for a single modality, such as red-green-blue (RGB) and infrared (IR) [12], [13]. Consequently, with respect to object detection, its capability to recognize objects on the Earth's surface remains insufficient due to the deficiency of complementary information between different modalities [14]. As imaging technology flourishes, RSIs collected from multimodality become available and provide an opportunity to improve detection accuracy. For example, as shown in Fig. 1, the fusion of two different multimodalities (RGB and IR) can effectively enhance the detection accuracy in RSI. Sometimes, the resolution of one modality is low, which requires technique to improve the resolution to enhance information. Recently, super resolution (SR) technology has shown great potential in remote sensing fields [15], [16], [17], [18]. Benefiting from the vigorous development of the convolutional neuron network (CNN), the resolution of the remote sensing image has achieved high texture information to be interpreted. However, due to the high computation cost of the CNN network, the application of the SR network in real-time practical tasks has become a hot topic in current research.

In this study, our motivation is to *propose an on-board real-time object detection framework for multimodal RSIs to achieve high detection accuracy and high inference speed without introducing additional computation overhead.* Inspired by recent advances in real-time compact neural network

Manuscript received 16 November 2022; revised 2 February 2023 and 26 February 2023; accepted 7 March 2023. Date of publication 17 March 2023; date of current version 31 March 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62071360. *(Corresponding author: Jie Lei.)*

Jiaqing Zhang, Jie Lei, Weiying Xie, and Yunsong Li are with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: jqzhang_2@stu.xidian.edu.cn; jielei@mail.xidian.edu.cn; wyxie@xidian.edu.cn; ysli@mail.xidian.edu.cn).

Zhenman Fang is with the School of Engineering Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (e-mail: zhenman@sfu.ca).

Qian Du is with the Department of Electronic and Computer Engineering, Mississippi State University, Starkville, MS 39759 USA (e-mail: du@ece.msstate.edu).

Digital Object Identifier 10.1109/TGRS.2023.3258666

1558-0644 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

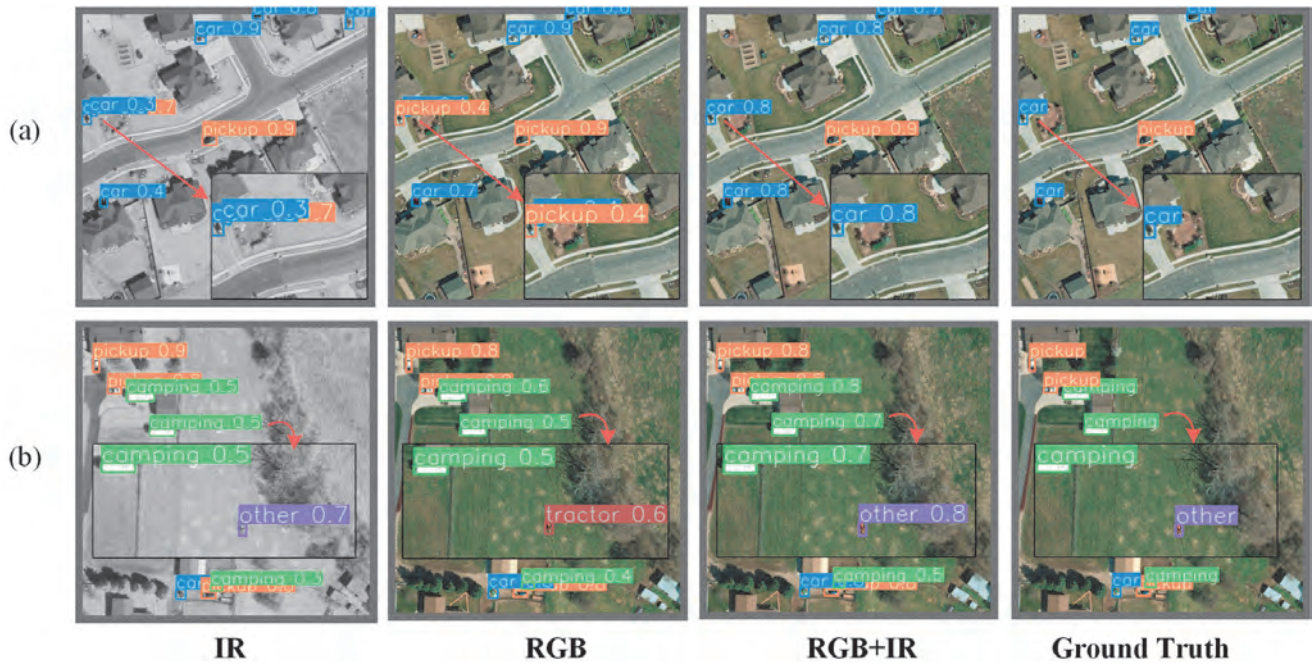

Fig. 1. Visual comparison of RGB image, IR image, and ground truth (GT). The IR image provides vital complementary information for resolving the challenges in RGB detection. The object car in (a) is considerably small within a vast area. In (b), the objects have large-scale variation, to which the scale of a car is smaller than that of a camping vehicle. The fusion of RGB and IR modalities effectively enhances detection performance.

models, we choose small-size YOLOv5s [19] structure as our detection baseline. It can reduce deployment costs and facilitate rapid deployment of the model. Considering the high-resolution (HR) retention requirements for small objects, we remove the Focus module in the baseline YOLOv5s model, which not only benefits defining the location of small dense objects but also enhances the detection performance. Considering the complementary characteristics in different modalities, we propose a multimodal fusion (MF) scheme to improve the detection performance for RSI. We evaluate different fusion alternatives (pixel-level or feature-level) and choose pixel-level fusion for low computation cost.

Lastly and most importantly, we develop an SR assurance module to guide the network to generate HR features that are capable of identifying small objects in vast backgrounds, thereby reducing false alarms induced by background-contaminated objects in RSI. Nevertheless, a naive SR solution can significantly increase the computation cost. Therefore, we set the auxiliary SR branch engaged in the training process and remove it in the inference stage, facilitating spatial information extraction in HR without increasing computation cost.

In summary, this article makes the following contributions.

1) We propose a computation-friendly pixel-level fusion method to combine inner information bidirectionally in a symmetric and compact manner. It efficiently decreases the computation cost without sacrificing accuracy compared with feature-level fusion.
2) We introduce an assisted SR branch into multimodal object detection for the first time. Our approach not only makes a breakthrough in limited detection performance but also paves a more flexible way to study outstanding HR feature representations that are capable of discriminating small objects from vast backgrounds with low-resolution (LR) input.
3) Considering the demand for high-quality results and low-computation cost, the SR module functioning as an auxiliary task is removed during the inference stage without introducing additional computation. The SR branch is general and extensible and can be inserted in the existing fully convolutional network (FCN) framework.
4) The proposed SuperYOLO markedly improves the performance of object detection, outperforming SOTA detectors in real-time multimodal object detection. Our proposed model shows a favorable accuracy–speed tradeoff compared to the state-of-the-art models.

II. Related Work

A. Object Detection With Multimodal Data

Recently, multimodal data have been widely leveraged in numerous practical application scenarios, including visual question answering [20], auto-pilot vehicles [21], saliency detection [22], and remote sensing classification [23]. It is found that combining the internal information of multimodal data can efficiently transfer complementary features to avoid certain information of a single modality from being omitted.

In the field of RSI processing, there exist various modalities (e.g., RGB, synthetic aperture radar (SAR), Light Detection and Ranging (LiDAR), IR, panchromatic (PAN), and multispectral (MS) images) from diverse sensors, which can be fused with complementary characteristics to enhance the performance of various tasks [24], [25], [26]. For example, the additional IR modality [27] captures longer thermal wavelengths to improve the detection under difficult weather

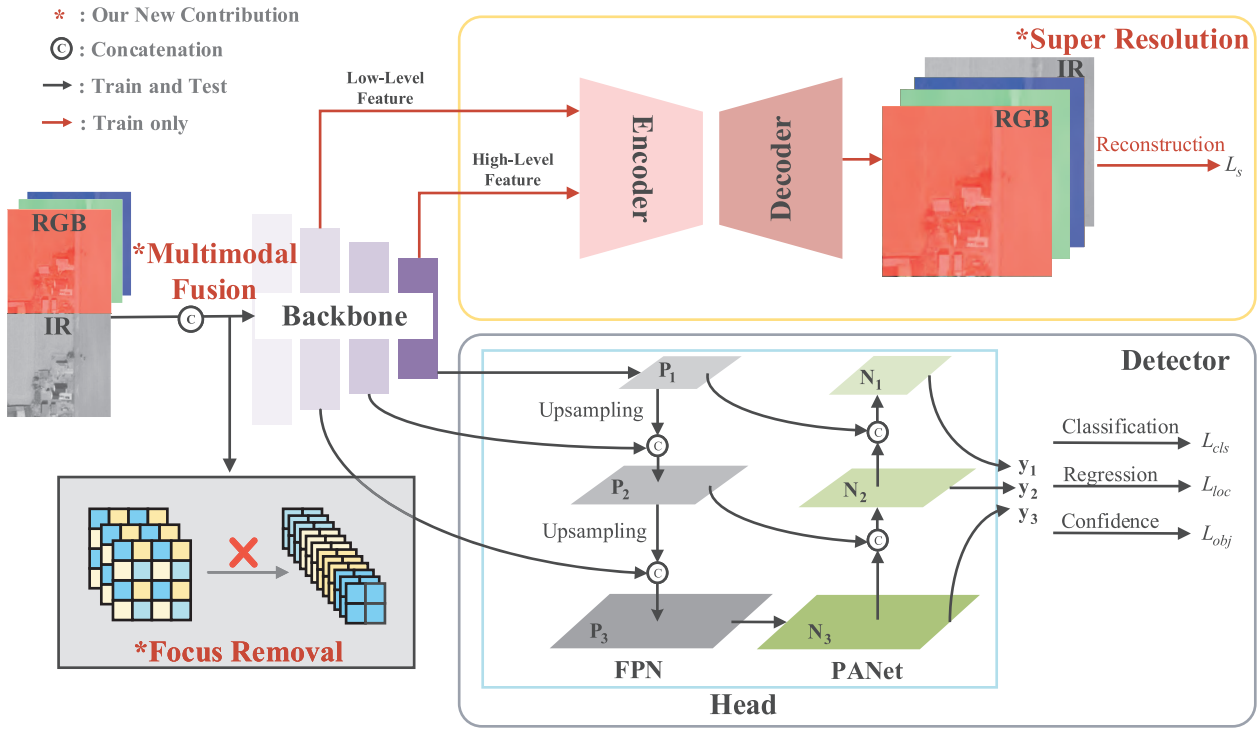

Fig. 2. Overview of the proposed SuperYOLO framework. Our new contributions include: 1) removal of the Focus module to reserve HR; 2) MF; and 3) assisted SR branch. The architecture is optimized in terms of mean square error (mse) loss for the SR branch and task-specific loss for object detection. During the training stage, the SR branch guides the related learning of the spatial dimension to enhance the HR information preservation for the backbone. During the test stage, the SR branch is removed to accelerate the inference speed equal to the baseline.

conditions. Manish et al. [27] proposed a real-time framework for object detection in multimodal remote sensing imaging, in which the extended version conducted mid-level fusion and merged data from multiple modalities. Despite that multisensor fusion can enhance the detection performance, as shown in Fig. 1, hardly can its low-accuracy detection performance and to-be-improved computing speed meet the requirements of real-time detection tasks.

The fusion methods are primarily grouped into three strategies, i.e., pixel-level fusion, feature-level fusion, and decision-level fusion methods [28]. The decision-level fusion methods fuse the detection results during the last stage, which may consume enormous computation resources due to repeated calculations for different multimodal branches. In the field of remote sensing, feature-level fusion methods are mainly adopted with multi branches. The multimodal images will be input into the parallel branches to extract respective independent features of different modalities, and then, these features will be combined by some operations, such as attention module or simple concatenation. The parallel branches bring repeated computation as the modalities increase, which is not friendly in the real-time tasks in remote sensing.

In contrast, the adoption of pixel-level fusion methods can reduce unnecessary computation. In this article, our proposed SuperYOLO fuses the modalities at the pixel level to significantly reduce the computation cost and design operations in spatial and channel domains to extract inner information in the different modalities, which can help enhance detection accuracy.

B. Super Resolution in Object Detection

In the recent literature, the performance of small object detection can be improved by multiscale feature learning [29], [30] and context-based detection [31]. These methods always enhance the information representation ability of the network in different scales but ignore the HR contextual information reservation. Conducted in a preprocessing step, SR has proven to be effective and efficient in various object detection tasks [32], [33]. Shermeyer and Van Etten [34] quantified its effect on the detection performance of satellite imaging by multiple resolutions of RSI. Based on generative adversarial networks (GANs), Courtrai et al. [35] utilized SR to generate HR images that were fed into the detector to improve its detection performance. Rabbi et al. [36] leveraged a Laplacian operator to extract edges from the input image to enhance the capability of reconstructing HR images, thus improving its performance in object localization and classification. Ji et al. [37] introduced a cycle-consistent GAN structure as an SR network and modified faster R-CNN architecture to detect vehicles from enhanced images that are produced by the SR network. In these works, the adoption of the SR structure has effectively addressed the challenges regarding small objects. However, compared with single detection models, additional computation is introduced, which attributes to the enlarged scale of the input image by HR design.

Recently, Wang et al. [38] proposed an SR module that can maintain HR representations with LR input while reducing the model computation in segmentation tasks. Inspired by Wang et al. [38], we design an SR assisted branch. In contrast

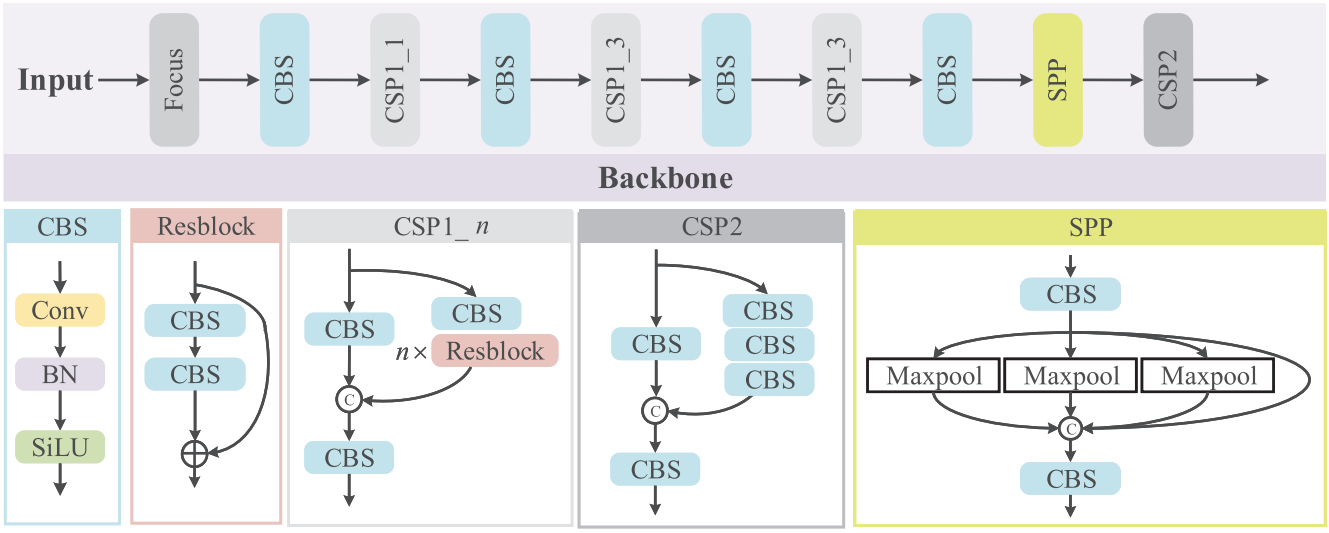

Fig. 3. Backbone structure of YOLOv5s. The low-level texture and high-level semantic features are extracted by stacked CSP, CBS, and SPP structures.

to the aforementioned work in which the SR is realized in the start stage, the assisted SR module guides the learning of high-quality HR representations for the detector, which not only strengthens the response of small dense objects but also improves the performance of the object detection in spatial space. Moreover, the SR module is removed in the inference stage to avoid extra computation.

III. Baseline Architecture

As shown in Fig. 2, the baseline YOLOv5 network consists of two main components: the backbone and head (including the neck). The backbone is designed to extract low-level texture and high-level semantic features. Next, these hint features are fed to the head to construct the enhanced feature pyramid network from top to bottom to transfer robust semantic features and from bottom to top to propagate a strong response of local texture and pattern features. This resolves the various scale issue of the objects by yielding an enhancement of detection with diverse scales.

In Fig. 3, CSPNet [39] is utilized as the backbone to extract the feature information, consisting of numerous sample Convolution-Batch-normalization-SiLu (CBS) components and cross stage partial (CSP) modules. The CBS is composed of operations of convolution, batch normalization, and activation function SiLu [40]. The CSP duplicates the feature map of the previous layer into two branches and then halves the channel numbers through 1×1 convolution, by which the computation is, therefore, reduced. With respect to the two copies of the feature map, one is connected to the end of the stage, and the other is sent into ResNet blocks or CBS blocks as the input. Finally, the two copies of the feature map are concatenated to combine the features, which is followed by a CBS block. The spatial pyramid pooling (SPP) module [41] is composed of parallel maxpool layers with different kernel sizes and is utilized to extract multiscale deep features. The low-level texture and high-level semantic features are extracted by stacked CSP, CBS, and SPP structures.

Limitation 1: It is worth mentioning that the Focus module is introduced to decrease the number of computations. As shown in Fig. 2 (bottom left), inputs are partitioned into individual pixels, reconstructed at intervals, and, finally, concatenated in the channel dimension. The inputs are resized to a smaller scale to reduce the computation cost and accelerate the network training and inference speed. However, this may sacrifice object detection accuracy to a certain extent, especially for small objects vulnerable to resolution.

Limitation 2: It is known that the backbone of YOLO employs deep convolutional neural networks to extract hierarchical features with a stride step of 2, through which the size of the extracted features is halved. Hence, the feature size retained for multiscale detection is far smaller than that of the original input image. For example, when the input image size is 608, the sizes of output features for the last detection layer are 76, 38, and 19, respectively. LR features may result in the missing of some small objects.

IV. SuperYOLO Architecture

As summarized in Fig. 2, we introduce three new contributions to our SuperYOLO network architecture. First, we remove the Focus module in the backbone and replace it with an MF module to avoid resolution degradation and, thus, accuracy degradation. Second, we explore different fusion methods and choose the computation-efficient pixel-level fusion to fuse RGB and IR modalities to refine dissimilar and complementary information. Finally, we add an assisted SR module in the training stage, which reconstructs the HR images to guide the related backbone learning in spatial dimension and, thus, maintain HR information. In the inference stage, the SR branch is discarded to avoid introducing additional computation overhead.

A. Focus Removal

As presented in Section III and Fig. 2 (bottom left), the Focus module in the YOLOv5 backbone partitions images at

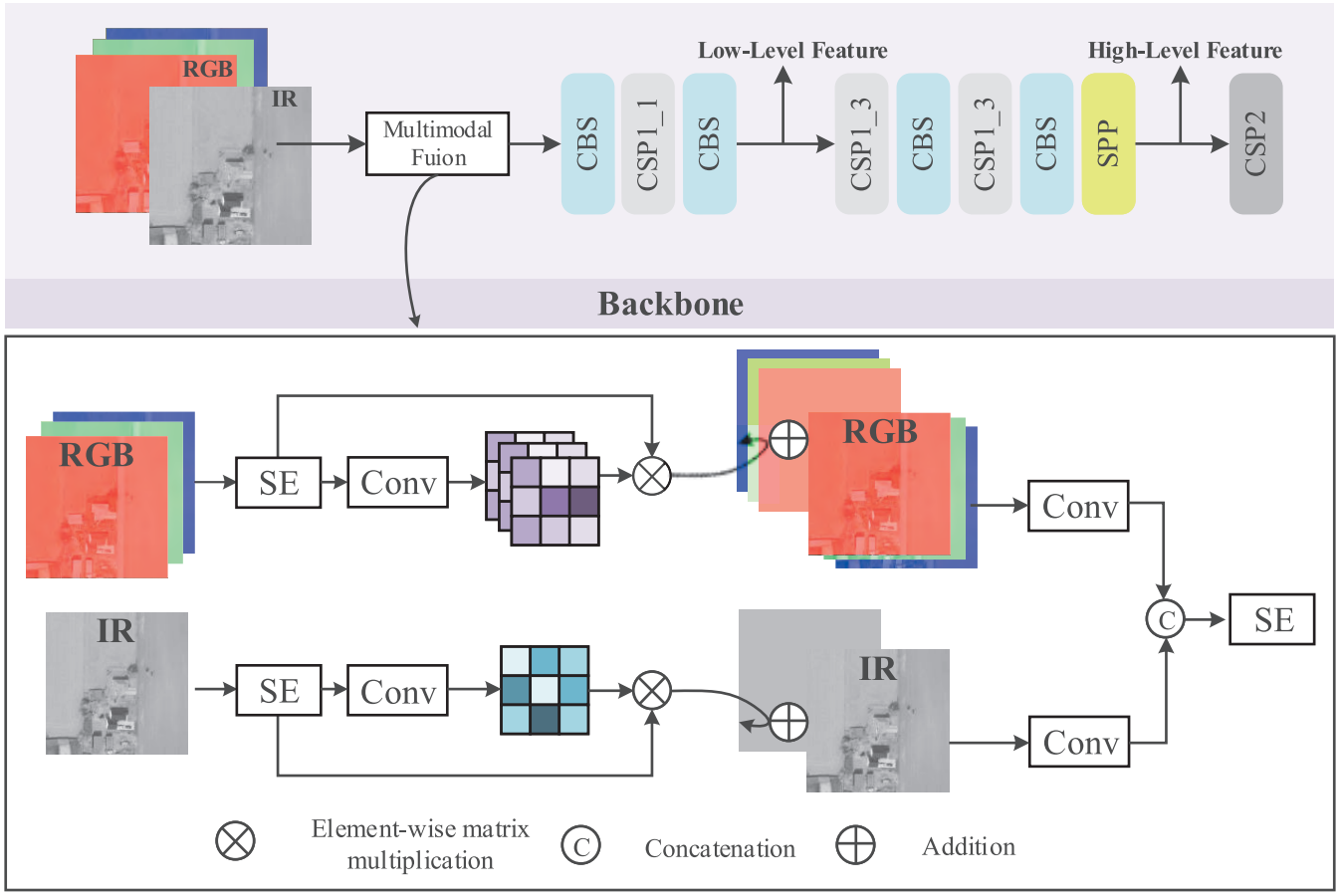

Fig. 4. Architecture of the MF module at the pixel level.

intervals on the spatial domain and then reorganize the new image to resize the input images. Specifically, this operation is to collect a value for every group of pixels in an image and then reconstruct it to obtain smaller complementary images. The size of the rebuilt images decreases with the increase in the number of channels. As a result, it causes resolution degradation and spatial information loss for small targets. Considering that the detection of small targets depends more heavily on higher resolution, the Focus module is abandoned and replaced by an MF module (as shown in Fig. 4) to prevent the resolution from being degraded.

B. Multimodal Fusion

The more the information is utilized to distinguish objects, the better the performance can be achieved in object detection. MF is an effective path for merging different information from various sensors. The decision-, feature-, and pixel-level fusions are the three mainstream fusion methods that can be deployed at different depths of the network. Since decision-level fusion requires enormous computation, it is not considered in SuperYOLO.

We propose a pixel-level MF to extract the shared and special information from the different modalities. The MF can combine multimodal inner information bidirectionally in a symmetric and compact manner. As shown in Fig. 4, for the pixel-level fusion, we first normalize an input RGB image and an input IR image into two intervals of [0, 1]. The input modalities $X_{\mathrm{RGB}}, X_{\mathrm{IR}} \in \mathbb{R}^{C\times H\times W}$ are subsampled to $I_{\mathrm{RGB}}, I_{\mathrm{IR}} \in \mathbb{R}^{C\times (H/n)\times (W/n)}$, which are fed to SE blocks extracting inner information in channel domain [42] to generate $F_{\mathrm{RGB}}, F_{\mathrm{IR}}$

$$F_{\mathrm{RGB}} = SE(I_{\mathrm{RGB}}), \quad F_{\mathrm{IR}} = SE(I_{\mathrm{IR}}). \tag{1}$$

Then, the attention map that reveals the inner relationship of the different modalities in the spatial domain is defined as

$$m_{\mathrm{IR}} = f_1(F_{\mathrm{IR}}), \quad m_{\mathrm{RGB}} = f_2(F_{\mathrm{RGB}}) \tag{2}$$

where f_1 and f_2 represent 1×1 convolutions for the RGB and IR modalities, respectively. Here, $\otimes$ denotes the element-wise matrix multiplication. Inner spatial information between the different modalities is produced by

$$F_{\mathrm{in1}} = m_{\mathrm{RGB}} \otimes F_{\mathrm{RGB}}, F_{\mathrm{in2}} = m_{\mathrm{IR}} \otimes F_{\mathrm{IR}}. \tag{3}$$

To incorporate internal inner view information and spatial texture information, the features are added by the original input modalities and then fed into 1×1 convolutions. The full features are

$$F_{\mathrm{ful1}} = f_3(F_{\mathrm{in1}} + I_{\mathrm{RGB}}), \quad F_{\mathrm{ful2}} = f_4(F_{\mathrm{in2}} + I_{\mathrm{IR}}) \tag{4}$$

where f_3 and f_4 represent 1×1 convolutions. Finally, the features are fused by

$$F_o = \mathrm{SE}(\mathrm{Concat}(F_{\mathrm{ful1}}, F_{\mathrm{ful2}})) \tag{5}$$

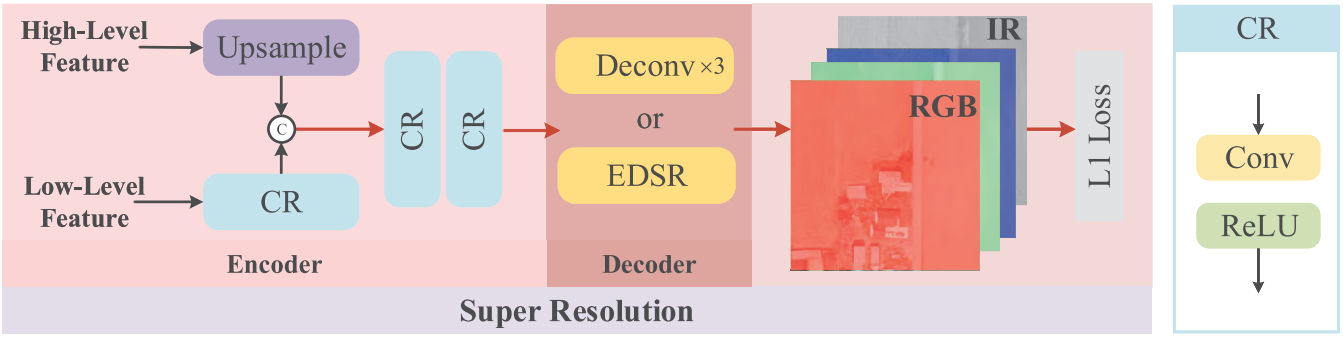

Fig. 5. SR structure of SuperYOLO. The SR structure can be regarded as a simple encode–decoder model. The low- and high-level features of the backbone are selected to fuse local textures patterns and semantic information, respectively.

where Concat(·) denotes the concatenation operation along the channel axis. The result is then fed to the backbone to produce multilevel features. Note that X is subsampled to $1/n$ size of the original image to accomplish the SR module discussed in Section IV-C and to accelerate the training process. The X represents the RGB or IR modality, and the sampled image is denoted as $I \in \mathbb{R}^{C\times(H/n)\times(W/n)}$ and generated by

$$I = D(X) \tag{6}$$

where $D(\cdot)$ represents the n times downsampling operation using bilinear interpolation.

C. Super Resolution

As mentioned in Section III, the feature size retained for multiscale detection in the backbone is far smaller than that of the original input image. Most of the existing methods conduct upsampling operations to recover the feature size. Unfortunately, this approach has produced limited success due to the information loss in texture and pattern, which explains that it is inappropriate to employ this operation to detect small targets that require HR preservation in RSI.

To address this issue, as shown in Fig. 2, we introduce an auxiliary SR branch. First, the introduced branch shall facilitate the extraction of HR information in the backbone and achieve satisfactory performance. Second, the branch should not add more computation to reduce the inference speed. It shall realize a tradeoff between accuracy and computation time during the inference stage. Inspired by the study of Wang et al. [38] where the proposed SR succeeded in facilitating segmentation tasks without additional requirements, we introduce a simple and effective branch named SR into the framework. Our proposal can improve detection accuracy without computation and memory overload, especially under circumstances of LR input.

Specifically, the SR structure can be regarded as a simple encode–decoder model. We select the backbone's low- and high-level features to fuse local textures and patterns, and semantic information, respectively. As depicted in Fig. 4, we select the result of the fourth and ninth modules as the low- and high-level features, respectively. The encoder integrates the low-level feature and high-level feature generated in the backbone. As illustrated in Fig. 5, in the encoder, the first CR module is conducted on the low-level feature. For the high-level feature, we use an upsampling operation to match the spatial size of the low-level feature, and then, we use a concatenation operation and two CR modules to merge the low- and high-level features. The CR module includes a convolution and ReLU. For the decoder, the LR feature is upscaled to the HR space in which the SR module's output size is twice larger than that of the input image. As illustrated in Fig. 5, the decoder is implemented using three deconvolutional layers. The SR guides the related learning of spatial dimension and transfers it to the main branch, thereby improving the performance of object detection. In addition, we introduce EDSR [43] as our encoder structure to explore the SR performance and its influence on detection performance.

To present a more visually interpretable description, we visualize the features of backbones for YOLOv5s, YOLOv5x, and SuperYOLO in Fig. 6. The features are upsampled to the same scale as the input image for comparison. By comparing the pairwise images of (c), (f), and (i); (d), (g), and (j); and (e), (h), and (k) in Fig. 6, it can be observed that SuperYOLO contains clearer object structures with higher resolution with the assistance of the SR. Eventually, we obtain a bumper harvest in high-quality HR representation with the SR branch and utilize the head of YOLOv5 to detect small objects.

D. Loss Function

The overall loss of our network consists of two components: detection loss L_o and SR construction loss L_s, which can be expressed as

$$L_{\text{total}} = c_1 L_o + c_2 L_s \tag{7}$$

where c_1 and c_2 are the coefficients for a balance of the two training tasks. The L1 loss (rather than L2 loss) [44] is used to calculate the SR construction loss L_s between the input image X and SR result S, to which the expression is written as

$$L_s = \|S - X\|_1. \tag{8}$$

The detection loss involves three components [19]: loss of judging whether there is an object L_{obj}, loss of object location L_{loc}, and loss of object classification L_{cls}, which are used to evaluate the loss of the prediction as

$$L_o = \lambda_{\text{loc}} \sum_{l=0}^{2} a_l L_{\text{loc}} + \lambda_{\text{obj}} \sum_{l=0}^{2} b_l L_{\text{obj}} + \lambda_{\text{cls}} \sum_{l=0}^{2} c_l L_{\text{cls}} \tag{9}$$

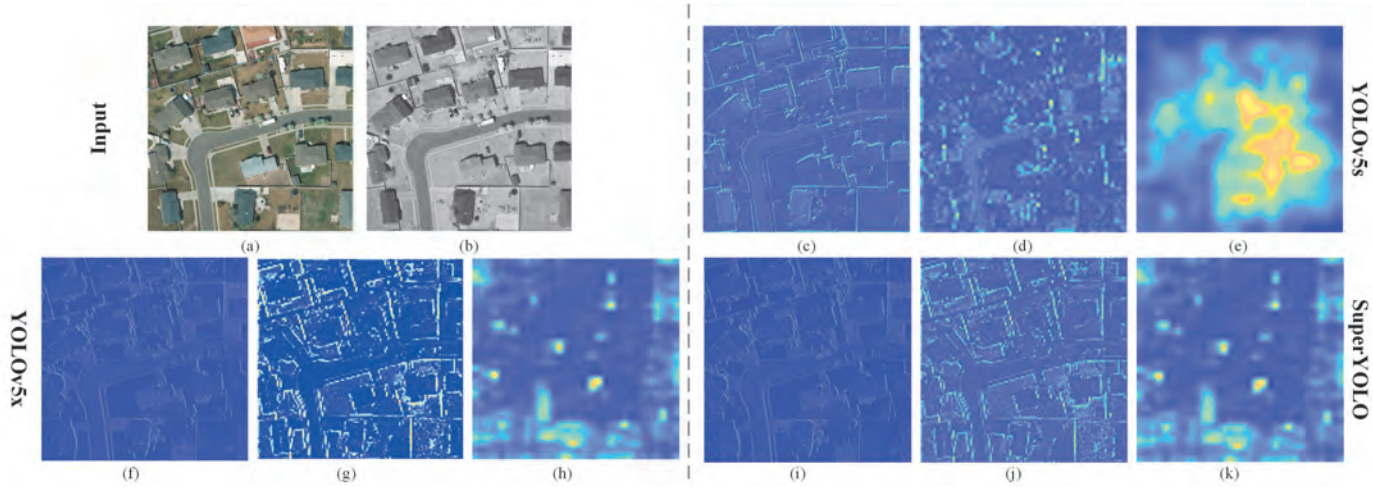

Fig. 6. Feature-level visualization of backbone for YOLOv5s, YOLOv5x, and SuperYOLO with the same input: (a) RGB input, (b) IR input, (c)–(e) features of YOLOv5s, (f)–(h) features of YOLOv5x, and (i)–(k) features of SuperYOLO. The features are upsampled to the same scale as the input image for comparison. (c), (f), and (i) Features in the first layer. (d), (g), and (j) Low-level features. (e), (h), and (k) High-level features in layers at the same depth.

where l represents the layer of the output in the head; a_l, b_l, and c_l are the weights of different layers for the three loss functions; and the weights λ_{loc}, λ_{obj}, and λ_{cls} regulate error emphasis among box coordinates, box dimensions, objectness, no-objectness, and classification.

V. Experimental Results

A. Dataset

The popular Vehicle Detection in Aerial Imagery (VEDAI) dataset [45] is used in the experiments, which contains cropped images obtained from the much larger Utah Automated Geographic Reference Center (AGRC) dataset. Each image collected from the same altitude in AGRC has approximately 16 000 × 16 000 pixels, with a resolution of about 12.5 cm × 12.5 cm per pixel. RGB and IR are the two modalities for each image in the same scenes. The VEDAI dataset consists of 1246 smaller images that focus on diverse backgrounds involving grass, highway, mountains, and urban areas. All images are in the size of 1024 × 1024 or 512 × 512 . The task is to detect 11 classes of different vehicles, such as car, pickup, camping, and truck.

B. Implementation Details

Our proposed framework is implemented in PyTorch and runs on a workstation with an NVIDIA 3090 GPU. The VEDAI dataset is used to train our SuperYOLO. Following [27], the VEDAI dataset is devised for tenfold cross-validation. In each split, 1089 images are used for training, and another 121 images are used for testing. The ablation experiments are conducted on the first fold of data, while the comparisons with previous methods are performed on the ten folds by averaging their results. The annotations for each object in the image contain the coordinates of the bounding box center, the orientation of the object concerning the positive x-axis, the four corners of the bounding box, the class ID, a binary flag identifying whether an object is occluded, and another binary flag identify whether an object is cropped. We do not consider classes with fewer than 50 instances in the dataset, such as plane, motorcycle, and bus. Thus, the annotations of the VEDAI dataset are converted to YOLOv5 format, and we transfer the ID of the interested class to $0, 1, \ldots, 7$, i.e., $N = 8$. Then, the center coordinates of the bounding box are normalized, and the absolute coordinate is transformed into a relative coordinate. Similarly, the length and width of the bounding box are normalized to [0, 1]. To realize the SR assisted branch, the input images of the network are downsampled from 1024 × 1024 size to 512 × 512 during the training process. In the test process, the image size is 512 × 512, which is consistent with the input of other algorithms compared. In addition, data are augmented with hue saturation value (HSV), multiscale, translation, left-right flip, and mosaic. The augmentation strategy is canceled in the test stage. The standard stochastic gradient descent (SGD) [46] is used to train the network with a momentum of 0.937, a weight decay of 0.0005 for the Nesterov accelerated gradients utilized, and a batch size of 2. The learning rate is set to 0.01 initially. The entire training process involves 300 epochs.

C. Accuracy Metrics

The accuracy assessment measures the agreements and differences between the detection result and the reference mask. The recall, precision, and mean Average Precision (mAP) are used as accuracy metrics to evaluate the performance of the methods to be compared with. The calculations of the precision and recall metrics are defined as

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{10}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{11}$$

where the true positive (TP) and true negative (TN) denote correct prediction, and the false positive (FP) and false negative (FN) denote incorrect outcome. The precision and

TABLE I
COMPARISON RESULTS OF MODEL SIZE AND INFERENCE ABILITY IN DIFFERENT BASELINE YOLO FRAMEWORKS ON THE FIRST FOLD OF THE VEDAI VALIDATION SET

Method	Layers ↓	Params ↓	GFLOPs ↓	mAP_{50} ↑
YOLOv3	270	61.5M	52.8	62.6
YOLOrs	241	20.2M	46.4	55.8
YOLOv4	393	52.5M	38.2	**65.7**
YOLOv5s	**224**	**7.1M**	**5.32**	62.2
YOLOv5m	308	21.1M	16.1	64.5
YOLOv5l	397	46.6M	36.7	63.9
YOLOv5x	476	87.3M	69.7	64.0

recall are correlated with the commission and omission errors, respectively. The mAP is a comprehensive indicator obtained by averaging AP values, which uses an integral method to calculate the area enclosed by the precision–recall curve and coordinate axis of all categories. Hence, the mAP can be calculated by

$$\text{mAP} = \frac{AP}{N} = \frac{\int_0^1 p(r)dr}{N} \tag{12}$$

where p denotes the precision, r denotes the recall, and N is the number of categories.

The giga floating-point operations per second (GFOLPs) and parameter size are used to measure the model complexity and computation cost. In addition, PSNR and SSIM are used for image quality evaluation of the SR branch. Generally, higher PSNR values and SSIM values represent the better quality of the generated image.

D. Ablation Study

First, we verify the effectiveness of our proposed method by designing a series of ablation experiments that are conducted on the first fold of the validation set.

1) Validation of the Baseline Framework: In Table I, the model size and inference ability of different base frameworks are evaluated in terms of the number of layers, parameter size, and GFLOPs. The detection performances of those models are measured by mAP_{50}, i.e., the detection metric of mAP at the intersection over union (IOU) = 0.5. Although YOLOv4 achieves the best detection performance, it has 169 more layers than YOLOv5s (393 versus 224), its parameter size (params) is 7.4 times larger than that of YOLOv5s (52.5M versus 7.1M), and its GFLOPs is 7.2 times higher than that of YOLOv5s (38.2 versus 5.3). With respect to YOLOv5s, although its mAP is slightly lower than those of YOLOv4 and YOLOv5m, its number of layers, parameter size, and GFLOPs are much smaller than those of other models. Therefore, it is easier to deploy YOLOv5s on board to achieve real-time performance in practical applications. The above fact verifies the rationality of YOLOv5s as the baseline detection framework.

2) Impact of Removing Focus Module: As presented in Section IV-A, the Focus module reduces the resolution of input images, which imposes encumbrance on the detection performance of small objects in RSI. To investigate the influence of the Focus module, we conduct experiments on the four YOLOv5 network frameworks: YOLOv5s, YOLOV5m, YOLOv5l, and YOLOv5x. Note that the results here are collected after the concatenation pixel-level fusion of RGB and IR modalities. As listed in Table II, after removing the Focus module, we observe a noticeable improvement in the detection performance of YOLOv5s (62.2%→69.5% in mAP_{50}), YOLOv5m (64.5%→72.2%), YOLOV5l (63.7%→72.5%), and YOLOv5x (64.0%→69.2%). This is because by removing the Focus module, not only can the resolution degradation be avoided, but also the spatial interval information be retained for small objects in RSI, thereby reducing the missing errors of object detection. Generally, removing the Focus module brings more than 5% improvement in the detection performance (mAP_{50}) of the whole frameworks.

Meanwhile, we notice that the above removal increases the inference computation cost (GFLOPs) in YOLOv5s (5.3→20.4), YOLOv5m (16.1→63.6), YOLOV5l (36.7→145), and YOLOv5x (69.7→276.6). However, the GFLOPs of YOLOv5s-noFocus (20.4) are smaller than those of YOLOv3 (52.8), YOLOv4 (38.2), and YOLOrs (46.4), as shown in Table I. The parameters of these models are slightly reduced after removing the Focus module. In summary, in order to retain the resolution to better detect smaller objects, priority shall be given to the detection accuracy, for which the convolution operation is adopted to replace the Focus module.

3) Comparison of Different Fusion Methods: To evaluate the influence of the devised fusion methods, we compare five fusion results on YOLOv5-noFocus, as presented in

TABLE II
INFLUENCE OF REMOVING THE FOCUS MODULE IN THE NETWORK ON THE FIRST FOLD OF THE VEDAI VALIDATION SET

Method		Params ↓	GFLOPs↓	mAP_{50} ↑
YOLOv5s	Focus	7.0739M	5.3	62.2
	noFocus	7.0705M	20.4	**69.5** (+7.3)
YOLOv5m	Focus	21.0677M	16.1	64.5
	noFocus	21.0625M	63.6	**72.2** (+7.7)
YOLOv5l	Focus	46.6406M	36.7	63.7
	noFocus	46.6337M	145.0	**72.5** (+8.8)
YOLOv5x	Focus	87.2487M	69.7	64.0
	noFocus	87.2400M	276.6	**69.2** (+5.2)

TABLE III
COMPARISON RESULT OF PIXEL- AND FEATURE-LEVEL FUSIONS IN YOLOV5S (NOFOCUS) FOR MULTIMODAL DATASET ON THE FIRST FOLD OF THE VEDAI VALIDATION SET

Method		Params ↓	GFLOPs ↓	mAP_{50} ↑
Pixel-level Fusion	Concat	**7.0705M**	**20.37**	69.5
	MF	7.0897M	21.67	**70.3**
Feature-level Fusion	Fusion1	7.0887M	21.76	66.0
	Fusion2	7.0744M	22.04	68.5
	Fusion3	7.1442M	24.22	64.8
	Fusion4	7.0870M	24.50	63.8
Multistage Feature-level Fusion		7.7545M	34.56	59.3

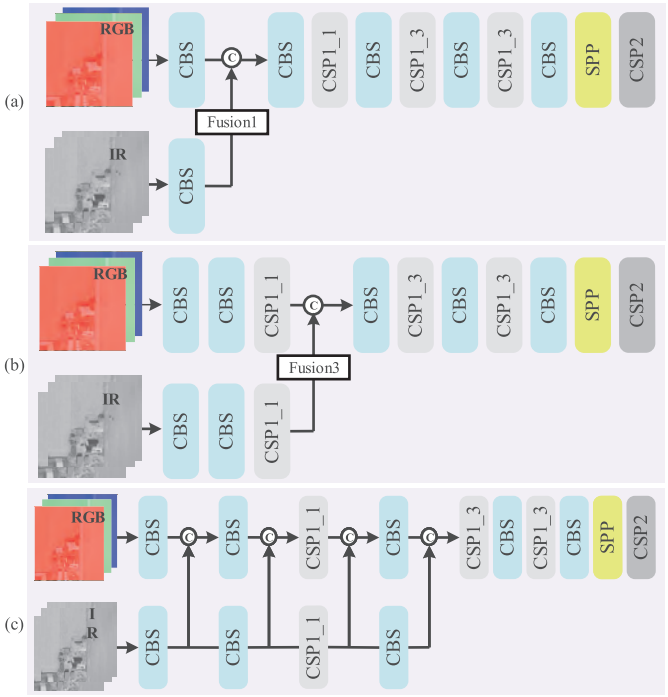

Fig. 7. Feature-level fusion of different blocks in the latent layers. Fusion-n represents the concatenation fusion operation performed in the *n*th blocks. (a) and (b) Feature-level fusion. (c) Multistage feature-level fusion.

Section IV-B. As shown in Fig. 7, fusion1, fusion2, fusion3, and fusion4 represent the concatenation fusion operation performed in the first, second, third, and fourth blocks, respectively. The IR image is expanded to three bands in feature-level fusion to obtain the features that have equal channels for the two modes. The final result is listed in Table III. The parameter size, GFLOPs, and mAP_{50} of pixel-level fusion with concatenation operation are 7.0705M, 20.37, and 69.5%, and those of the pixel-level fusion with MF module are 7.0897M, 21.67, and 70.3%, which are the best among all the compared methods. There are some reasons why the model parameters of the feature-level fusions are close to the pixel-level fusion. First, the feature-level fusion is completed in the latent layers rather than the whole two separate models. Second, the modules before the concatenation fusion are different, making the different fusion channels cause different parameters. However, it can be proved that the calculation cost is increased with the layer of fusion becoming deeper. In addition, we compare the multistage feature-level fusion [as shown in Fig. 7(c)] with the proposed pixel-level fusion. As shown in Table III, the accuracy of multistage feature-level fusion is only 59.3% mAP_{50} lower than that of pixel-level fusion, while its computation cost is 34.56 GFLOPs with 7.7545M parameters, which is higher than that of pixel-level fusion. These findings suggest that innovative pixel-level fusion methods are more effective than multistage shallow feature-level fusion because the multiple stages of fusion can lead to the accumulation of redundant information. The above results suggest that pixel-level fusion can accurately detect objects while reducing the computation. Our proposed MF fusion can improve detection accuracy with some computation costs. Overall, the proposed method only uses pixel-level fusion to contain the lower computation cost.

TABLE IV
INFLUENCE OF DIFFERENT RESOLUTIONS FOR INPUT IMAGE ON NETWORK PERFORMANCE ON THE FIRST FOLD OF THE VEDAI VALIDATION SET

Method	Train-Val Size	Test Size	Params ↓	GFLOPs ↓	mAP_{50} ↑
YOLOv5s	512	512	7.0739M	**5.3**	**62.2**
		1024	7.0739M	21.3	10.6
	1024	1024	7.0739M	21.3	**77.7**
		512	7.0739M	**5.3**	48.2
YOLOv5s (noFocus)	512	512	7.0705M	**20.4**	**69.5**
		1024	7.0705M	81.5	13.4
	1024	1024	7.0705M	81.5	**79.3**
		512	7.0705M	**20.4**	62.9
YOLOv5s (noFocus) +SR	512	512	7.0705M	**20.4**	**78.0**

4) Impact of High Resolution: We compare different training and test modes to explore more possibilities in terms of the input image resolution in Table IV. First, we compare cases where the image resolutions of the training set and the test set are the same. By comparing the result of YOLOv5s, the detection metric mAP_{50} is improved from 62.2% to 77.7%, causing a 15.5% increase when the image size is doubled from 512 to 1024. Similarly, YOLOv5s-noFocus (1024) outperforms YOLOv5s-noFocus (512) by 9.8% mAP_{50} score (79.3% versus 69.5%). The mean recall and mean precision increase simultaneously, suggesting that ensuring resolution reduces the commission and omission errors in object detection. Based on the above analysis, we argue that the characteristics of HR significantly influence the final performance of object detection. However, it is noteworthy that maintaining an HR input image of the network introduces a certain amount of calculation. The GFLOPs with a size of 1024 (HR) is higher than that with 512 (low resolution) in both YOLOv5s (21.3 versus 5.3) and YOLOv5s-noFocus (81.5 versus 20.4).

As shown in Table IV, the use of different sizes of the image during the training process (train size) and the test process (test size) results in the score reduction of mAP_{50}, i.e., (10.6% versus 62.2%), (48.2% versus 77.7%), (13.4% versus 69.5%), and (62.9% versus 79.3%). This may attribute to the inconsistent scale of objects in the test process and in the training process, where the size of the predicted bounding box is not suitable for the objects of test images anymore.

Finally, mAP_{50} of YOLOv5s-noFocus + SR is close to the YOLOv5-noFocus HR (1024) one (78.0% versus 79.3%), and the GFOLPs is equal to that of YOLOv5-noFocus LR (512) one (20.4 versus 20.4). Our proposed network decreased the resolution of input images in the test process to reduce computation and maintain accuracy by remaining the identical resolution of the training and testing data, thereby highlighting the advantage of the proposed SR branch.

5) Impact of Super Resolution Branch: Some ablation experiences about the SR branch are completed in Table V. Compared with the upsampling operation, the YOLOv5s

TABLE V
ABLATION EXPERIMENT RESULTS ABOUT THE INFLUENCE OF SR BRANCH ON DETECTION PERFORMANCE ON THE FIRST FOLD OF THE VEDAI VALIDATION SET

	Branch	Small-scale Detector	Decoder (EDSR)	L1 Loss	Params ↓	GFLOPs ↓	mAP_{50} ↑	PSNR ↑	SSIM ↑
YOLOv5s (noFocus)	Upsample				7.0705M	20.37G	76.2	-	-
	SR				7.0705M	20.37G	78.0	-	-
	SR	✓			4.8259M	16.68G	79.0	23.811	0.602
	SR	✓	✓		4.8259M	16.68G	79.9	23.902	0.604
	SR	✓	✓	✓	**4.8259M**	**16.68G**	**80.9**	**26.203**	**0.659**

TABLE VI
EFFECTIVE VALIDATION OF THE SR BRANCH FOR THE DIFFERENT BASELINE ON THE FIRST FOLD OF THE VEDAI VALIDATION SET

Method	Layers	Params ↓	GFLOPs ↓	mAP_{50} ↑
YOLOv3	270	61.5M	52.8	62.6
YOLOv3+SR	270	61.5M	52.8	**71.8**
YOLOv4	393	52.5M	38.2	65.7
YOLOv4+SR	393	52.5M	38.2	**69.0**
YOLOv5s	224	7.1M	5.3	62.2
YOLOv5s+SR	224	7.1M	5.3	**64.4**

(noFocus) added SR network shows favorable performance and gets mAP_{50} 1.8% better than upsampling operation. The SR network is a learnable upsampling method with a more vital reconstruction ability that can help the feature extraction in the backbone for detection. We deleted the PANet structure and two detectors, which are responsible for enhancing middle- and large-scale target detections because the objects in RSI datasets, such as VEDAI are on the small scale and can be detected with the small-scale detector. When we only use one detector, the number of parameters (7.0705M versus 4.8259M) and GFLOPs (20.37 versus 16.68) can be decreased, and the detection accuracy can be increased (78.0% versus 79.0%). When we utilize the EDSR network (rather than three ordinal deconvolutions) as a decoder and L1 loss (rather than L2 loss) as an SR loss function in the SR branch, which is powerful in the SR task, not only the performance of SR is improved but also the performance of the detection network enhanced meantime because the SR branch helps the detection network to extract more effective and superior features in the backbone, accelerating the convergence of the detection network and, thus, improving the performance of the detection network. The performance of SR and object detection is complementary and cooperative.

Table VI shows the favorable accuracy–complexity tradeoff of the SR branch. At the different baselines, the influence of the SR branch on object detection is positive. Compared with bare baseline, baseline added SR shows favorable performance: YOLOv3 + SR performs mAP_{50} 9.2% better than YOLOv3, YOLOv4 + SR is mAP_{50} 3.3% better than YOLOv4, and YOLOv5s + SR performs mAP_{50} 2.2% better than YOLOv5s. Notably, SR can be removed in the inference stage. Hence, no extra parameters and computation costs are introduced, which is impressive considering that the SR branch does not require a lot of manpower to refine the design of the detection network. The SR branch is general and extensible, and can be utilized in the existing FCN framework.

E. Comparisons With Previous Methods

The visual detection results of the compared YOLO methods and SuperYOLO are shown in Fig. 8, for a diverse set of scenes. It can be observed that SuperYOLO can accurately detect those objects that are not detected, or predicted into a wrong category or with uncertainty, in YOLOv4, YOLOv5s, and YOLOv5m. The objects in RSIs are challenging to detect on small scales. In particular, **Pickup** and **Car** or **Van** and **Boat** are easily confused in the detection process due to their similarities. Hence, improving the detection classification is of necessity in object detection tasks except for location detection, which can be accomplished by the proposed SuperYOLO with better performance.

Table VII summarizes the performance of the **YOLOv3** [47], **YOLOv4** [48], **YOLOv5s-x** [19], **YOLOrs** [27], **YOLO-Fine** [49], **YOLOFusion** [50], and our proposed **SuperYOLO**. Note that the AP scores of multimodal modes are significantly higher than those of unimodal (RGB or IR) modes for most classes. The overall mAP_{50} of multimodal (multi) modes outperforms those of RGB or IR modes. These results confirm that MF is an effective and efficient strategy for object detection based on information complementation between multimodal inputs. However, it should be noted that the slight increase in parameters and GFLOPs with MF reflects the necessity of choosing pixel-level fusion rather than feature-level fusion.

It is obvious that the SuperYOLO achieves higher mAP_{50} than the other frameworks except for YOLOFusion. The results of YOLOFusion are slightly better than SuperYOLO, as YOLOFusion uses pretrained weight which is trained on MS COCO [7]. However, its parameter count is approximately three times that of SuperYOLO. The performance of YOLO-Fine is good on a single modality, but it lacks the development of multimodality fusion techniques. In particular, the SuperYOLO outperforms the YOLOv5x by a 12.44% mAP_{50} score in multimodal mode. Meanwhile, parameter size and GFLOPs of SuperYOLO are about 18× and 3.8× less than YOLOv5x.

In addition, it can be noticed that superior performance is achieved for the classes of **Car**, **Pickup**, **Tractor**, and **Camping**, which have the most training instances. YOLOv5s performs superior on GFLOPs, which depends on the Focus

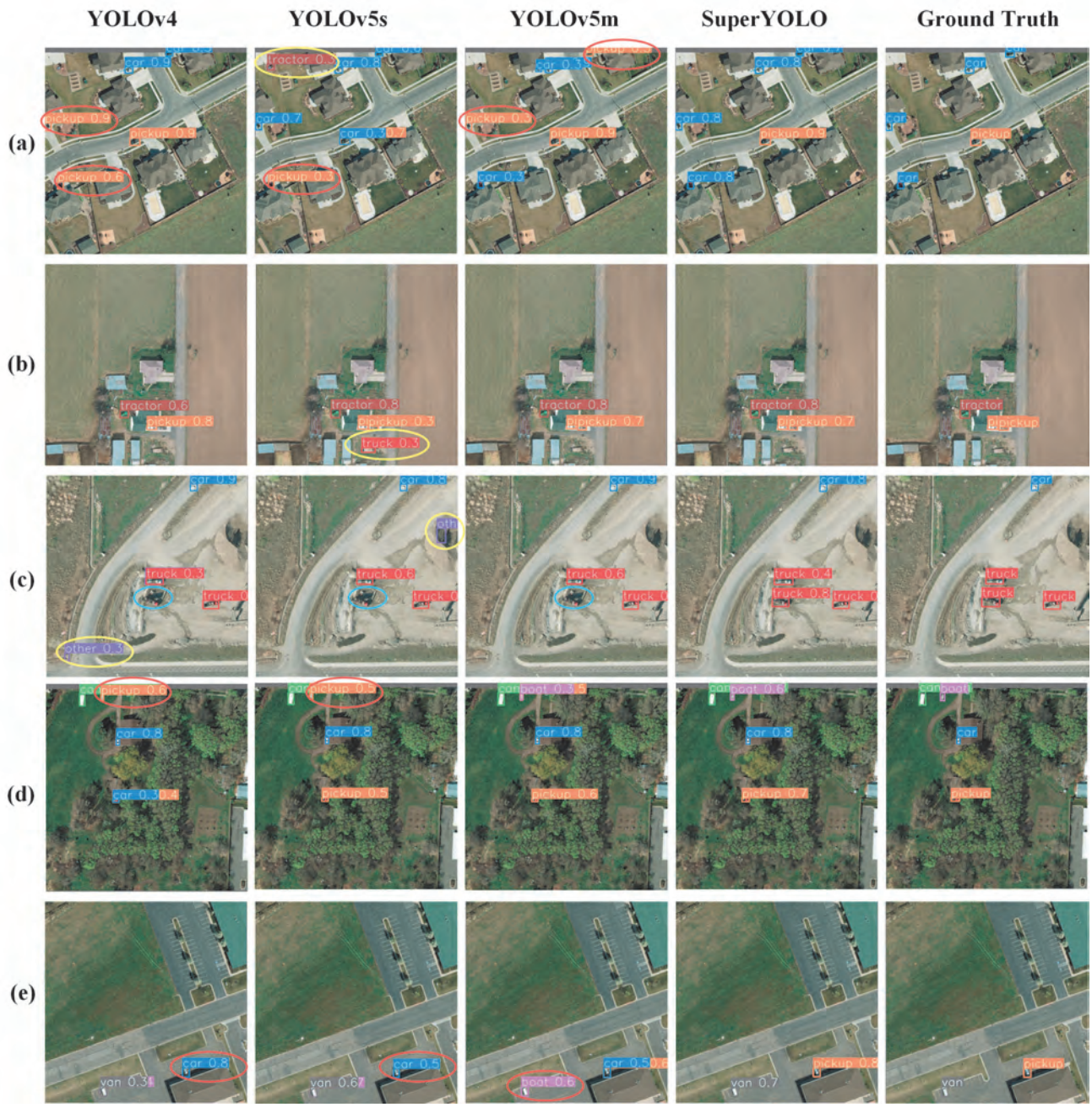

Fig. 8. Visual results of object detection using different methods involving YOLOv4, YOLOv5s, YOLOv5m, and the proposed SuperYOLO. The red cycles represent the false alarms, the yellow ones denote the FP detection results, and the blue ones are FN detection results. (a)–(e) Different images in the VEDAI dataset.

module to slim the input image, but results in lousy detection performance, especially for small objects. The SuperYOLO performs 18.30% mAP_{50} better than YOLOv5s. Our proposed SuperYOLO shows a favorable speed–accuracy tradeoff compared to the state-of-the-art models.

F. Generalization to Single Modal Remote Sensing Images

At present, although there are massive multimodal images in remote sensing, the labeled dataset in object detection tasks is lacking due to the expensive cost of manually annotating. To validate the generalization of our proposed network, we compare the SuperYOLO with different one- or two-stage methods using data from the single modality, including a large-scale Dataset for Object Detection in Aerial images (DOTA), object DetectIon in Optical Remote sensing images (DIOR), and Northwestern Polytechnical University Very-High-Resolution 10-class (NWPU VHR-10) datasets.

1) DOTA: The DOTA dataset was proposed in 2018 for object detection of remote sensing. It contains 2806 large images and 188 282 instances, which are divided into 15 categories. The size of each original image is 4000 × 4000, and the images are cropped into 1024 × 1024 pixels with an overlap of 200 pixels in the experiment. We select half of the original images as the training set, 1/6 as the validation set, and 1/3 as the testing set. The size of the image is fixed to 512 × 512.

TABLE VII

CLASSWISE AVERAGE PRECISION AP, MEAN AVERAGE PRECISION mAP_{50}, PARAMETERS, AND GFLPS FOR THE PROPOSED SUPERYOLO, YOLOV3, YOLOV4, YOLOV5S-X, YOLORS, YOLO-FINE, AND YOLOFUSION, INCLUDING UNIMODAL AND MULTIMODAL CONFIGURATIONS ON VEDAI DATASET. * REPRESENTS USING PRETRAINED WEIGHT

Method		Car	Pickup	Camping	Truck	Other	Tractor	Boat	Van	mAP_{50} ↑	Params ↓	GFLOPs ↓
YOLOv3 [47]	IR	80.21	67.03	65.55	47.78	25.86	40.11	32.67	53.33	51.54	**61.5351M**	**49.55**
	RGB	83.06	71.54	**69.14**	59.30	**48.93**	**67.34**	33.48	55.67	61.06	**61.5351M**	**49.55**
	Multi	**84.57**	**72.68**	67.13	**61.96**	43.04	65.24	**37.10**	**58.29**	**61.26**	61.5354M	49.68
YOLOv4 [48]	IR	80.45	67.88	68.84	53.66	30.02	44.23	25.40	51.41	52.75	**52.5082M**	**38.16**
	RGB	83.73	**73.43**	71.17	59.09	**51.66**	65.86	**34.28**	**60.32**	62.43	**52.5082M**	**38.16**
	Multi	**85.46**	72.84	**72.38**	**62.82**	48.94	**68.99**	**34.28**	54.66	**62.55**	52.5085M	38.23
YOLOv5s [19]	IR	77.31	65.27	66.47	51.56	25.87	42.36	21.88	48.88	49.94	**7.0728M**	**5.24**
	RGB	80.07	68.01	66.12	51.52	45.76	**64.38**	21.62	40.93	54.82	**7.0728M**	**5.24**
	Multi	**80.81**	**68.48**	**69.06**	**54.71**	**46.76**	64.29	**24.25**	**45.96**	**56.79**	7.0739M	5.32
YOLOv5m [19]	IR	79.23	67.32	65.43	51.75	26.66	44.28	26.64	56.14	52.19	**21.0659M**	**16.13**
	RGB	81.14	70.26	65.53	53.98	**46.78**	**66.69**	**36.24**	49.87	58.80	**21.0659M**	**16.13**
	Multi	**82.53**	**72.32**	**68.41**	**59.25**	46.20	66.23	33.51	**57.11**	**60.69**	21.0677M	16.24
YOLOv5l [19]	IR	80.14	68.57	65.37	53.45	30.33	45.59	27.24	**61.87**	54.06	**46.6383M**	**36.55**
	RGB	81.36	71.70	68.25	57.45	45.77	**70.68**	35.89	55.42	60.81	**46.6383M**	**36.55**
	Multi	**82.83**	**72.32**	**69.92**	**63.94**	**48.48**	63.07	**40.12**	56.46	**62.16**	46.6406M	36.70
YOLOv5x [19]	IR	79.01	66.72	65.93	58.49	31.39	41.38	31.58	58.98	54.18	**87.2458M**	**69.52**
	RGB	81.66	72.23	68.29	59.07	48.47	66.01	**39.15**	**61.85**	62.09	**87.2458M**	**69.52**
	Multi	**84.33**	**72.95**	**70.09**	**61.15**	**49.94**	**67.35**	38.71	56.65	**62.65**	87.2487M	69.71
YOLOrs [27]	IR	82.03	73.92	63.80	**54.21**	43.99	54.39	**21.97**	43.38	54.71	-	-
	RGB	**85.25**	72.93	**70.31**	50.65	42.67	**76.77**	18.65	38.92	57.00	-	-
	Multi	84.15	**78.27**	68.81	52.60	**46.75**	67.88	21.47	**57.91**	**59.73**	-	-
YOLO-Fine [49]	IR	76.77	74.35	64.74	63.45	**45.04**	**78.12**	**70.04**	**77.91**	68.18	-	-
	RGB	**79.68**	**74.49**	**77.09**	**80.97**	37.33	70.65	60.84	63.56	**68.83**	-	-
YOLOFusion* [50]	IR	86.7	75.9	66.6	77.1	43.0	62.3	70.7	**84.3**	70.8	-	-
	RGB	91.1	82.3	75.1	**78.3**	33.3	**81.2**	**71.8**	62.2	71.9	-	-
	Multi	**91.7**	**85.9**	**78.9**	78.1	**54.7**	71.9	71.7	75.2	**75.9**	12.5M	-
SuperYOLO	IR	87.90	81.39	76.90	61.56	39.39	60.56	46.08	71.00	65.60	**4.8256M**	**16.61**
	RGB	90.30	82.66	76.69	68.55	53.86	79.48	58.08	70.30	72.49	**4.8256M**	**16.61**
	Multi	**91.13**	**85.66**	**79.30**	**70.18**	**57.33**	**80.41**	**60.24**	**76.50**	**75.09**	4.8451M	17.98

2) NWPU VHR-10: The dataset of NWPU VHR-10 was proposed in 2016. It contains 800 images, of which 650 pictures contain objects, so we use 520 images as the training set and 130 images as the testing set. The dataset contains ten categories, and the size of the image is fixed to 512 × 512.

3) DIOR: The DIOR dataset was proposed in 2020 for the task of object detection, which involves 23 463 images and 192 472 instances. The size of each image is 800 × 800. We choose 11 725 images as the training set and 11 738 images as the testing set. The size of the image is fixed to 512 × 512.

The training strategy is modified to accommodate the new dataset. The entire training process involves 150 epochs for NWPU and DIOR datasets, and 100 epochs for DOTA. The batch size of the DOTA and DIOR is 16 and of NWPU is 8. To verify the superiority of the SuperYOLO proposed in this article, we selected 11 generic methods for comparison: one-stage algorithms (**YOLOv3** [47], **FCOS** [53], **ATSS** [54], **RetainNet** [51], and **GFL** [52]); two-stage method (**Faster R-CNN** [5]); lightweight models (**MobileNetV2** [55] and **ShuffleNet** [56]); distillation-based methods (**ARSD** [59]); and remote sensing designed approaches (**FMSSD** [58] and **O2DNet** [57]).

As presented in Table VIII, our SuperYOLO achieves the optimal detection result (69.99%, 93.30%, and 71.82% mAP_{50}), and the model parameters (7.70M, 7.68M, and 7.70M) and GFLOPs (20.89, 20.86, and 20.93) are much smaller than other SOTA detectors regardless of the two-stage, one-stage, lightweight, or distillation-based method. The PANet structure and three detectors are responsible for enhancing small-, middle-, and large-scale target detections in consideration of the big objects, such as playgrounds in these three datasets. Hence, the model parameters of SuperYOLO are more than those in Table VII. We also compare two detectors designed for RSI, such as FMSSD [58] and O2DNet [57].

TABLE VIII
PERFORMANCE OF DIFFERENT ALGORITHMS ON DOTA, NWPU, AND DOTA TESTING SETS

	DOTA-v1.0			NWPU			DIOR		
Method	**mAP$_{50}$**	**Params(M)**	**GFLOPs**	**mAP$_{50}$**	**Params(M)**	**GFLOPs**	**mAP$_{50}$**	**Params(M)**	**GFLOPs**
Faster R-CNN [5]	60.64	60.19	289.25	77.80	41.17	127.70	54.10	60.21	182.20
RetainNet [51]	50.39	55.39	293.36	89.40	36.29	123.27	65.70	55.49	180.62
YOLOv3 [47]	60.00	61.63	198.92	88.30	61.57	121.27	57.10	61.95	122.22
GFL [52]	66.53	19.13	159.18	88.80	19.13	91.73	68.00	19.13	97.43
FCOS [53]	67.72	31.57	202.15	89.65	31.86	116.63	67.60	31.88	123.51
ATSS [54]	66.84	18.97	156.01	90.50	18.96	89.90	67.70	18.98	95.50
MobileNetV2 [55]	56.91	10.30	124.24	76.90	10.29	71.49	58.20	10.32	76.10
ShuffleNet [56]	57.73	12.11	142.60	83.00	12.10	82.17	61.30	12.12	87.31
O2-DNet [57]	71.10	209.0	-	-	-	-	68.3	209.0	-
FMSSD [58]	**72.43**	136.04	-	-	-	-	69.5	136.03	-
ARSD [59]	68.28	13.08	68.03	90.92	11.57	26.65	70.10	13.10	41.60
SuperYOLO	69.99	**7.70**	**20.89**	**93.30**	**7.68**	**20.86**	**71.82**	**7.70**	**20.93**

Although these models have a close performance with our lightweight model, the huger parameters and GFLOPs seem to be a massive cost in computation resources. Hence, our model has a better balance in consideration of detection efficiency and efficacy.

VI. CONCLUSION AND FUTURE WORK

In this article, we have presented SuperYOLO, a real-time lightweight network that is built on top of the widely used YOLOv5s to improve the detection performance of small objects in RSI. First, we have modified the baseline network by removing the Focus module to avoid resolution degradation, through which the baseline is significantly improved and overcomes the missing error of small objects. Second, we have conducted research fusion of multimodality to improve the detection performance based on mutual information. Lastly and most importantly, we have introduced a simple and flexible SR branch facilitating the backbone to construct an HR representation feature, by which small objects can be easily recognized from vast backgrounds with merely LR input required. We remove the SR branch in the inference stage, accomplishing the detection without changing the original structure of the network to achieve the same GFOLPs. With joint contributions of these ideas, the proposed SuperYOLO achieves 75.09% mAP$_{50}$ with lower computation cost on the VEDAI dataset, which is 18.30% higher than that of YOLOv5s, and more than 12.44% higher than that of YOLOv5x.

The performance and inference ability of our proposal highlight the value of SR in remote sensing tasks, paving way for the future study of multimodal object detection. Our future interests will be focusing on the design of a low-parameter mode to extract HR features, thereby further satisfying real-time and high-accuracy motivations.

REFERENCES

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[2] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[4] P. Tang, X. Wang, X. Bai, and W. Liu, "Multiple instance detection network with online instance classifier refinement," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3059–3067.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[6] D. Jia, D. Wei, R. Socher, J. Lili, K. Li, and F. Li, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 248–255.

[7] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2014, pp. 740–755.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[9] Z. Zheng, Y. Zhong, J. Wang, and A. Ma, "Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4096–4105.

[10] J. Pang, C. Li, J. Shi, Z. Xu, and H. Feng, "R^2-CNN: Fast tiny object detection in large-scale remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5512–5524, Aug. 2019.

[11] Z. Deng, H. Sun, S. Zhou, J. Zhao, L. Lei, and H. Zou, "Multi-scale object detection in remote sensing imagery with convolutional neural networks," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 3–22, Apr. 2018.

[12] J. Ding, N. Xue, Y. Long, G.-S. Xia, and Q. Lu, "Learning RoI transformer for oriented object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2844–2853.

[13] Z. Liu, H. Wang, H. Weng, and L. Yang, "Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 8, pp. 1074–1078, Aug. 2016.

[14] D. Hong et al., "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, May 2021.

[15] Z. Wang, K. Jiang, P. Yi, Z. Han, and Z. He, "Ultra-dense GAN for satellite imagery super-resolution," *Neurocomputing*, vol. 398, pp. 328–337, Jul. 2020.

[16] M. T. Razzak, G. Mateo-García, G. Lecuyer, L. Gómez-Chova, Y. Gal, and F. Kalaitzis, "Multi-spectral multi-image super-resolution of Sentinel-2 with radiometric consistency losses and its effect on building delineation," *ISPRS J. Photogramm. Remote Sens.*, vol. 195, pp. 1–13, Jan. 2023.

[17] K. Jiang, Z. Wang, P. Yi, G. Wang, T. Lu, and J. Jiang, "Edge-enhanced GAN for remote sensing image superresolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5799–5812, Jun. 2019.
[18] Y. Xiao, X. Su, Q. Yuan, D. Liu, H. Shen, and L. Zhang, "Satellite video super-resolution via multiscale deformable convolution alignment and temporal grouping projection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5610819.
[19] (2021). *Ultralytics/Yolov5:v5.0*. [Online]. Available: https://github.com/ultralytics/yolov5
[20] S. Zhang, M. Chen, J. Chen, F. Zou, Y.-F. Li, and P. Lu, "Multimodal feature-wise co-attention method for visual question answering," *Inf. Fusion*, vol. 73, pp. 1–10, Sep. 2021.
[21] Y.-T. Chen, J. Shi, Z. Ye, C. Mertz, D. Ramanan, and S. Kong, "Multimodal object detection via probabilistic ensembling," 2021, *arXiv:2104.02904*.
[22] Q. Chen et al., "EF-Net: A novel enhancement and fusion network for RGB-D saliency detection," *Pattern Recognit.*, vol. 112, Apr. 2021, Art. no. 107740.
[23] H. Zhu, M. Ma, W. Ma, L. Jiao, and B. Hou, "A spatial-channel progressive fusion ResNet for remote sensing classification," *Inf. Fusion*, vol. 70, no. 1, pp. 72–87, 2020.
[24] Y. Sun, Z. Fu, C. Sun, Y. Hu, and S. Zhang, "Deep multimodal fusion network for semantic segmentation using remote sensing image and LiDAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5404418.
[25] W. Li, Y. Gao, M. Zhang, R. Tao, and Q. Du, "Asymmetric feature fusion network for hyperspectral and SAR image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Feb. 18, 2022, doi: 10.1109/TNNLS.2022.3149394.
[26] Y. Gao et al., "Hyperspectral and multispectral classification for coastal wetland using depthwise feature interaction network," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5512615.
[27] M. Sharma et al., "YOLOrs: Object detection in multimodal remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1497–1508, 2021.
[28] L. Gómez-Chova, D. Tuia, G. Moser, and G. Camps-Valls, "Multimodal classification of remote sensing images: A review and future directions," *Proc. IEEE*, vol. 103, no. 9, pp. 1560–1584, Sep. 2015.
[29] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.
[30] C. Li, T. Yang, S. Zhu, C. Chen, and S. Guan, "Density map guided object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 190–191.
[31] C. Chen, M.-Y. Liu, O. Tuzel, and J. Xiao, "R-CNN for small object detection," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2017, pp. 214–230.
[32] J. Noh, W. Bae, W. Lee, J. Seo, and G. Kim, "Better to follow, follow to be better: Towards precise supervision of feature super-resolution for small object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9725–9734.
[33] M. Haris, G. Shakhnarovich, and N. Ukita, "Task-driven super resolution: Object detection in low-resolution images," 2018, *arXiv:1803.11316*.
[34] J. Shermeyer and A. Van Etten, "The effects of super-resolution on object detection performance in satellite imagery," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 1432–1441.
[35] L. Courtrai, M.-T. Pham, and S. Lefèvre, "Small object detection in remote sensing images based on super-resolution with auxiliary generative adversarial networks," *Remote Sens.*, vol. 12, no. 19, p. 3152, Sep. 2020.
[36] J. Rabbi, N. Ray, M. Schubert, S. Chowdhury, and D. Chao, "Small-object detection in remote sensing images with end-to-end edge-enhanced GAN and object detector network," *Remote Sens.*, vol. 12, no. 9, p. 1432, May 2020.
[37] H. Ji, Z. Gao, T. Mei, and B. Ramesh, "Vehicle detection in remote sensing images leveraging on simultaneous super-resolution," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 4, pp. 676–680, Apr. 2020.
[38] L. Wang, D. Li, Y. Zhu, L. Tian, and Y. Shan, "Dual super-resolution learning for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3773–3782.
[39] C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 1571–1580.
[40] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Netw.*, vol. 107, pp. 3–11, Nov. 2018.
[41] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2014.
[42] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[43] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 136–144.
[44] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, Mar. 2017.
[45] S. Razakarivony and F. Jurie, "Vehicle detection in aerial imagery: A small target detection benchmark," *J. Vis. Commun. Image Represent.*, vol. 34, pp. 187–203, Jan. 2016.
[46] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. 19th Int. Conf. Comput. Statist.*, vol. 2010, pp. 177–186.
[47] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[48] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[49] M.-T. Pham, L. Courtrai, C. Friguet, S. Lefèvre, and A. Baussard, "YOLO-fine: One-stage detector of small objects under various backgrounds in remote sensing images," *Remote Sens.*, vol. 12, no. 15, p. 2501, Aug. 2020.
[50] F. Qingyun and W. Zhaokui, "Cross-modality attentive feature fusion for object detection in multispectral remote sensing imagery," *Pattern Recognit.*, vol. 130, Oct. 2022, Art. no. 108786.
[51] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[52] X. Li et al., "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 21002–21012.
[53] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9627–9636.
[54] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9759–9768.
[55] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.
[56] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.
[57] H. Wei, Y. Zhang, Z. Chang, H. Li, H. Wang, and X. Sun, "Oriented objects as pairs of middle lines," *ISPRS J. Photogramm. Remote Sens.*, vol. 169, pp. 268–279, Nov. 2020.
[58] P. Wang, X. Sun, W. Diao, and K. Fu, "FMSSD: Feature-merged single-shot detection for multiscale objects in large-scale remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3377–3390, Dec. 2020.
[59] Y. Yang et al., "Adaptive knowledge distillation for lightweight remote sensing object detectors optimizing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5623715.

Jiaqing Zhang received the B.E. degree in telecommunications engineering from Ningbo University, Ningbo, Zhejiang, China, in 2019. She is currently pursuing the Ph.D. degree with the Image Coding and Processing Center, State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, China.

Her research interests include multimodal image processing, remote sensing object detection, and network compression.

Jie Lei (Member, IEEE) received the M.S. degree in telecommunication and information systems and the Ph.D. degree in signal and information processing from Xidian University, Xi'an, China, in 2006 and 2010, respectively.

He was a Visiting Scholar with the Department of Computer Science, University of California at Los Angeles, Los Angeles, CA, USA, from 2014 to 2015. He is currently a Professor with the School of Telecommunications Engineering, Xidian University, where he is also a member of the Image Coding and Processing Center, State Key Laboratory of Integrated Services Networks. He is also with the Science and Technology on Electro-Optic Control Laboratory, Luoyang, China. His research interests focus on image and video processing, computer vision, and customized computing for big-data applications.

Weiying Xie (Member, IEEE) received the B.S. degree in electronic information science and technology from the University of Jinan, Jinan, China, in 2011, the M.S. degree in communication and information systems from Lanzhou University, Lanzhou, China, in 2014, and the Ph.D. degree in communication and information systems from Xidian University, Xi'an, China, in 2017.

She is currently an Associate Professor with the State Key Laboratory of Integrated Services Networks, Xidian University. Her research interests include neural networks, machine learning, hyperspectral image processing, and high-performance computing.

Zhenman Fang (Member, IEEE) received the Ph.D. degree in computer science from Fudan University, Shanghai, China, in 2014.

He did his post-doctoral research at the University of California at Los Angeles (UCLA), Los Angeles, CA, USA, from 2014 to 2017, and worked as a Staff Software Engineer at Xilinx, San Jose, CA, USA, from 2017 to 2019. He is currently an Assistant Professor with the School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada. His recent research focuses on customizable computing with specialized hardware acceleration, including emerging application characterization and acceleration, novel accelerator-rich and near-data computing architecture designs, and corresponding programming, runtime, and tool support.

Dr. Fang is a member of the Association for Computing Machinery (ACM).

Yunsong Li (Member, IEEE) received the M.S. degree in telecommunication and information systems and the Ph.D. degree in signal and information processing from Xidian University, Xi'an, China, in 1999 and 2002, respectively.

He joined the School of Telecommunications Engineering, Xidian University, in 1999, where he is currently a Professor. He is currently the Director of the Image Coding and Processing Center, State Key Laboratory of Integrated Services Networks, Xidian University. His research interests focus on image and video processing and high-performance computing.

Qian Du (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the University of Maryland, Baltimore, MD, USA, in 2000.

She is currently the Bobby Shackouls Professor with the Department of Electrical and Computer Engineering, Mississippi State University, Starkville, MS, USA. She is also an Adjunct Professor with the College of Surveying and Geo-Informatics, Tongji University, Shanghai, China. Her research interests include hyperspectral remote sensing image analysis and applications, pattern classification, data compression, and neural networks.

Dr. Du is a fellow of the SPIE-International Society for Optics and Photonics. She received the 2010 Best Reviewer Award from the IEEE Geoscience and Remote Sensing Society. She was the Co-Chair of the Data Fusion Technical Committee of the IEEE Geoscience and Remote Sensing Society from 2009 to 2013 and the Chair of the Remote Sensing and Mapping Technical Committee of the International Association for Pattern Recognition from 2010 to 2014. She was the General Chair of the fourth IEEE GRSS Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing held in Shanghai, China, in 2012. She has served as an Associate Editor for the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING, the *Journal of Applied Remote Sensing*, and the IEEE SIGNAL PROCESSING LETTERS. Since 2016, she has been the Editor-in-Chief of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING.